DESIGN AND CONTROL

OF

HUMAN-FRIENDLY ROBOTS

DESIGN AND CONTROL

OF

HUMAN-FRIENDLY ROBOTS

BY

LINGQI ZENG, M. ENG.

A Thesis

Submitted to the School of Graduate Studies

in Partial Fulfillment of the Requirements

for the Degree

Doctorate of Philosophy

McMaster University

DOCTORATE OF PHILOSOPHY MCMASTER UNIVERSITY

(Mechanical Engineering) Hamilton, Ontario

TITLE:	Design and control of human friendly robots
AUTHOR:	Lingqi Zeng, M. Eng (Xi'an University of Technology)
SUPERVISOR:	Dr. Gary M. Bone
NUMBER OF PAGE:	xlvii, 279

Abstract

In this thesis, solutions to two of the problems encountered in the design and control of human-friendly robots are investigated. The first problem is severe human injuries can occur when an accidental human-manipulator impact happens. A theoretical and experimental study on using foam coverings to reduce the severity of a human-manipulator impact and enhance human safety is presented. An improved human-manipulator impact model that incorporates the manipulator dynamics, foam covering dynamics and the coupling between the human head and torso is introduced. A method for approximating the configuration-dependent dynamics of robotics manipulators with the dynamics of a single DOF manipulator is proposed. With this model, the design parameters that significantly influence the human head acceleration are investigated. A model-based foam covering design procedure to properly select parameters of foam coverings in accordance with safety criteria and the foam thickness constraint is then proposed. The impact model and the foam covering design procedure are validated experimentally with two manipulators. The maximum error between the predicted and experimental head acceleration was less than 9%. The maximum error between the predicted and experimental foam compressed depth was less than 12%.

The second problem is mobile robot navigation in the presence of humans and other motion-unpredictable obstacles. A novel navigation algorithm, based on the virtual force field (VFF) method, is proposed as a solution. It features improved functions for the repulsive and detour virtual forces, and a new stabilizing virtual force. Methods to

calculate sizes of the active and critical regions for different obstacles are developed. Stability of the new VFF is proven using a novel piecewise Lyapunov function and Lyapunov's second method. Based on simulations for different obstacle configurations, the new VFF-based algorithm successfully produces collision-free paths while five well-known navigation algorithms incurred collisions in one of the configurations. With the new VFF-based navigation algorithm, simulations and experiments are successfully performed with a holonomic robot and a nonholonomic robot for several configurations, including multiple moving obstacles.

Acknowledgement

Foremost, I would like to express my appreciations to Dr. Gary Bone, my supervisor, for his inspiration, encouragement, direction and advices. He helped me overcome many difficulties in the research and gave me so much assistance with this thesis.

I also thank Dr. Tim Nye, Dr Stephen Veldhuis, and Dr. Shahin Sirouspour. Their encouragement gave me more confidence in my study and research; their comments and suggestions helped me make this research more complete.

I thank my colleagues not only for their direct assistance and discussions, but also for their friendship.

I will like to thank my wife and daughter, my parents for their love, care, patient and support. Without them, I would not complete this thesis.

Table of contents

List of figures

List of tables

List of Symbols

a_2, a_3	Lengths of the second and third links
a_h	Human head acceleration
$a_{h,\max}$	Peak head acceleration during an impact
$\mathbf{a}_{mr}$	Vector of mobile robot's acceleration
$\overline{a}_{mr}$	Mobile robot acceleration/deceleration limit
a_{mrd}	Magnitude of the mobile robot's deceleration
a_{mrx}, a_{mry}	X and Y components of the mobile robot acceleration vector
$\mathbf{a}_o$	Linear acceleration of the obstacle
A_{dd}	Parameter related to the viscous friction and effective moment of inertia of the direct drive manipulator
$A_h(s)$	Human head acceleration in the frequency domain
$A_{mr,i}$	Parameter related to the viscous friction and effective moment of inertia of the i^{th} wheel of the holonomic robot
$A_{nh,i}$	Parameter related to the viscous friction and effective moment of inertia of the i^{th} wheel of the nonholonomic robot
$\mathbf{A}_{mr,i}$	Matrix used to describe the i^{th} wheel dynamics in state space form
b	VFF gain ratio
$b_{1,r}$, $b_{2,r}$	Coefficients related to the dynamic parameters of the manipulator
b_{tube}	Thickness of the tube wall
B	Distance between the robot's position and the boundary of the

	rectangular obstacle perpendicular to $V_{mr,n}$
$\mathbf{B}$	Vector of the Coriolis and centripetal torques of a manipulator
$\mathbf{B}_{mr,i}$	Matrix used to describe the i^{th} wheel dynamics in state space form
$\mathbf{B}_{23}$	Coriolis and centripetal torques vector for joints 2 and 3
B_{dd}	Summed motor viscous friction coefficient
B_i	i^{th} row element of the $\mathbf{B}$ vector
B_{len}, B_{wid}	Length and width of the rectangular obstacle
$B_{mr,i}$	Viscous friction coefficient of the i^{th} motor and gear for the holonomic robot
$B_{nh,i}$	Viscous friction coefficient of the i^{th} motor and gear for the nonholonomic robot
c_i	Boolean coefficient for the i^{th} obstacle
$\mathbb{C}_1$	Region excluding $\mathbb{C}_2$ and $\mathbb{C}_3$
$\mathbb{C}_2$	Active region of the obstacle
$\mathbb{C}_{2,cir}$,	Active region of the stationary circular obstacle
$\mathbb{C}_{2,mobile}$	Active regions of the Obstacle Robot
$\mathbb{C}_{2,rect}$	Active region of the stationary rectangular obstacle
$\mathbb{C}_3$	Critical region of the obstacle
$\mathbb{C}_{3,cir}$	Critical region of the stationary circular obstacle
$\mathbb{C}_{3,mobile}$	Critical region of the Obstacle Robot
$\mathbb{C}_{3,rect}$	Critical region of the stationary rectangular obstacle
C_c	Foam damping

C_{cont}	Control damping of a joint
C_h	Damping of the human neck
C_J	Damping of a revolute joint
$C_{J\,i}$	Damping of the i^{th} joint
C_{mech}	Mechanical damping of a joint
$\mathbf{C}_{mr,i}$	Vector used to describe the i^{th} wheel dynamics in state space form
$C_{r\,i}$	Effective manipulator stiffness for the i^{th} impact scenario
C_x, C_y	Coordinates of the centre of radial lens distortion
d , $\dot{d}$	Euclidean norm of **D**, and its time derivative
$d_{circular}$	Centre to centre distance between the mobile robot and the stationary circular obstacle
$d_{cir,\min}$	Minimum $d_{circular}$ during navigation
d_{disk}	Distance from the centre of mass of the disk to the joint centre.
d_h	Centre-to-centre distance between the mobile robot and human
$d_{h,\min}$	Minimum d_h during navigation
d_i	Centre-to-centre distance between the mobile robot and i^{th} obstacle
$\dot{d}_i$	Time derivative of d_i
d_{k2}	Size of the active region used by Khatib
d_{mobile}	Centre-to-centre distance between the Obstacle Robot and the mobile robot
d_{rect}	Shortest distance between the mobile robot and the stationary rectangular obstacle model

$d_{rect,\min}$	Minimum d_{rect} during navigation
d_s	Shortest distance between the model of the mobile robot and the exterior contour of an obstacle model
$d_{s,\min}$	Minimum d_s during navigation
$d_{s,\min,cir}$	Shortest distance between the mobile robot model and the model of the stationary circular obstacle
$d_{s,\min,h}$	Shortest distance between the mobile robot model and the model of the human
$d_{s,\min,i}$	$d_{s,\min}$ for the i^{th} obstacle
$d_{s,\min,rect}$	Shortest distance between the mobile robot model and the model of the stationary rectangular obstacle
d_{tube}	Distance from the centre of mass of the tube to the joint centre.
$\mathbf{D}$, $\dot{\mathbf{D}}$	Vector from the obstacle centre to the robot centre, and its time derivative
e_{dd}	Error between the desired angular position and the actual angular position of the joint of the direct drive manipulator
$e_{mr,i}$	Error between the desired angular position and the actual angular position of i^{th} wheel of the holonomic robot
$e_{nh,i}$	Error between the desired angular position and the actual angular position of i^{th} wheel of the nonholonomic robot
$\mathbf{E}$, e	Vector from the robot to its goal, and its Euclidean norm
$\dot{\mathbf{E}}$	Time derivative of $\mathbf{E}$

$E_m(s)$	Error between the actual joint dynamics and the reduced order model in the frequency domain
E_x, $\dot{E}_x$, $\ddot{E}_x$	Length, time derivative and second time derivative of **E** along the X-axis
E_y, $\dot{E}_y$, $\ddot{E}_y$	Length, time derivative and second time derivative of **E** along the Y-axis
f	Vector of the human-manipulator contact force
f_c	Human-manipulator impact force
f_{ci}	Impact force for the i^{th} impact scenario
f_{cam}	Focal length of the camera
$f_{friction}$	Friction force on the apparatus used to measure the foam stiffness and damping
f_{M3}, f_{C3}, f_{K3}	Forces caused by mass, damping and stiffness in third impact scenario, respectively
f_{static}	Static force on the apparatus used to measure the foam stiffness and damping
f_t	Tangential force from the human and manipulator impact
$\mathbf{F}_a$	Attractive virtual force
$F_c(s)$	Impact force, f_c, in the frequency domain
$F_{c,cst}$	The constrained impact force in the frequency domain
$\mathbf{F}_L$	Stabilizing virtual force
$\mathbf{F}_v$	Total virtual force

$\mathbf{F}_{v,\mathbf{P}_r \in \mathbb{C}_i}$	Total virtual force in $\mathbb{C}_i$ for i =1 and 2
F_x, F_y	X and Y components of the total virtual force
$\mathbf{F}_\Lambda$	Repulsive virtual force
$\mathbf{F}_{\Lambda,i}$	Repulsive virtual force from the i^{th} obstacle
$\mathbf{F}_\psi$	Detour virtual force
$\mathbf{F}_{\psi,i}$	Detour virtual force from the i^{th} obstacle
$\mathbf{F}'_\psi$	Negative of $\mathbf{F}_\psi$
g	Gravitational acceleration
h_{tube}	Height of the tube
h_Λ , h_ψ	Positive time varying coefficients used in the stability analysis
$h_{\Lambda,i}$, $h_{\psi,i}$	h_Λ and h_ψ values for the i^{th} obstacle
H	Total performance criterion for mobile robot navigation
H_D	Avoidance criterion for mobile robot navigation
H_E	Energy loss criterion for mobile robot navigation
HIC	Value of the head injury criterion
H_i	H value for i^{th} obstacle configuration
H_{mean}	Mean value of H for obstacle configurations 1, 2, and 3
H_{opt}	Value of H_{mean} with the optimal gains
H_T	Arrival time criterion for mobile robot navigation
i	Index
$\mathbf{i}$	Vector of the currents that the motors use to drive the robot

$\mathbf{i}^*$	Vector of the currents that the motors use to drive the robot to reach its goal without obstacles
$i_{dd,i}$	Current for the i^{th} motor of the direct drive manipulator
i_i	Current for the i^{th} motor of the mobile robot
I_{disk}, $I_{disk,centre}$	Moment of inertia of the disk in the direct drive manipulator
I_J	Moment of inertia driven by the joint
$I_{J\,i}$	Moment of inertia of the i^{th} joint
I_{tube}, $I_{tube,centre}$	Moment of inertia of the tube used in the direct drive manipulator
$\mathbf{J}$	Jacobian matrix of a manipulator
$\mathbf{J}_{23}$	Jacobian matrix for joints 2 and 3 of a manipulator
J_{ij}	Element from the i^{th} row and j^{th} column of $\mathbf{J}$
$J_{mr,i}$	Effective inertia of the robot plus the inertia of the i^{th} motor, gear and wheel for the holonomic robot
$J_{nh,i}$	Effective inertia of the robot plus the inertia of the i^{th} motor, gear and wheel for the nonholonomic robot
k	Boolean coefficient to calculate H
k_{nh}	Positive gain used in the nonholonomic robot controller
k_{rad}	Radial lens distortion coefficient
K_1, K_2	Positive attractive force gains
K_3, K_4	Positive repulsive force gains
K_5, K_6	Positive detour force gains
K_{amp}	Amplifier gain

K_{B1}, K_{B3}	Virtual force gains used by Borenstein and Koren
K_{BT}	Virtual torque gain used by Borenstein and Koren
K_c	Foam stiffness
K_{cont}	Control stiffness of a joint
$K_{d,dd}$	Derivative gain for the direct drive manipulator
$K_{d,pnm}$	Derivative gain for the pneumatic cylinder
$K_{D,mr,i}$	Derivative gain for the i^{th} wheel controller of the holonomic robot
$K_{D,nh,i}$	Derivative gain for the i^{th} wheel controller of the nonholonomic robot
K_{ffa}, K_{ffv}	Feedforward gains for the pneumatic cylinder
K_{G1}, K_{G2}, K_{G3}	Virtual force gains used by Ge and Cui
K_h	Stiffness of the human neck
K_J	Stiffness of a revolute joint
$K_{J\,i}$	Stiffness of the i^{th} joint
K_{K3}	Repulsive virtual force gain used by Khatib
K_L	Time varying parameter used in the stabilizing force function
K_{mech}	Mechanical stiffness of a joint
$K_{mr,i}$	Product of the motor torque constant, the amplifier gain and the gear ratio for the nonholonomic robot
$K_{nh,i}$	Product of the motor torque constant, the amplifier gain and the gear ratio for the nonholonomic robot
$K_{p,dd}$	Proportional gain for the direct drive manipulator

$K_{p,pnm}$	Proportional gain for the pneumatic cylinder
$K_{P,mr,i}$	Proportional gain for the i^{th} wheel controller of the holonomic robot
$K_{P,nh,i}$	Proportional gain for the i^{th} wheel controller of the nonholonomic robot
K_{ri}	Effective manipulator stiffness for the i^{th} impact scenario
$K_{t,dd}$	Summed motor torque constant
$K_{t,dd,i}$	Torque constant of the i^{th} motor of the direct drive manipulator
K_x, K_y	Time varying positive gains used in the nonholonomic robot controller
K_Λ	Parameter used in the repulsive force function
$\dot{K}_\Lambda$	Time derivative of K_Λ
K_ϕ	Time varying positive gain used in the nonholonomic robot controller
l_i	Distance between the impact point and the i^{th} joint
$l_{cam,ds}$	Camera calibration parameter
l_{tube}	Length of the tube
l_w	Distance between the wheels and the centre of the holonomic robot
$l_{w,nh}$	Distance between Wheels 1 and 2
L_{energy}	Energy wasted by the motors used to drive the mobile robot
L^*_{energy}	Nominal wasted energy
$\mathbf{L}_{mr,i}$	Luenburger observer gain vector

m_{dd}	Moment of inertia of the direct drive manipulator
m_{disk}	Mass of the steel disk used in direct drive manipulator
m_i	Mass of the i^{th} link
m_m	Total moving mass of the apparatus used to measure the foam parameters
m_{mr}	Mass of the mobile robot
M_t	Mass of a human torso
m_{tube}	Mass of the aluminum tube used in direct drive manipulator
$\mathbf{M}$	Mass matrix of a manipulator
$\mathbf{M}_{23}$	Mass matrix related to joints 2 and 3
M_h	Mass of a human head
M_{ij}	Element from the i^{th} row and j^{th} column of $\mathbf{M}$ matrix
M_r	Effective mass of a manipulator
M_{ri}	Effective manipulator mass for the i^{th} impact scenarios
n	Index
N	Number of obstacles
N_1, N_2, N_3	Parameters for normalizing the total performance criterion
$p_{1,i}$, $p_{2,i}$	Two desired poles for the Luenberger observer of the i^{th} wheel
$p_{d,pnm}$, $\dot{p}_{d,pnm}$	Desired position and velocity of the pneumatic cylinder piston
$\ddot{p}_{d,pnm}$	Desired acceleration of the pneumatic cylinder piston
p_{pnm}, $\dot{p}_{pnm}$	Actual position and velocity of the pneumatic cylinder piston
P_c	Impact point

$\mathbf{P}_g$	Goal position vector
P_{gx}, P_{gy}	X and Y components of the goal position
$\mathbf{P}_h$	Human position vector
$\mathbf{P}_{h0}$	Vector of the initial human position
$\mathbf{P}_{h1}$	Point that the human reached in the collinear condition
P_{hx}, P_{hy}	X and Y components of the human position
P_{hx0}, P_{hy0}	X and Y components of initial human position
$\mathbf{P}_r$	Mobile robot position vector
$\mathbf{P}_{r0}$	Vector of the initial robot position
$\mathbf{P}_{r1}$	Point reached by the robot in the collinear condition
P_{rx}, P_{ry}	X and Y components of the robot position
$\dot{P}_{rx}$, $\dot{P}_{ry}$	X and Y components of the robot velocity
P_{rx0}, P_{ry0}	X and Y components of the initial robot position
$\dot{P}_{rx,d}$, $\dot{P}_{ry,d}$	X and Y components of the reference holonomic robot's velocity
$\mathbf{P}_{r,d}$ $\dot{\mathbf{P}}_{r,d}$	Position and velocity of the reference holonomic robot used in the nonholonomic robot control system
q_1, q_2, q_3, q_4	Coefficients related to the dynamic parameters of the manipulator foam covering and the coupling from the head to torso
$\mathbf{Q}_i$	Positive definite matrix for the i^{th} Lyapunov function
r_2	Radius of the active region of the obstacle for the holonomic robot
$r_{2,cir}$	Radius of the active region of the stationary circular obstacle for the holonomic robot

$r_{2,cir,nh}$	Radius of the active region of the stationary circular obstacle for the nonholonomic robot
$r_{2,mob}$	Radius of the active region of the Obstacle Robot for the nonholonomic robot
$r_{2,nh}$	Active region size of an obstacle for the nonholonomic robot
$\hat{r}_{2,rect}$	Distance for the robot to avoid the stationary rectangular obstacle
$r_{2,rect}$	Size of the active region of the stationary rectangular obstacle for the holonomic robot
$r_{2,rect,nh}$	Size of the active region of the stationary rectangular obstacle for the nonholonomic robot
r_3	Radius of the critical region of the obstacle
$\hat{r}_3$	Stopping distance for avoiding a human
$r_{3,cir}$	Radius of the critical region of the stationary circular obstacle
$r_{3,mob}$	Radius of the critical region of the Obstacle Robot
$r_{3,rect}$	Size of the critical region of the stationary rectangular obstacle for the holonomic robot
$\hat{r}_{3,rect}$	Stopping distance for avoiding a rectangular obstacle
$r_{3,rect,nh}$	Size of the critical region of the stationary rectangular obstacle for the nonholonomic robot
r_{disk}	Radius of the steel disk used in direct drive manipulator
r_w	Radius of each wheel of the holonomic robot
$r_{w,nh}$	Radius of each wheel of the nonholonomic robot

$\mathbf{R}_a$	Diagonal matrix of the winding resistances of mobile robot motors
R_a	Winding resistance of a motor
$\mathbf{R}_{cam}$	3-by-3 rotation matrix for the transformation between the world and camera coordinates,
s	Laplace transform variable
$s_{pix,x}$, $s_{pix,y}$	Fixed parameters of the camera dependent only on the size of the imaging sensor and the image resolution
S_b	Relative sensitivity of H_{mean} to the VFF gain ratio, b
S_{cam}	Scale factor to account for the pixel size
S_{K_i}	Relative sensitivity of H_{mean} to the VFF gain, K_i
t	Time value
T_{arrive}	Time when a mobile robot arrives at its goal
T^*_{arrive}	Nominal arrival time, defined as the time consumed by the mobile robot to reach its goal without obstacles
$\mathbf{T}_{cam}$	Translation components for the transformation between the world and camera coordinates
T_{rem}	Remaining foam thickness where the foam stiffness is nonlinear
T_s	Sampling period of the navigation system
T_u	Uncompressed foam thickness
T_x, T_y, T_z	Translation components of $\mathbf{T}_{cam}$
$\mathbf{u}$	Vector of motor control commands
u_{dd}	Control command for the motors of the direct drive manipulator

$u_{mr,i}$	Control command for the i^{th} motor of the holonomic robot
$u_{nh,i}$	Control command for the i^{th} motor of the nonholonomic robot
u_{pnm}	Control command for the valve of the pneumatic cylinder
$\mathbf{u}_E$	Unit vector in the direction of $\mathbf{E}$
$\mathbf{u}_L$	Unit vector in the direction of the stabilizing force
$\mathbf{u}_{mr}$	Vector of control commands for the motors of the mobile robot
$u_n - u_t$	Coordinate axes at the impact point
$\mathbf{u}_\Lambda$	Unit vector in the direction of $\mathbf{D}$
$\mathbf{u}_\psi$	Unit vector in the direction of the detour force direction
$\mathbf{U}_{dd}$	Array of the u_{dd} of the direct drive manipulator
$\mathbf{U}_{mr,i}$	Array of the $u_{mr,i}$ of the holonomic robot
$\mathbf{U}_{nh,i}$	Array of the $u_{nh,i}$ of the nonholonomic robot
v_c	Impact velocity
v_h	Linear human head velocity
$v_h(0)$	Initial human head velocity
v_i	Linear velocity of the i^{th} wheel
$\mathbf{v}_L$	Vector used in the stabilizing force function
$\mathbf{v}_{L,N}$	$\mathbf{v}_L$ when the robot shares its work area with N obstacles
$\mathbf{v}_{L,\Delta t}$	$\mathbf{v}_L$ value after an infinitesimal time interval
v_n	Component of the human head velocity in the normal direction
v_{P_c}	Velocity of the impact point, P_c
v_r	Linear velocity of the manipulator

$v_r(0)$	Initial velocity of the manipulator
v_{ri}	Linear velocity of the manipulator for the i^{th} impact scenario
v_t	Component of the human head velocity in the tangential direction
V	Piecewise Lyapunov function candidate
$\bar{V}_h$	Maximum human walking velocity
$\mathbf{V}_h$	Human velocity vector
V_i	i^{th} function in the piecewise Lyapunov function candidate
$\bar{V}_{mob}$	Maximum velocity of the Obstacle Robot
$\mathbf{V}_{mr}$	Velocity vector of the mobile robot
V_{mr}	Mobile robot velocity magnitude
$\bar{V}_{mr}$	Mobile robot maximum linear velocity
$V_{mr,n}$, $V_{mr,t}$	Normal and tangential components of the robot velocity vector
V_o	Obstacle velocity magnitude
w_1, w_2, w_3	Weights used to calculate H
w_{tube}	Width of the tube
W	$W = \|\mathbf{W}\|_2$
$\mathbf{W}$	Vector from the obstacle to the goal
$\dot{x}$	Component of the nonholonomic robot linear velocity along the X-axis
x_d, $\dot{x}_d$	Desired linear position and velocity of the impact point
X_{ds}, Y_{ds}	Distorted image plane coordinates
x_e	Error between the reference holonomic robot and nonholonomic

	robot along the X-axis
x_h, $\dot{x}_h$, $\ddot{x}_h$	Position, velocity and acceleration of the human head
x_{mr}, $\dot{x}_{mr}$	Robot position, velocity in the X direction of the robot coordinate frame
x_r, $\dot{x}_r$, $\ddot{x}_r$	Actual linear position, velocity and acceleration of the impact point
x_t, $\dot{x}_t$	Position and velocity of the human torso
X, Y, Z	World coordinates of the colour patch positions
X_{cam}, Y_{cam}, Z_{cam}	Camera coordinates
$X_h(s)$	Linear position of the head in the frequency domain
$\mathbf{X}_i$	State vector of the VFF for the i^{th} Lyapunov function
X_{im}, Y_{im}	Pixel coordinates of the colour patch
$X_r(s)$	Linear position of the manipulator in the frequency domain
X_u, Y_u	Undistorted image plane coordinates
y_0	Position where the cylindrical part first contacts the foam
y_c, $\dot{y}_c$, $\ddot{y}_c$	Displacement, velocity and acceleration of the apparatus used for measuring the foam damping
y_e	Error between the reference holonomic robot and nonholonomic robot along the Y-axis
y_{fall}	Initial height of the apparatus in the foam damping experiments
y_{mr}, $\dot{y}_{mr}$	Robot position and velocity in the Y direction of the robot coordinate frame
$\mathbf{Y}_{dd}$	Array of the A_{dd} of the direct drive manipulator

$\mathbf{Y}_{mr,i}$	Array of the $A_{mr,i}$ of the holonomic robot
$\mathbf{Y}_{nh,i}$	Array of the $A_{nh,i}$ of the nonholonomic robot
z_i	Angle of the wheel shaft
α	The angle of **W**
β	The angle of **D**
α_{rect}	The angle of **W** for the rectangular obstacle
β_{rect}	The angle of **D** for the rectangular obstacle
γ_c	Angle between the head initial velocity and the normal impact direction
γ_{link}	Angle between the centreline of the manipulator and the line from the joint to the impact point
γ_{mr}	Angle between the mobile robot velocity and the rectangular obstacle's exterior contour
γ_r	Angle between the manipulator velocity and the normal impact direction
Γ_{dd}	Parameter related to static friction of the direct drive manipulator
δ_i	Error between the desired and actual linear positions of the impact point for the i^{th} impact scenario
$\bar{\delta}_{hl}$	Maximum moving distance of the human before the mobile robot stops
$\bar{\delta}_{hx2}$	Maximum moving distance of the human along the X-axis in the collinear condition

$\bar{\delta}_{mobile}$	Maximum moving distance of the Obstacle Robot before the mobile robot stops
$\bar{\delta}_{r1}$	Stopping distance of the mobile robot from $\bar{V}_{mr}$
δ_{rx2}	Moving distance of the mobile robot along the X-axis in the collinear condition when the obstacle is a human
$\bar{\delta}_{rx2}$	Maximum moving distance of the mobile robot along the X-axis in the collinear condition when the obstacle is a human
$\bar{\delta}_{rx2,circular}$	Maximum moving distance of the mobile robot along the X-axis in the collinear condition when the obstacle is the circular obstacle
$\bar{\delta}_{rx2,mobile}$	Maximum moving distance of the mobile robot along the X-axis in the collinear condition when the obstacle is the Obstacle Robot
$\bar{\delta}_{rx2,rect}$	Maximum moving distance of the mobile robot along the X-axis when the obstacle is the rectangular obstacle
δ_{ry2}	Moving distance of the mobile robot along the Y-axis in the collinear condition when the obstacle is a human
Δt	Time interval
Δt_1	Time interval required for stopping the mobile robot from $\bar{V}_{mr}$
Δt_2	Time interval for computing the size of the human's active region
$\Delta t_{2,circular}$	Time interval for computing the size of the stationary circular obstacle's active region
$\Delta t_{2,mob}$	Time interval for computing the size of the Obstacle Robot's active region

$\Delta t_{2,rect}$	Time interval for computing the size of the rectangular obstacle's active region
Δt_{HIC}	Time interval for calculating HIC
$\Delta t_{HIC,\max}$	Maximum time interval for HIC
Δt_{imp}	Impact period
ε	Compressed depth of the foam
ε_0	Initial compressed depth of the foam caused by gravity when measuring the foam stiffness
ε_c	Foam compressed depth minus the initial compressed depth
$\varepsilon_{csf}(s)$	The compressed depth in the constrained impact in the frequency domain
$\varepsilon_{\max}$	Maximum foam compressed depth
$\varepsilon(s)$	Foam compressed depth in the frequency domain
$\boldsymbol{\theta}$, $\ddot{\boldsymbol{\theta}}$	Vector of joint positions, and vector of joint accelerations
$\tilde{\theta}$	Error between θ_J and θ_d
$\tilde{\theta}_i$	Error between the desired and actual angular positions of the i^{th} joint
$\dot{\tilde{\theta}}_i$	Error between the desired and actual angular velocity of the i^{th} joint
Θ_{dd}	Parameter related to the acceleration of the direct drive manipulator
θ_{enc}	Sensed angular position of the joint

θ_d	Desired position of the joint
$\theta_{di}, \dot{\theta}_{di}, \ddot{\theta}_{di}$	Desired angular position, velocity and acceleration of the i^{th} joint
$\theta_{d,dd}$	Desired angular position of the direct drive manipulator
$\dot{\theta}_{d,dd}$, $\ddot{\theta}_{d,dd}$	Time derivation and second time derivation of $\theta_{d,dd}$
$\theta_{i,d}$, $\dot{\theta}_{i,d}$, $\ddot{\theta}_{i,d}$	Desired angular position, velocity and acceleration of the i^{th} motor
θ_J	Actual angular position of the joint
$\theta_{Ji}, \dot{\theta}_{Ji}, \ddot{\theta}_{Ji}$	Actual angular position, velocity and acceleration of the i^{th} joint
$\theta_{J1,init}$	Value of θ_{J1} when the head and manipulator make first contact
κ_{dd}	Parameter related to the viscous friction and moment of inertia of direct drive manipulator
$\kappa_{mr,i}$	Parameter related to the viscous friction and moment of inertia of the i^{th} wheel in the holonomic robot
$\kappa_{nh,i}$	Parameter related to the viscous friction and moment of inertia of the i^{th} wheel of the nonholonomic robot
λ	Threshold defining the linear range of the foam stiffness
Λ, $\dot{\Lambda}^*$	Parameters used in the repulsive force function
Λ_i, $\dot{\Lambda}_i^*$	Parameters used in the repulsive force function for the i^{th} obstacle
Λ_{rect}, $\dot{\Lambda}_{rect}^*$	Parameters used in the repulsive force function for the rectangular obstacle
ξ_c	Damping ratio of the foam
ξ_h	Damping ratio of the human neck
ξ_{nh}	Desired damping coefficient in the nonholonomic robot controller

$\rho_{circular}$	Radius of the stationary circular obstacle model
ρ_h	Radius of the human model
ρ_{mobile}	Radius of the Obstacle Robot model
ρ_r	Radius of the model of the mobile robot
$\boldsymbol{\tau}$	Vector of joint torques
τ_{dd}	Input torque of the direct drive manipulator
τ_{imp}	Torque acting on the joint caused by the impact force
$\tau_J(s)$	Joint torque in the frequency domain
τ_{Ji}	Torque of the i^{th} joint
$\tau_m(s)$	Motor torque in the frequency domain
$\tau_{mr,i}$	Static friction torque of the i^{th} motor and gear for the holonomic robot
$\tau_{nh,i}$	Static friction torque of the i^{th} motor and gear for the nonholonomic robot
$\tau_{st,dd}$	Summed motor static friction torque
υ	Linear velocity of the nonholonomic robot
υ_d	Linear velocity magnitude of the reference holonomic robot
Υ_{BT}	Difference between the angle of $\mathbf{F}_V$ and current heading angle of the nonholonomic robot
ϕ	Heading angle of the mobile robot
$\dot{\phi}$	Angular velocity of the mobile robot
ϕ_d	Heading angle of the reference holonomic robot

Φ	Angle α minus angle β
χ_{tube}	Density of the tube used in the direct drive manipulator
ψ , $\dot{\psi}^*$	Parameters used in the detour force function
ψ_{rect} , $\dot{\psi}^*_{rect}$	Parameters used in the detour force function for the rectangular obstacle
ψ_i , $\dot{\psi}^*_i$	Parameters used in the detour force function for the i^{th} obstacle
Ω	Time varying variable used in the stability analysis
Ω_i	Variable Ω for i^{th} Lyapunov function
ω	Angular velocity of the nonholonomic robot
ω_d	Angular velocity of the reference holonomic robot
ω_{nh}	Parameter used in the nonholonomic robot controller

Chapter 1

Introduction

1.1 The three main types of robots

Three main types of robots are being used in manufacturing, service and research applications today, namely robotic manipulators (see Fig. 1.1a), mobile robots (see Fig. 1.1b) and mobile manipulators (see Fig. 1.1c). A robotic manipulator (also called *manipulator*) is similar to a crane. It has joints and links connected in series to a stationary base. Those joints are driven by actuators, such as electric motors and pneumatic cylinders. The links are nearly rigid beams used to connect the joints. With a stationary base and finite length links, the manipulator can only move within a limited space, termed its *workspace*. The manipulator has a tool (*i.e.* termed an *end-effector*) at its distal end to perform a variety of tasks, such as picking up parts and assembling them. A manipulator requires six or more degrees of freedom (DOF) to be able to arbitrarily position and orient its end-effector. For a typical manipulator the number of DOF equals the number of driven joints. In Fig. 1.1a, a manipulator with the articulated configuration is presented as an example of manipulators. It has six revolute joints (*i.e.* each joint produces a rotary motion), and six DOF. The main purpose of the first three revolute joints is to move the end-effector to its desired position. The other three revolute joints (not drawn in Fig. 1.1a) located inside the Euler wrist are primarily used to move the end-effector to its desired orientation.

In contrast to a manipulator with a stationary base, a mobile robot is capable of moving itself around in its work area. Due to its mobility, it is suitable to be used in applications such as delivery, cleaning and security. Currently, most mobile robots are wheel-driven. A typical design is shown in Fig. 1.1b. This mobile robot has a passive caster or omnidirectional wheel (Wheel 3) and two driven wheels (Wheel 1 and Wheel 2) that are normally driven by electric motors. The two driven wheels provide the robot with

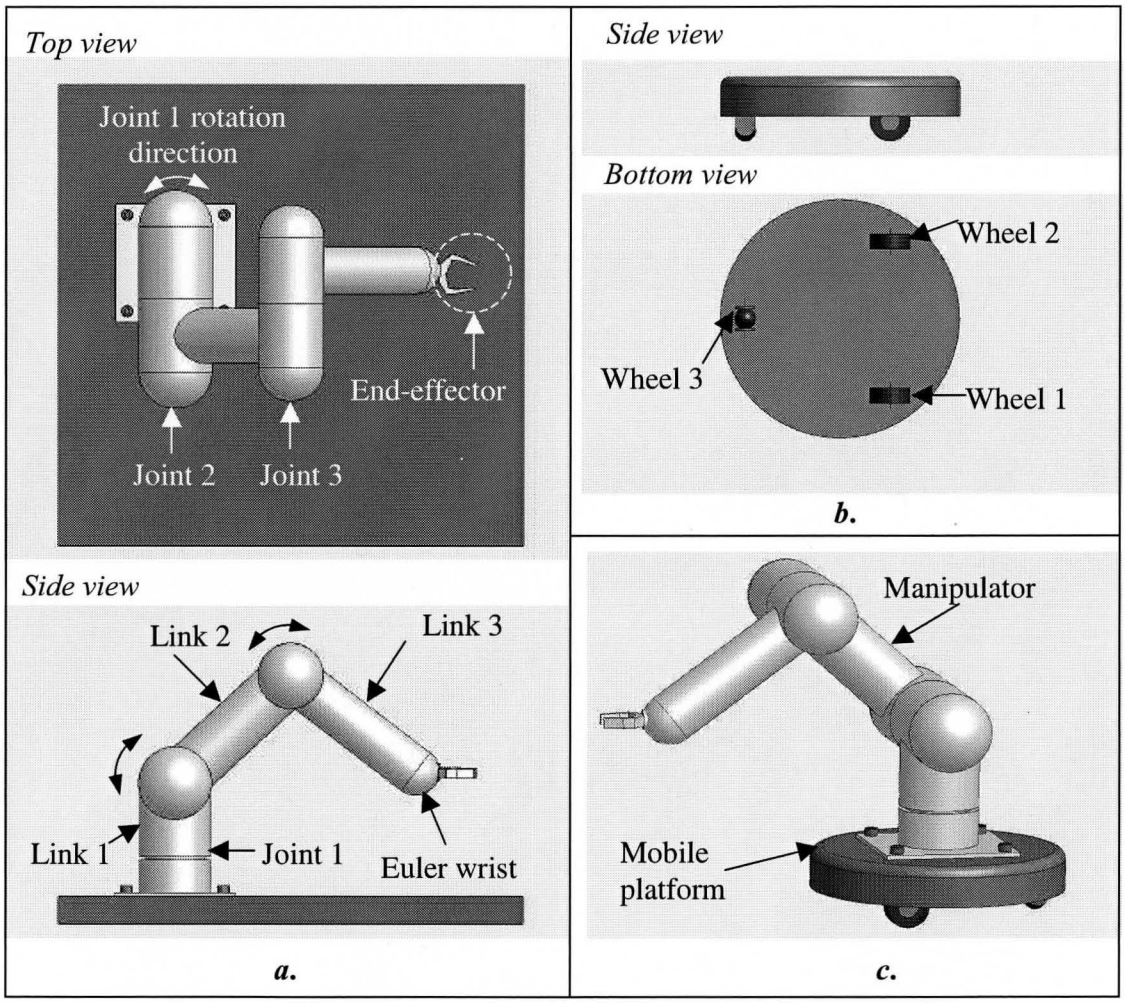


Fig. 1.1. The three main types of robots: ***a***. articulated robotic manipulator; ***b***. mobile robot; and ***c***. mobile manipulator

two controllable DOF: moving forwards or backwards, and turning left or right. By suitably controlling those two DOF, the robot is able to reach any position in its work area with any orientation.

By mounting a manipulator on a mobile platform (a mobile robot), a mobile manipulator (Fig. 1.1c) is created. It combines the advantage of mobility from the mobile platform with dexterous manipulation capability of the manipulator. Mobile manipulators therefore have many potential uses in manufacturing, service and security applications, such as going into the battlefield and disabling bombs.

1.2 The challenges of human-robot safety

To be effective, future robots will be required to have much more intelligence and execute more complex tasks than before. These tasks could require the robots to cooperate with humans physically, such as helping elderly people to get up from bed. Human-robot cooperation has become a topic of increasing importance in robotics research. The essential requirement of a robot designed for human-robot cooperation is that it must guarantee the safety of humans. On the other hand, due to the motions of humans, the human cooperation often creates an ever-changing environment that requires more intelligence from the robot.

Each of the three types of robots has their own safety issues when cooperating with humans. For a manipulator in a manufacturing application, humans have to be in close proximity to a manipulator for tasks such as training, programming and maintenance. Accidental human-manipulator physical contact can happen. The contacts may cause severe human injuries. Jiang and Gainer (1987) surveyed 32 severe human-

manipulator accidents in industrial environments. Those accidents involved the robot operators, maintenance workers and programmers; and were caused by either human error, improper workplace design or robot design. With service applications of manipulators, such as assisting elderly people, accidental and even intentional human-manipulator contact is more likely to occur. Therefore, the safety issue with manipulators is preventing human injuries when human-manipulator contacts happen.

In any application where a mobile robot shares its work area with humans, it could collide with the humans or other objects (termed *obstacles* in the robotics literature) that block its path. The collision may injure the humans, damage the obstacles and damage the robot itself. Therefore, the safety issue with a mobile robot is navigating itself to avoid the collisions with humans and obstacles and reach its goal (termed *navigation*). As an example, when a mobile robot carries a part and delivers it in a factory, humans, such as workers could be confronted. The robot has to avoid those humans and deliver the part to a specified position (termed the *goal* of the robot). A mobile manipulator has to address both safety issues since it is the combination of a mobile robot and a manipulator.

The term *human-friendly robot* has been used in the robotics literature since the 1990s, *e.g.* Kosuge, Yoshida and Fukuda (1993). For a robot to be human-friendly, human safety must always be its highest priority[1]. In other words, the humans should be allowed to conduct their own work without worrying about the robots surrounding them. The robot must monitor the motions of the humans, and respond to those motions to

[1] Other aspects of human-friendliness, such as emotional intelligence, are not studied in this thesis.

ensure the safety of the humans while performing its own task. There could also be other objects within the work area. The robot must also be prevented from damaging those objects and itself.

1.3. Existing solutions for human-robot safety

One of the traditional solutions for human-robot safety is to limit the access of the humans to the robots. This can be achieved by using a physical fence and/or sensor curtain to enclose the robot and its workspace. It is clear that this solution is only suitable for manipulators since they have a stationary base. This fence prevents the human from coming into close proximity with the manipulator when the manipulator is performing its task. In Fig.1.2, a polishing manipulator is shown enclosed with a fence. There is a gate in the fence to allow humans to come in for conducting maintenance and programming. At that time the manipulator must be stopped or move at a slow speed in accordance with safety standards (ISO10218-1-2006) to protect the humans. However, human injuries can still occur. This is because the tasks are performed near the robot within the fence. Unintentional impact can happen and cause injuries. As an example, the maintenance worker could collide with a link of the manipulator when standing up from a squatting pose.

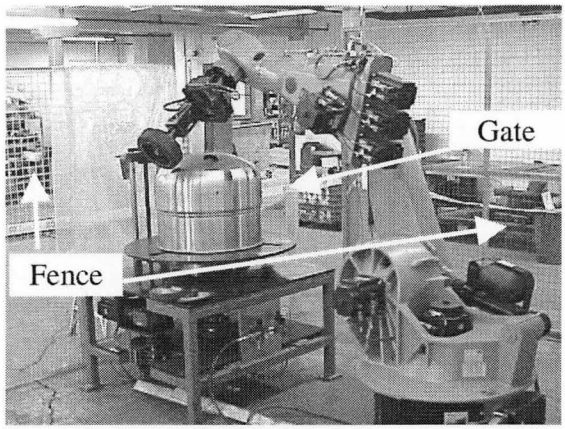


Fig. 1.2 A polishing manipulator (from Pushcorp Inc.) enclosed with a fence for human safety.

Another solution is to limit the potential and kinetic energy of the robot, *e.g.* by limiting its mass and velocity. As a service robot example, a Roomba® robotic vacuum (a mobile robot similar to the one shown in Fig. 1.1c) has only a 3 kg mass and its velocity is smaller than 0.5 m/s. It also possesses a collision sensor (a bumper equipped with a switch) to detect obstacles. When the bumper is hit (*e.g.* by a person's foot), the robot will be commanded to stop. This method ensures human safety, but severely limits the capability of the robot.

Similarly, the newest robot safety standard, ISO10218-1-2006 provides the requirements and guidelines to eliminate or reduce the risks associated with human-robot cooperation for industrial robots. With this standard, the robot may not initiate greater static force than 150 N or 80 W of impact power against a person. In this standard, the *safety speed* is also defined. It specified as the end-effector's speed when the manipulator is fully extended must be less than 0.25 m/s. The standard states that this limit was chosen such that the human has enough time to avoid an impact. In other words, the standard expects the human to always watch the movement of the manipulator; otherwise

the human will not have enough time to avoid an impact or stop the manipulator. This is not a reasonable approach to human-robot cooperation since the human will not be able to focus on other tasks.

The technique of machine vision is also being used. For example, Pilz Corp. introduced SafetyEYE^{TM}, a multi-camera system that provides a 3D detection field to monitor the robot workspace, in 2007. Fig. 1.3 illustrates the concept of a fenceless manipulator with SafetyEYE^{TM}. With this concept, multiply safety zones are provided with respect to the distance to the robot workspace. According to Ponticel (2007), when the human comes into the *warning zone* of SafetyEYE^{TM}, the robot should slow down and keep performing its task. Then if the human comes into the *protected zone*, the robot should be commanded to stop and/or trigger an alarm. This method is similar to the traditional method with fences and is only suitable for manipulators. Its advantage is that

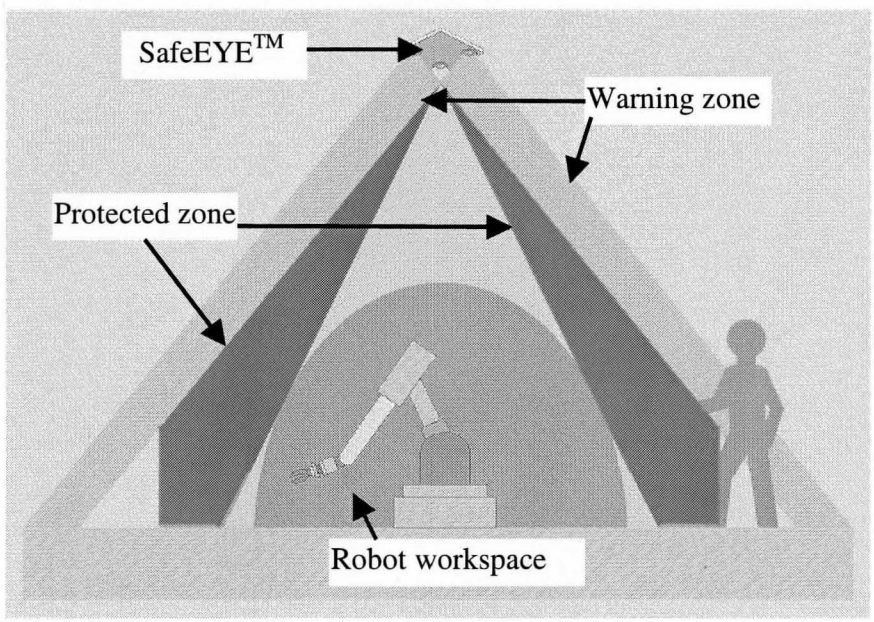


Fig. 1.3 The concept of the fenceless robot with SafetyEYE^{TM} from Ponticel (2007)

it is easy to modify the numbers of zones, their sizes and the manipulator actions in those zones.

1.4 Research objectives

Conventional robotic systems and traditional safety solutions are too inflexible to apply in situations and applications where a human-friendly robot will play a major role (Lund 2004). Therefore, new solutions are needed. The goal of this research is to advance the state of the art in the design and control of human-friendly robots. The research will provide solutions for two of the safety problems discussed in Section 1.3. The first problem is severe human injuries can be caused when a human-manipulator impact happens. The first research objective is to develop a reliable method to reduce the severity of the impact and enhance human safety. Our method will utilize foam covering for the reasons explained in the conclusion section of Chapter 2. The second problem is mobile robot navigation. The second research objective is to develop a new navigation algorithm for navigating within manufacturing and service environments in the presence of humans more effectively than existing solutions.

1.5. Thesis layout

This thesis is organized as follows: In Chapter 2, the state of the art literature on impact force modelling and reduction; and on mobile robot navigation are reviewed. In Chapter 3, an impact dynamic model is developed that includes the coupled dynamics of the robotic manipulator, the human head-neck-torso system and the foam covering. A

model-based method for determining the maximum and minimum bounds of the foam stiffness and damping is presented. In Chapter 4, experiments are performed to verify the impact model and the foam covering design process. A custom built direct-drive robotic manipulator and a Puma 560 manipulator are used in the experiments. In Chapter 5, a novel collision avoidance algorithm suitable for avoiding dynamic obstacles including humans is proposed. Its stability is proven using Lyapunov's second method. In Chapter 6, navigation simulations and experiments with a holonomic robot are presented and compared with two important conventional algorithms. A moving human, a stationary rectangular obstacle and a stationary circular obstacle are employed in different configurations to test the navigation algorithm. Simulations and experiments with a nonholonomic robot are performed and compared with simulation results for the holonomic robot in Chapter 7. In Chapter 8, the conclusions of this research are drawn.

Chapter 2

Literature Review

2.1 Impact force modelling and reduction

2.1.1 Introduction

The aim of impact force reduction is to reduce the impact force from incidental human-robot collisions to a level that prevents serious human injuries. The conventional methods can be classified into the four categories: active control methods, actuation methods, passive mechanisms and compliant coverings. Impact force models are sometimes used to make these methods more effective. Various criteria have been designed for preventing human injuries due to impact. The relevant literature on these topics will be reviewed in the proceeding subsections.

2.1.2 Active control methods

Previous researchers have utilized active force/torque control methods to reduce the impact force. Heinzmann and Zelinsky (2003) developed an *impact potential* control scheme that limited the impact force of the robot by restricting the torque commands. This has the disadvantage of limiting the payload capacity of their manipulator to roughly 1 kg. de Luca and Mattone (2003) introduced a collision detection/impact force control system for a two DOF planar manipulator employing a hybrid position/force control algorithm. Since the impact force influences the motion of the manipulator, their algorithm detects the impact by comparing the desired positions and the actual positions

of the manipulator. The disadvantage of this system is it requires switching of the control structure when the impact is detected. This can make the control system unstable. Jeong and Takahashi (2009) presented a control algorithm for stopping the manipulator with maximum deceleration to reduce the impact force. A high-bandwidth air pressure collision detection sensor is used to detect the impact, activate the controller, and provide a shock absorbing function.

2.1.3 Actuation methods

Other researchers have combined sophisticated actuation approaches with active control methods. Mechanical impedance is the ratio of the output force to the input displacement as a function of frequency. It is well known that a lower mechanical impedance will reduce the impact force. Pratt and Williamson (1995) proposed the *series elastic actuator* (SEA) in 1995. The SEA adds a spring between the joint and the link of a manipulator to decrease the effective impedance, as shown in Fig. 2.1. Due to the reduced joint impedance, the method decreases the impact force but leads to lower actuator bandwidth and decreased trajectory-tracking ability. Zollo *et al.* (2003) implemented three impedance control schemes on a cable-actuated manipulator and compared them with standard proportional plus integral plus derivative (PID) control. Their experimental results demonstrate that impedance control can reduce the impact

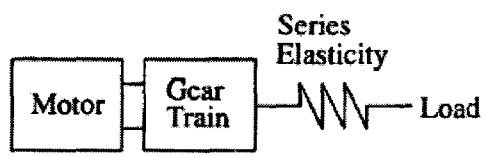


Fig. 2.1 Serial Elastic Actuator (SEA) from Pratt and Williamson (1995)

force to two times smaller than the force with PID control. Zinn *et al.* (2004) developed a sophisticated hybrid actuation method combining the SEA with a high bandwidth electric motor and a PID control scheme to increase the bandwidth limit of SEA while maintaining low impedance. This method improves the tracking performance of the manipulator but increases the manipulator's expense significantly. Laffranchi, Tsagaraskis and Caldwell (2009) extended the SEA with an energy regulation control method to further reduce the impact force. The reference trajectory is modified online to limit the energy of the manipulator to a safe value. Experiments with a very lightweight manipulator (driven mass is 0.41 kg) were conducted to verify this control method.

2.1.4 Passive mechanisms

Passive mechanisms have also been proposed for reducing the impact force. Lim and Tanie (2000) presented the design and testing of a viscoelastic trunk for a mobile manipulator. This viscoelastic trunk reduces the stiffness of the manipulator but is only effective for horizontal impacts (*i.e.* the impact force vector lies in the horizontal plane). Park *et al.* (2007) designed a complex *safety link* mechanism with a shock-absorbing device. When the impact happens, the shock-absorbing device is triggered to deform at the middle of the mechanism and bend the link backwards to reduce the impact force.

2.1.5 Compliant coverings and impact force models

Compliant coverings, such as elastomeric foam coverings have been used for human-friendly robot designs and also for head protection in helmets. Suita *et al.* (1995) proposed an approach for selecting the elastic modulus and viscosity coefficient of a foam covering to ensure the human-robot impact force is smaller than the pain tolerance

of the human body (which they specified as 50 *N*). The impact force was computed by assuming the human and robot velocities were constant throughout the impact. The mass and stiffness values of both the human body and the robotic manipulator were not considered. This makes their simulation results unrealistic for most cases. They mentioned the importance of knowing the minimum foam thickness required to avoid it becoming fully compressed but did not provide a method for finding it. No experiments were presented. Yamada *et al.* (1997) extended this approach by controlling the manipulator to reduce its velocity during the impact. Experiments were conducted and showed the impact force after 0.1 s can be significantly reduced by the combination of their control system and foam covering.

Ikuta and Nokata (2003) introduced the concept of a "danger index" to evaluate the safety of the impact. This index equals the ratio of the actual impact force to the value of a predefined impact force threshold. They presented a simple human-robot impact dynamics model. Their model assumes that the human and robot are free point masses and that the manipulator is rigid (*i.e.* its joints and links are infinite stiff) and is covered with a compliant covering that can be modeled as a spring and damper. Using this model, they concluded that a smaller robot inertia, stiffness and velocity will produce a smaller impact force.

Bicchi and Tonietti (2004) proposed a more realistic human-robot impact dynamic model, as shown in Fig. 2.2. The robotic manipulator was modeled as a single degree of freedom (DOF) prismatic joint actuating a mass, representing the driven mass of the joint. This mass was coupled by a spring and damper (representing the stiffness

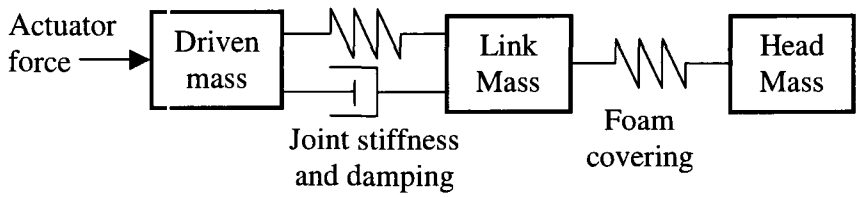


Fig. 2.2. The impact dynamics model from Bicchi and Tonietti (2004).

and damping of the joint) to a second mass, representing the link. The human's head was modeled as a point mass, and the compliant covering was modeled as a compression spring. The human head was assumed to be a free mass during the impact. This is unrealistic since the head is not actually a free mass; it is constrained by the neck and the torso of the human. Their focus was on the use of a variable stiffness transmission to reduce impact force. They did not study compliant coverings in any detail and they did not present any experimental results.

Oberer and Schraft (2007) presented impact simulation results from a finite element model. A SMART NS 16 robotic manipulator from Comau Robotics and a dummy representing the human were modeled. In the assumptions of this model, the human was covered with rubber and the manipulator was rigid. No results of impact experiments were published. Their simulations showed that the maximum head acceleration during the impact is nearly proportional to the impact velocity.

Vermeulen and Wisse (2007) established a human-robot impact model by assuming the robotic manipulator and the human head are both free masses, and the human head is stationary before the impact. An elastomeric covering with 5 kN/m stiffness was used to cover the surface of the manipulator. With this model, the relationship between the manipulator's velocity and moving mass and the impact force

was investigated. They did not study the force-reduction effects of elastomeric coverings. No experimental results were presented.

Haddadin, Albu-Schäffer and Hirzinger (2007 and 2009) performed experiments to explore the severity of human-manipulator impact. Different robotic manipulators (LWRIII from Germany Aerospace Center, KR6 and KR500 from Kuka Robotics Corp.) were moved at various velocities to impact the head of a sitting dummy and the chest of a sitting human volunteer. The experimental results demonstrated that the impact velocity is a key factor in causing human injuries. The impact force also increased with increasing effective masses of the robotic manipulators. Haddadin *et al.* (2008) introduced an impact model for a robotic manipulator colliding with a human head. In this model, the human head is represented as a mass-spring system; the stiffness of the maxilla bone is used as the stiffness and the head mass is 4 kg. The compliant covering is modeled as a spring and the robotic manipulator is represented by a mass-spring-mass model to represent the coupling of joint mass, joint stiffness and link mass. Then the authors studied the relationship between the impact force and the foam elastic modulus and the relationship between the foam covering compressed depth and the elastic modulus. They showed a softer covering (with a smaller elastic modulus) helps reduce the impact force but causes a larger compressed depth. As an example, a foam covering with an elastic modulus of 200 kN/m^2 requires a thickness of 0.15 m to prevent fully being compressed. Haddadin *et al.* (2010) simulated a human-robotic manipulator impact with this impact model. It is concluded that changing the stiffness of manipulator joints has no obvious effect on the impact force, which was also indicated from their experiment results in Haddadin, Albu-

Schäffer and Hirzinger (2007).

Park and Song (2009) established another human-manipulator impact model. The human head, neck and torso are modeled as a mass-damper-spring system. The dynamics of a robotic manipulator is simplified as a single DOF planar manipulator, however the method to simplify a common multi-DOF manipulator to a single DOF manipulator is not presented in this paper. With this model, simulations were performed to show the effects of the manipulator's mass and the impact velocity on the impact force. No experiment results were published to verify this model.

Regarding head protection in helmets, Townsend and McCammond (1975) derived an analytical model for a helmet-foam-head-neck-torso system. The foam was modeled as a compression spring and damper. The helmet and the head were represented by point masses. The neck was modeled as a compression spring and a damper; and the torso was assumed to be static during the impact. This model is more realistic for head impacts than the conventional head-neck-torso models. Using their model, they demonstrated that helmet foam lining materials with lower stiffness helped to reduce the impact force and the head injuries. Mills and Gilchrust (2006) employed finite element analysis to model head impact with foam-lined helmets. This technique can provide

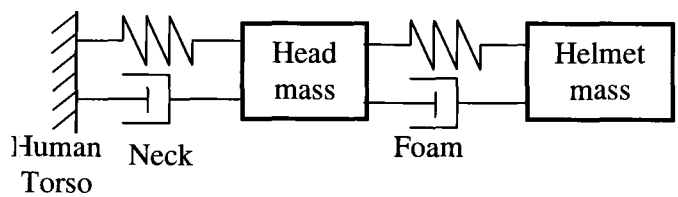


Fig. 2.3. The human-neck-torso model with foam liner and helmet from Townsend and McCammond (1975).

greater realism than lumped parameter models but does not provide useful equations for foam covering design.

2.1.6 Safety criteria for human safety

Human-injury criteria are required for developing an impact force reduction method (De Santis *et al.* 2008). We will review the criteria in this subsection. In terms of the Abbreviated Injury Scale (AIS) (AAAM, 1980 and 2008) which is an internationally established definition of injury severity, classifying it from 0 (no injury) to 6 (fatal). AIS gives no hint how to measure possible injury. This is provided by so called severity indices or injury criteria. The ISO-10218 standard (ISO Standard 10218-1, 2006) defines collaborative operation requirements for industrial robots. It states that one of the following conditions must always be fulfilled for allowing human-robot interaction: the maximum dynamic power < 80 W, or the maximum static force <150 N. According to Haddadin, Albu-Schäffer and Hirzinger (2008), these requirements are overly restrictive, too undifferentiated and strongly limit the performance of the robot.

Since the head is the most vulnerable part of the human body whereby head impacts can cause severe or possibly fatal injuries, the criteria for head injuries have been used in many existing researches. Haddadin, Albu-Schäffer and Hirzinger (2008) presented the fracture tolerances for every skull bone. The 660 *N* fracture force of the maxilla bone was the smallest and was chosen as their safety criterion. According to the EuroNCAP standard (EuroNCAP, 2004), a HIC value below 650 corresponds to AIS < 3, and only causes recoverable injuries. HIC is a head-acceleration-based criterion, and it was first introduced in automobile crash tests to evaluate human head injuries (Versace

1971). It is defined as:

$$HIC = \max\left(\frac{1}{\Delta t^{1.5}}\int_{\Delta t} a_h(t)^{2.5}\right) \qquad (2.1.1)$$
$$\Delta t \le \Delta t_{\max}$$

where $a_h(t)$ is the acceleration in g of the human head during the time interval Δt. The time interval Δt should be chosen so as to maximize the HIC subject to either $\Delta t_{\max} = 0.015$ s (Bicchi and Tonietti 2004; Haddadin, Albu-Schäffer and Hirzinger 2007) or $\Delta t_{\max} = 0.036$ s (Versace 1971; Haddadin, Albu-Schäffer and Hirzinger 2010). A smaller HIC implies higher safety. A value of HIC = 650 ($\Delta t_{\max} = 0.015$ s) corresponds to an average head acceleration of 72 g. According to EuroNCAP, a 0.003 s criterion was also proposed. This criterion requires the average of the resulting head acceleration during any 0.003 s interval of an impact to be less than 72 g for AIS < 3. In fact, the corresponding velocity change of the head during the 0.003 s is over 2 m/s. This may occur when a human impacts a manipulator without a compliant covering. With a compliant covering, the impact period will be much larger than 0.003 s. The impact force and the head acceleration will be reduced and will not reach those high values. On the other hand, recoverable injuries are also unacceptable since they may scare humans away from work closely with the manipulator. No injury (AIS = 0) or at least superficial injuries (AIS =1) should be the goal.

2.2 Mobile robot navigation for avoiding moving obstacles

2.2.1 Introduction

There are two types of wheeled mobile robots according to their kinematics. Holonomic robots (also called omni-directional robots) can independently translate forwards and backwards; translate left and right; and turn left and right. These robots have relatively complex designs. Three or more motors are used to drive three or more special omni-directional wheels. They have better maneuverability than the second mobile robot type. This ability is important for conducting navigation in a crowded area, such as a hospital corridor. The second type, nonholonomic robots have a simpler design. They only need two motors and standard wheels. The mobile robot shown in Fig. 1b is a nonholonomic robot. The term *nonholonomic* means there exist nonintegrable constraints on the robot motion that constrain the velocity, but not the position. For a nonholonomic robot, the constraint is the robot cannot move sideways without turning first. Automobiles are also nonholonomic. The nonholonomic constraint makes the software for controlling nonholonomic robots more complex than that of holonomic robots. Due to the low cost of nonholonomic robots, they are more popular in manufacturing and service applications. Details of the kinematics of these two types of robots will be presented in Appendix D and E, respectively. Many conventional navigation algorithms only work for one of these robot types.

Most of the algorithms for mobile robot navigation are also only suitable for avoiding static obstacles, known-numbers moving obstacles or moving obstacles whose motions are predictable, such as other mobile robots. However, avoiding humans is

different from avoiding predictable obstacles, and poses the following additional challenges:

1) Human motion is unpredictable, changing its speed or direction arbitrarily. For example, in kinematics, human motion is holonomic. A human can stop and move sideways without turning or directly move sideways without a stop. Predicted human positions in accordance with the previous human motion are not reliable. Precise future position information of humans is unobtainable for use in navigation algorithms. This makes the path planning of the navigation system more difficult.
2) For safety, humans must always possess a higher priority in a navigation system than robots or inanimate objects. Humans should be able to pursue their activities unhindered and without worrying about the robots around them. Therefore, the robot may not in the human's sight. Robots should always be aware of the presence of humans and avoid them.
3) Considering human emotional-psychological reactions, when a robot avoids a human it should not remain too close to the human or block the human's path. This can be considered rude behaviour and could also frighten the human, which may cause a sudden action (such as jumping away) and make the human unsafe.
4) Complete knowledge of the current robot work region with humans can only be obtained with a global human sensing system, such as a vision-based tracking system in a closed room to detect the humans in the entire room. However, using this system for a large room is very complex and expensive, and the detection is

also significantly hampered by occlusions. Therefore this knowledge may be impossible to obtain. Further, humans may enter or exit the work region, changing the number of humans that the system must cope with. Some robot navigation algorithms require the number of obstacles to be known *a priori*. This makes those algorithms unsuitable for work regions where humans are present.

Currently, the majority of navigation algorithms for mobile robots can be classified into two categories. The algorithms in the first category directly provide the collision free path(s) of the robot(s) by assuming the shapes of obstacles are known and their motion is predictable. Furthermore, the robot(s) dynamics are neglected. The algorithms in the second category indirectly generate the collision free path(s) by using the current kinematic and geometric information of the robot(s) and obstacles, such as the distances between obstacles and robot(s), and the current velocities of obstacles. These algorithms consider the robot(s) dynamics. The relevant literature on mobile robot navigation algorithms will be reviewed in the proceeding subsections.

2.2.2 The performance criteria for navigation

Various performance criteria for mobile robot navigation have been used. Lee and Lee (1987) and Tsoularis and Kambhampati (1999) used the robot's arrival time to reach the goal, which we will term as T_{arrive}. Vannoy and Xiao (2008) defined the performance criterion as a sum of T_{arrive} and the energy consumption. We will review those researches in further detail in subsections 2.2.3-2.2.4. In the remainder of this subsection, we will review the performance criteria for navigation of autonomous vehicles other than mobile robots.

Spangelo and Egeland (1994) defined performance criteria for navigation of underwater vehicles The criteria they used are T_{arrive} and energy consumption. They defined energy consumption as the integral of the squared input control signals for the DC motors. The friction on the motor and the propeller is ignored in their calculation. A different optimization method was used by Cheng, Liu and Zhang (2006) for the navigation of ships. The authors maximize the distance between the obstacles (*i.e.* islands) and the ship and minimize the distance between the goal and the ship.

Saravanan, Ramabalan and Balamurugan (2009) utilized T_{arrive} , energy consumption and the sum of the minimum distances between a mobile manipulator and the obstacles as the criteria for robotic manipulator navigation in environments with only static obstacles. Their combined criterion is:

$$H = \frac{w_1 T_{arrive}}{N_1} + \frac{w_2 \int_0^{T_{arrive}} \mathbf{u}_{mr}^{\mathrm{T}} \mathbf{u}_{mr} dt}{N_2} + \frac{w_3 \sum_{i=1}^{N} \min\left(d_i^{-1}\right)}{N_3} \tag{2.2.1}$$

where w_1, w_2 and w_3 are the performance weights; N_1, N_2 and N_3 are the parameters used to normalize each performance index; $\mathbf{u}_{mr}$ is the vector of control signals to motors of the mobile robot; and d_i^{-1} is the reciprocal of the distance between the robot and the i^{th} obstacles. A smaller value of H implies the better performance.

2.2.3 Algorithms that directly provide collision free path(s)

Algorithms in the first category have the advantage that optimal collision free paths can be found with stationary obstacles (for examples see: Divebiss and Wen (1997); Earl and D'Andrea (2005); Jaillet, Cortés, and Siméon (2010)) and with moving

obstacles whose motions are predictable. These paths are optimal in the sense that they minimize or maximize one of the performance criteria from subsection 2.2.2. However, those algorithms do not work for moving obstacles like humans due to the lack of accurate position predictions. We will only review the algorithms for avoiding moving obstacles since that is the focus of this thesis.

Lee and Lee (1987) introduced the concept of the *collision map* for avoiding moving obstacles. The collision map shows the predicted collision relationship between the trajectories of the robot and moving obstacles. The trajectories of the robot can be chosen from the collision map with the objective of minimizing T_{arrive}. Park and Lee (2006) extended this algorithm by incorporating fuzzy logic. No experimental results were included in either paper.

Stentz (1994) developed the D* algorithm for navigating a mobile robot in the presence of moving obstacles. The D* algorithm resembles the well-known A* algorithm (for example see: Hart, Nilsson and Rafael 1968) except that it is an incremental search and the arc cost is updated in terms of the robot and obstacle position's change during the navigation. In Ferguson and Stentz (2006), the D* algorithm was extended with linear interpolation to calculate the path cost accurately and generate a smooth path. Similar ideas can be seen in Koenig and Likhachev (2002) and Jacob *et al.* (2010). Koenig and Likhachev developed an incremental search version of A* (referred as Lifelong planning A*) by reusing the previous search results. Jacob *et al.* defined a collision-free triangle in front of the robot. The size of the triangle is related to the robot velocity. The feasible solution for the next sampling period can be searched from this triangle by using A*

algorithm.

Tsubouchi *et al.* (1995) introduced another path planning algorithm for mobile robots. This algorithm first plans a path towards the goal, and then examines whether the robot will collide with any obstacle in its path. Future positions of moving obstacles are predicted by assuming the obstacles will move with zero acceleration. If the robot is predicted to collide with an obstacle, the path is bent to detour around the obstacle. After that, the new path is examined for other obstacles. By iteratively forecasting collisions and re-planning the path every sampling period of the sensing system, a collision-free path is obtained. Based on their simulation results, this algorithm can avoid moving obstacles whose velocities are less than 0.3 m/s. No experimental results were included.

Fox, Burgard, and Thrun (1997) proposed the *dynamic window* method. They define a dynamic window around the robot based on the velocities that the robot can reach without a collision in the next time interval. The window is directly derived from the predicted motions of the robot and obstacles. A velocity for the robot is chosen by maximizing the magnitude of the velocity, maximizing the distance to the obstacle and minimizing the difference between the direction of the velocity and the direction to the goal. The experiments were performed with only stationary obstacles. Marija and Ivan (2007) extended this method for moving obstacles by using D* searching (Stentz 1994) and motion predictions of the robot and the obstacles.

Fiorini and Shiller (1998) presented an algorithm for avoiding moving obstacles. They proposed the concept of *velocity obstacles*, defined as the set of all feasible velocity vectors of the robot that lead to a collision-free path. This is computed by assuming the

velocities of the obstacles do not change during the next sampling period of the sensing system. In their simulations, the robot velocity with maximized magnitude and reachable under the velocity and acceleration limits of the robot is selected for the next sampling period. No results of experiments were published. Yamamoto, Shimada and Mohri (2001) utilized the concept of velocity obstacle for avoiding moving obstacles including humans. In this paper, the feasible velocity vector closest to the velocity vector pointing from the robot towards the goal, and capable of keeping at least a predefined distance away from obstacles, was selected as a command to the robot. This distance defines a safe circular region around an obstacle such that the robot has sufficient space to avoid the obstacle. The authors derived this distance by assuming that the worst condition of the navigation system is a moving obstacle accelerating with its maximum acceleration while approaching the robot. In fact, the worst case for a navigation system is when the robot and obstacles are approaching at their maximum speeds, because the robot then has the least time for avoidance. Simulations with a 2 m/s robot were used to verify their method. This velocity obstacle method was further developed by Large, Laugier and Shiller (2005) with the concept of non-linear velocity obstacle, which is defined in accordance with current velocity and path curvature of the moving obstacle. Two criteria were used to optimally select the robot velocity in the next sampling period, minimizing the traveling time to the goal and the inverse of the time to collision. Larger time to collision means higher safety. They did not present any experimental results.

Rapidly-exploring random tree (RRT) was introduced by LaValle (1998). RRT employs randomization to explore large state spaces and efficiently find path plans. An

RRT is iteratively expanded by moving the mobile robot towards randomly-selected collision-free points. In Lavelle and Kuffner (2001), this RRT algorithm was used to complete the navigation in high-dimensional state spaces. This algorithm can handle nonholonomic constraints and robot dynamics. In those two papers, stationary obstacles were used in simulations. Bruce and Veloso (2002) extended the RRT algorithm by iteratively performing the RRT and taking the previous planned-path points into account when growing the RRT. Authors claimed that this extended RRT reduces re-planning time significantly. They inserted the dynamic window method to their extended RRT algorithm (Bruce and Veloso 2006). Experiments were conducted by using four robots, each with a maximum velocity of 2 m/s. Hsu *et al.* (2002) presented another motion planner for avoiding moving obstacles, which is similar to the RRT algorithm. This planner iteratively builds a tree-shaped roadmap (*i.e.* an interconnected set of collision-free points) in the coordinates of the 3-D space consisting of the predicted robot X-Y position and the future time sequence. Those points are connected by the feasible trajectories that satisfy the kinematic and dynamic motion constraints, including the nonholonomic constraint. The roadmap is recomputed during every sampling interval to avoid moving obstacles. Simulations and experiments with a holonomic robot were performed to validate the algorithm. The maximum velocity of the robot was 0.2 m/s. The maximum velocities of the obstacles were less than or equal to 0.2 m/s, which is much slower than that of moving humans.

Fraichard and Asama (2004) introduced the concept of *inevitable collision states* that is defined as a state for which, no matter what the future trajectory of the robot is, a

collision with the obstacle eventually happens. To define the states for a robot, the shapes, the relative velocity of the robot and the moving obstacle and the dynamics of the robot are taken into account. After the states are defined, the RRT algorithm is used to avoid those states and avoid the obstacles. In 2009, Martinez-Gomez and Fraichard improved the inevitable collision states-based navigation system by using a collision avoidance scheme, ICS-AVOID. However, the moving obstacle's trajectory must be predictable or known *a priori*.

Vannoy and Xiao (2008) proposed a path optimization algorithm for mobile manipulators for avoiding unpredictable obstacles. The performance criterion in their optimization is a combination of minimizing T_{arrive} and energy consumption. The planner generates a set of random trajectory candidates, eliminates the candidates that will produce collision if the velocities of obstacles remain constant, and optimally selects one from the remainder. The path is re-planned during every sampling interval. Simulations of various environments were included that show the algorithm is flexible and able to avoid moving obstacles. However, no experiments were included.

By combining a collision prediction method with beam curvature methods, Shi, Wang and Yang (2010) developed a navigation algorithm for partially known environments. Based on their current velocities, the potential collisions of the robot and obstacles are forecast for ten sampling periods. Beam curvature methods are used to calculate the best heading direction and velocity of the robot to avoid the predicted collisions. Experiments were conducted with robot and obstacle velocities lower than 0.5 m/s.

Also belonging to this category are approaches that assume the obstacles positions are stochastic. For example, Hu, Brady and Probert (1991) used a probabilistic analysis to obtain a region covering 98% of all possible positions of a moving obstacle after a certain time span by assuming the motion of the obstacle is normally distributed. The path is planned to avoid this region. This method has the advantage that it does not require precise predictions of the obstacle's positions. However, the probabilistic analysis requires knowing the variances of Gaussian distributions of the obstacle *a priori*, and a 2% probability of collisions is not reasonable when the obstacles are humans. A cost function that includes the squared velocity difference and squared heading angle difference between the planned path and the actual path of the robot was minimized to generate the control commands for a nonholonomic robot. No experimental results were included in the paper.

2.2.4 Artificial potential field and virtual force field-based algorithms

The navigation algorithms in the second category are based on the artificial potential field (APF) concept (introduced by Khatib 1985) or the virtual force field concept (VFF) concept (introduced by Borenstein and Koren 1989). These algorithms are normally simple to implement, consume less computational load than those in the first category, and are suitable for applications requiring online or real-time navigation. Since virtual forces can be obtained by using the gradient descent method on APFs, they are closely related and belong to the same category. These algorithms assume the existence of a repulsive artificial potential field or a repulsive virtual force surrounding each obstacle to push robots away from the obstacle. An attractive potential field or virtual

force is also assumed to be acting to pull the robot to reach the goal. These algorithms do not directly provide a collision-free path for a robot. The desired accelerations of the robot are given through the robot's dynamic interactions with the virtual forces. The collision-free path can be obtained by integrating the accelerations with Euler's method. APF or VFF-based algorithms have been extensively studied for avoiding stationary obstacles (for examples see: Khatib 1985; Borenstein and Koren 1989 and 1991; Masoud and Masoud 2002; and Vanualailai, Sharma and Nakagiri 2008). In those algorithms, the force functions or the potential fields are only related to the distance from the robot to the goal and the distance(s) to the obstacle(s). VFF or APF-based algorithms have also been proposed for avoiding moving obstacles. These algorithms incorporate the velocities of the obstacles and robots in the force functions and potential fields to help the robots avoid moving obstacles. An *active region* surrounding each obstacle is used in most APF and VFF algorithms to define where the repulsive potential field (or repulsive virtual force) from the obstacle is active. If the robot is inside an obstacle's active region, the repulsive potential field is applied to repel the robot away from the obstacle and the attractive potential field may also be active to move the robot towards goal during navigation. If the robot is outside the region, only the attractive potential field is applied to the robot to pull it towards the goal. The artificial potential field is divided into two parts by the boundary of the active region. The size of the active region is an important factor to design an APF or VFF-based avoidance system. It determines how much space is available for the robot to avoid the obstacles. However, no methods have appeared in the literature for choosing the region size.

In following discussion, we assume there is only one circular obstacle. This will allow us to present and compare the virtual force functions used in the most important papers. Khatib (1985) presented the gradient results of his APF as:

$$\mathbf{F}_a = K_1\mathbf{E} + K_2\dot{\mathbf{E}} \quad \text{and} \tag{2.2.2}$$

$$\mathbf{F}_\Lambda = \begin{cases} -\dfrac{K_{K3}(d_{k2} - d_s)}{d_s^3}\dot{\mathbf{D}} & \text{if } \mathbf{P}_r \in \mathbb{C}_2 \\ 0 & \text{otherwise} \end{cases} \tag{2.2.3}$$

where $\mathbf{F}_a$ and $\mathbf{F}_\Lambda$ are the attractive and repulsive virtual forces, respectively; K_1 and K_2 are the positive attractive force gains; K_{K3} is the positive repulsive force gain; d_s is the shortest distance between the models of the obstacle and the robot; $\mathbf{E}$ is the vector from the robot centre to its goal and $\mathbf{D}$ is the vector from the obstacle centre to the robot centre; $\mathbf{P}_r$ is the position vector of the robot centre; $\mathbb{C}_2$ denotes the active region; and d_{k2} defines the size of the active region of the obstacle. d_{k2} is defined as the shortest distance between the edge of the active region to the exterior contours of the model of the obstacle. In (2.2.2), the attractive force is in the form of the well-known proportional plus derivative (PD) controller. In (2.2.3), the obstacle velocity is included. When the obstacle and the robot approach (*i.e.* when $\dot{\mathbf{D}} < 0$), the obstacle velocity makes the magnitude of $\dot{\mathbf{D}}$ increase. The magnitude of $\mathbf{F}_\Lambda$ increases to push the robot away. However, when the obstacle and the robot move farther apart, $\dot{\mathbf{D}} > 0$, and $\mathbf{F}_\Lambda$ will try to pull the robot towards the obstacle. This is not safe. From (2.2.2) and (2.2.3), we can see that if the robot is inside an obstacle's active region, the attractive potential field is also active to

move the robot towards goal during avoidance. The resultant virtual force, $\mathbf{F}_V$ is:

$$\mathbf{F}_V = \mathbf{F}_a + \mathbf{F}_\Lambda \tag{2.2.4}$$

where $\mathbf{F}_V$ is defined in two-dimensional Cartesian coordinates, *i.e.* $\mathbf{F}_\mathbf{v} = \begin{bmatrix} F_x & F_y \end{bmatrix}^T$. For nonholonomic robots, the force component perpendicular to the robot's heading direction will be neglected since the robot cannot move sideways without turning first. As a consequence, the robot would maintain its previous moving direction rather than moving sideways to avoid the obstacles. The solution to this problem is not provided in this paper. Khatib *et al.* (1999) defined their repulsive force as being proportional to d_s. Since the velocities of the obstacles and robots are not considered, this algorithm is not suitable for motion-unpredictable obstacles.

Borenstein and Koren (1989 and 1991) proposed an alternate repulsive force as follows:

$$\mathbf{F}_\Lambda = \begin{cases} \dfrac{K_{B3}}{d_s}\mathbf{u}_\Lambda & \text{if } \mathbf{P}_\mathrm{r} \in \mathbb{C}_2 \\ 0 & \text{otherwise} \end{cases} \tag{2.2.5}$$

where $\mathbf{u}_\Lambda$ is a unit vector pointing from the obstacle centre to the robot centre, and K_{B3} is the positive repulsive force gain. In (2.2.5), the velocities of the obstacles are not considered. So this VFF is unsuitable for avoiding moving obstacles. Their attractive force, $\mathbf{F}_a$ is simply:

$$\mathbf{F}_a = \begin{cases} 0 & \text{if } \mathbf{P}_\mathrm{r} \in \mathbb{C}_2 \\ K_{B1}\mathbf{u}_E & \text{otherwise} \end{cases} \tag{2.2.6}$$

where $\mathbf{u}_E$ is a unit vector pointing from the robot to its goal, and K_{B1} is the positive

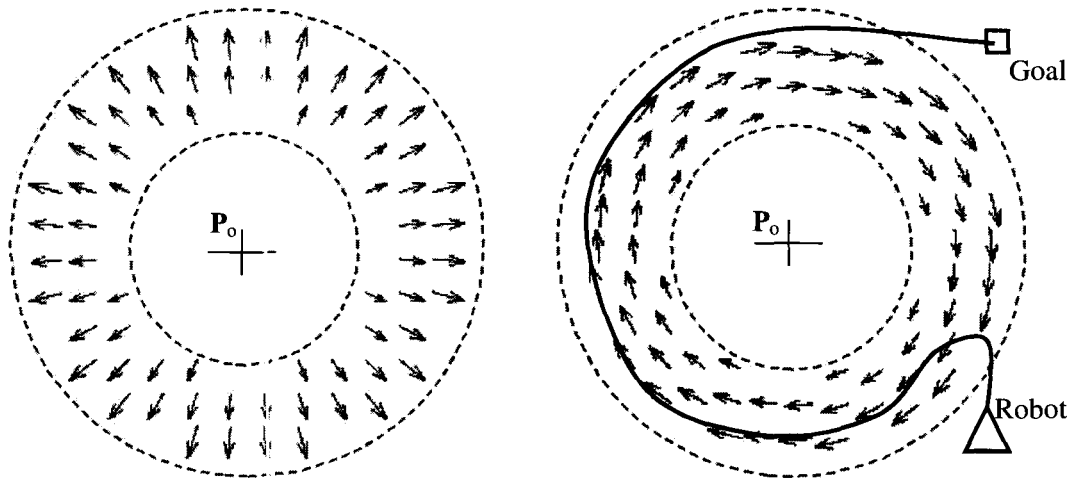
attractive force gain. Since the moving direction of the robot may not point to the goal, the robot may not reach the goal when using (2.2.6). To solve this problem, a virtual torque, M_ϕ was used:

$$M_\phi = K_{BT}\Upsilon_{BT} \tag{2.2.7}$$

where K_{BT} is the positive virtual torque gain; and Υ_{BT} is the angle difference between the angle of $\mathbf{F}_V$ and current heading angle of the robot. Eq. (2.2.7) also helps a nonholonomic robot conduct the avoidance since the virtual torque makes the nonholonomic robot turn. However, the robot's path can oscillate which will increase T_{arrive}. In their simulations, only stationary obstacles were used.

Ko and Lee (1996) introduced a *virtual distance function* that is a function of the current distance from a moving obstacle to the robot and the rate of change of this distance. The repulsive forces are built by using this function to replace d_s in (2.2.3), and the attractive force (2.2.2) was also used. This algorithm was verified with several simulations. In their simulations, only holonomic robots and an obstacle with a constant velocity were used.

Masoud and Masoud (2000) added a tangential component to the repulsive radial component of their potential field, as shown in Fig. 2.4. In the traditional potential fields only the potential magnitudes are specified and the force direction comes from the gradient. With their *vector potential*, the direction may be specified independently of the gradient. This tangential component helps the robot to detour around the obstacle. Simulations were presented including avoiding several mobile robots. In Masoud (2007),



a. Radial component b. Tangential component

Fig. 2.4. The repulsive (a.) potential and the vector potential (b.) in Masoud (2007). $\mathbf{P}_o$ is position of the obstacle (centre of the circular obstacle). The solid line shows a robot may move a longer path to its goal with the tangential component of the potential field.

a Lyapunov function was used to prove the convergence of the potential field to the robot's goal. The attractive, repulsive and vector potential functions were simplified, and their virtual forces are as follows:

$$\mathbf{F}_a = K_{MM1}\mathbf{E}, \tag{2.2.8}$$

$$\mathbf{F}_\Lambda = K_{MM3}\mathbf{D} \quad \text{if } P_r \in \mathbb{C}_2, \text{ and} \tag{2.2.9}$$

$$\mathbf{F}_\psi = K_{MM5}\|\mathbf{D}\|_2 \mathbf{u}_{MM\psi} \quad \text{if } P_r \in \mathbb{C}_2 \tag{2.2.10}$$

where K_{MM1}, K_{MM3} and K_{MM5} are positive force gains; $\mathbf{F}_\psi$ is the virtual force obtained from the vector potential. Since this force acts perpendicular to the repulsive force and helps the robot detour around the obstacle, in this thesis, we will refer to it as the *detour virtual force*. $\mathbf{u}_{MM\psi}$ is the unit vector denoting the direction of this force. $\mathbf{u}_{MM\psi}$ is defined (see Fig. 2.4) independent of the position of the robot. No experiments were presented in those two papers.

Ge and Cui (2002) also used the detour virtual force. They claimed this force helps the robot to detour around the obstacle and reduces T_{arrive}. Their attractive, repulsive, detour and resultant virtual forces are:

$$\mathbf{F}_a = K_1\mathbf{E} + K_2\dot{\mathbf{E}} \quad \text{if } \mathbf{P}_r \notin \mathbb{C}_3, \tag{2.2.11}$$

$$\mathbf{F}_\Lambda = \left[\left(K_{G3} - \dot{d}/\bar{a}_{mr}\right)/\left(d - r_3\right)^2\right]\mathbf{u}_\Lambda \quad \text{if } \mathbf{P}_r \in \mathbb{C}_2, \tag{2.2.12}$$

$$\mathbf{F}_\psi = \left\{\left(K_{G3}\dot{d}\dot{\beta}d\right)/\left(d - r_3\right)^2\right\}\mathbf{u}_\psi \quad \text{if } \mathbf{P}_r \in \mathbb{C}_2 \text{, and} \tag{2.2.13}$$

$$\mathbf{F}_V = \mathbf{F}_a + \mathbf{F}_\Lambda + \mathbf{F}_\psi \tag{2.2.14}$$

where K_{G3} is the positive force gain; β is the angle of the vector of $\mathbf{D}$; $\bar{a}_{mr}$ is the maximum acceleration of the mobile robot; and $\mathbb{C}_3$ defines a region, that we term the *critical region*, that is very close to the obstacle. Since a robot may not have sufficient space to complete avoidance, an alternative to VFF is needed inside this region. This issue was not addressed in this paper. r_3 is the radius of $\mathbb{C}_3$. According to this paper, the region should ensure the robot will not collide with the obstacle before the robot's velocity is reduced to zero while applying a maximum magnitude of the deceleration to the robot. r_3 is computed with:

$$r_3 = \frac{\dot{d}^2}{2\bar{a}_{mr}} \tag{2.2.15}$$

This equation is only suitable for static obstacles. For moving obstacles, such as humans, if an obstacle accelerates to approach the robot (2.2.15) will not provide enough space for stopping the robot. $\mathbf{u}_\psi$ is a unit vector and its direction must satisfy:

$$\begin{aligned} \mathbf{u}_{\Lambda} \cdot \mathbf{u}_{\psi} &= 0 \\ \mathbf{D} \cdot \mathbf{u}_{\psi} &> 0 \end{aligned} \tag{2.2.16}$$

They used the virtual torque from (2.2.7) for the navigation of the nonholonomic robots. Simulations and experiments were performed with a nonholonomic robot and several dynamic and stationary obstacles. The velocities of the moving obstacles were smaller than 0.5 m/s.

Neural networks have also been used in the APF-based algorithms. In Yang and Meng (2003), a neural dynamics based navigation algorithm was proposed for environments with moving obstacles. In this algorithm, a topologically organized neural network system was trained using the completed knowledge of the current environment. The neural dynamic of each neuron is characterized by a function related to *e* and *d*. With those neurons, a potential field was built to navigate the mobile robot. The stability of the neural network was proven with the Lyapunov's second method. Qu *et al.* (2009) extended this algorithm by utilizing a modified pulse-coupled neural network to reduce the computation burden of the network and obtain the shortest path to the goal. No experiments were presented in the two papers. Jung, Jang, and Hsia (2005) used a feed-forward neural network to learn and compensate the uncertainty of obstacles in their VFF-based algorithm. They used (2.2.8) as their attractive force and their repulsive force is proportional to *d*. No experiments were published.

Ren, McIsaac and Patel (2008) introduced a different navigation algorithm based on a dipolar inverse potential field and the modified Newton's method. By modeling the moving obstacle as position and velocity constraints, the robot control commands for

nonholonomic robots were optimally selected to minimize the difference between the feasible robot heading direction and the gradient direction of the potential field. Simulations with two moving obstacles were conducted to verify this algorithm. No experimental results were presented.

The navigation algorithm of Fahimi, Nataraj and Ashrafiuon (2009) modeled the obstacles as rocks in a water flow, and guides the robot to follow the flow and avoid the obstacles. Harmonic attractive and repulsive potential functions are utilized along with the panel method from the field of fluid mechanics. Experiments were performed with obstacles moving at 0.1 m/s to verify their algorithm.

With their VFF-based algorithm, Rosales *et al.* (2010) employed the repulsive force function from Khatib *et al.* (1999) to reduce the velocity of the robot, and changed its heading angle using a new linear algebra-based robot controller. Reducing the velocity in this manner can increase T_{arrive} significantly. Simulations and experiments were presented with multiple moving and stationary obstacles.

2.2.5 Problems with conventional VFF-based or APF-based algorithms

A problem with conventional APF and VFF-based algorithms, that we will term the *collinear condition*, was described by Ge and Cui (2002). With this condition, an obstacle is located between the robot and the goal, and the centres of the robot, obstacle and goal are collinear. The velocity directions of the robot and the obstacle are also on this line (see Fig. 2.5). The attractive force and the repulsive force with conventional VFF algorithms will all be along this line. Hence, the robot either collides with the obstacle, or is pushed away so that it will never reach its goal. Obviously, the solution is to move the

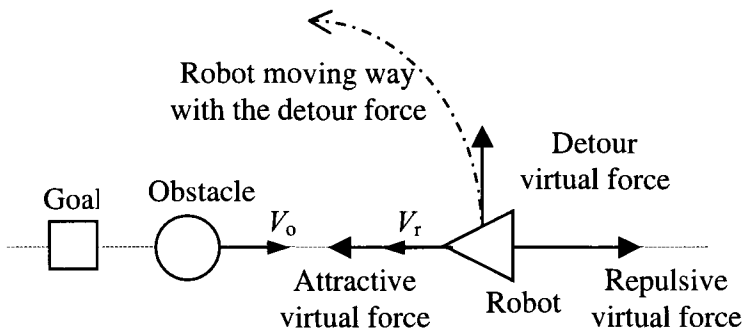


Fig. 2.5. Collinear condition for collision avoidance.

robot sideways. Furthermore, severe path oscillations occur with the existing VFF algorithms whenever the robot, obstacle and goal locations are near to the collinear condition. A non-zero detour force can satisfy this requirement. This was not provided in their paper. In Masoud and Masoud (2000 and 2007), a non-zero detour force is provided. However, since their detour force direction may the mobile robot in a poor direction, a larger T_{arrive} may be caused. An example is shown in Fig. 2.4b.

In APF-based algorithms, the gradient descent method can cause the robot path to oscillate. The modified Newton's method is used to solve this oscillation problem by Ren, McIsaac and Patel (2008). Since the gradient descent method is not used, this source of oscillations does not exist with VFF methods. However, with VFF-based algorithms, if the virtual forces at the boundary of the active region of an obstacle are discontinuous, oscillations may also occur. An example is shown in Fig. 2.6. When the robot first enters into the active region (the disk region in this figure), the repulsive force activates and jumps to a large value with discontinuous VFF algorithms. After this large force pushes the robot out of the region, the attractive force directed towards the goal pulls the robot back into the region, and an oscillating path occurs.

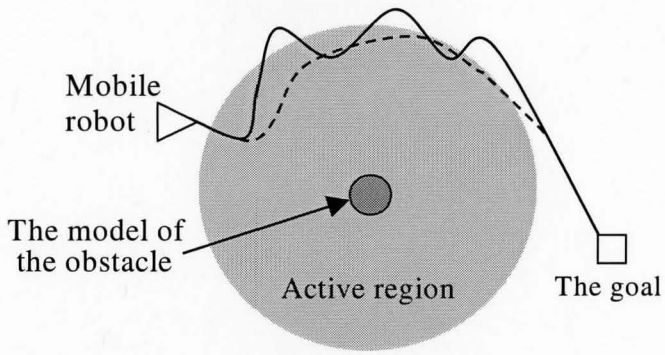


Fig. 2.6. Example of the path oscillation problem. The solid line is the path with a discontinuous VFF. The dashed line is the path with a continuous VFF.

Stability is another unsolved problem for VFF methods with moving obstacles. Some researchers used their APF functions as the Lyapunov function in their APF-based algorithms. Masoud and Masoud (2002) divided a pre-known robot-working region with only stationary obstacles into a grid. Using the analogy of an electrical circuit, each cell of the grid was considered as a resistor, cells occupied by an obstacle as high electric potential and the goal as ground. Then they established a Lyapunov function such that the value of the function is the voltage of the cell presently occupied by the robot. The robot is moved to the surrounding cell that has the lowest potential. In Masoud (2007), this Lyapunov function is extended for moving obstacles. However, the number of the moving obstacles must be known *a priori*, otherwise their Lyapunov function will be discontinuous. Rossetter and Gerdes used a Lyapunov function defined as the quadratic function of the offset of the car from the lane center and used it to solve the stability problem of a lane keeping system for cars. However, this system is mainly one-dimensional and is unhelpful for the two-dimensional mobile robot avoidance problem. Loizou and Kyriakopoulos (2008) utilized the modified dipolar potential field for multi-

robot navigation. By using this field as the Lyapunov function of the navigation system, a backstepping controller is designed to be globally asymptotically stable. The effectiveness of this method is verified through simulations. However, this method requires a perfect knowledge of the working environment to build their APF, and therefore cannot be applied for environments with unpredictable obstacles. Vanualailai *et al.* (2008) designed an APF to be proportional to the distance between the robot and its goal and inversely-proportional to the distance between the robot and an obstacle. By using their APF function as a Lyapunov function, they verified the stability for avoiding stationary obstacles. In their stability analysis, the robot failing to reach its goal in the collinear condition was also emphasized.

2.3. Conclusions

2.3.1 Conclusions for impact force modelling and reduction

Previous researchers have utilized active control methods, actuation methods, passive mechanisms and compliant coverings to reduce the impact force. Active control algorithms suffer from the low bandwidth of the actuators, potentially unstable control systems and reduced payload capability. With the actuator methods, the trajectory tracking performance is significantly reduced due to the decreased impedance. This reduction can be improved by adding a high-bandwidth electric motor but the manipulator's expense is then increased significantly. Passive mechanisms are relatively complex and expensive to implement and maintain. Compliant coverings have been used by several researchers. With their results, we can conclude that using a thin layer of compliant covering to cover the robotic manipulator surface is an inexpensive, simple

and effective way to reduce the impact force. Furthermore, compliant covering, such as elastomeric foam covering can be applied to conventional robots and newly designed robots. Due to its passivity, it is unaffected by power failures and control system faults. Human-manipulator impact dynamic models were proposed in some papers. However, those models are unrealistic since the human dynamics during the impact are simplified incorrectly.

In this thesis, we will investigate the use of elastomeric foam coverings to cover manipulators to reduce the impact force and enhance human safety. Several unsolved problems will be addressed. To effectively perform this investigation, we will derive a more realistic impact dynamic model in Chapter 3. A procedure for determining the maximum and minimum bounds of the foam stiffness and damping will also be presented.

2.3.2 Conclusions for mobile robot navigation

From the literature, the navigation algorithms in the first category are only suitable for the moving obstacles whose motions are predictable. The VFF-based navigation algorithms in the second category are also suitable for avoiding moving obstacles whose motions are unpredictable, such as humans. This is because the critical region can be included in the algorithm and the velocities of the obstacles can be considered in the virtual force functions. Simulations with the algorithms in first category will be used to compare with the algorithms in second category in Chapter 4. However, the following problems remain unsolved to date for the algorithms in the second category: path oscillation, stability for a variable number of obstacles, and the sizes of the

active and critical regions.

In this thesis, we will develop a novel VFF-based navigation algorithm for mobile robots to avoid the humans. The sizes of the active and critical regions will be derived in terms of the kinematic constraints of the human and mobile robot. The new virtual force functions will be designed to be continuous and diminish the path oscillation. An improved detour virtual force will be proposed to solve the collinear condition and reduce T_{arrive}. Lyapunov's second method will be used to analyze the stability of the new VFF. A control scheme will be presented to control nonholonomic robots with the Cartesian VFF. Experiments will be performed and compared with simulations to verify the navigation performance with the new VFF-based algorithm.

Chapter 3

Design of elastomeric foam-covered manipulators to enhance human safety

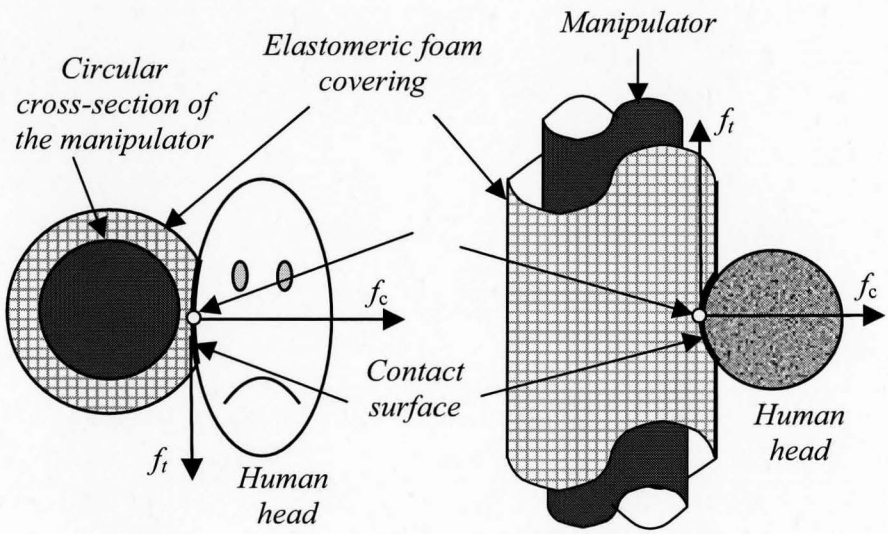
3.1 Introduction

As discussed in Chapter 2, using a foam covering to cover a manipulator is an effective way to reduce the severity of the impact and enhance human safety. In this chapter, we will first state the problem and assumptions. Next, the dynamics of the commonly used articulated manipulator will then be modelled for the three typical impact scenarios. Next, the human-manipulator impact dynamics will be studied by combining the configuration dependent dynamics of the robotic manipulator, the human head-neck-torso system and the foam covering. Based on this model, the key design parameters significantly influencing the impact will be investigated to provide a guide for designing foam-covered manipulators. A model-based procedure for designing elastomeric foam coverings is then proposed. The chapter ends with conclusions.

3.2 Problem description and assumptions

Foam coverings can be attached to all exterior surfaces of a manipulator except its end-effector. Since the end-effector typically has to contact objects in the environment, using a foam covering could interfere with its function. Quick release mechanisms, such as the QuickSTOP collision device (Applied Robotics Inc.), can be employed to reduce the impact force at the end-effector. Impacts on the links of the manipulator will be the

focus of this research. As shown in Fig. 3.1, when a human and manipulator move into each other, the foam is compressed and the compression incurs a force, f_c, that is normal to the contact surface, and a force that is tangential to the surface, f_t. f_t will be discussed later in this section. We will refer to f_c as the *impact force*. During an impact the magnitude of the impact force will vary with time. The initial contact velocity difference between the head and manipulator along this normal direction is defined as the *impact velocity*. The contact surface is the surface where the foam is compressed. As is true for many manipulators, we assume that the manipulator surface does not include sharp corners, so the contact will generate a contact patch with a nonzero surface area. The impact point P_c is the point with the deepest compressed depth of the foam within the contact surface. When there is a difference between the head velocity and manipulator



a. Side view *b*. Top view

Fig. 3.1. The components of the impact force.

velocity that is tangential to the contact surface, the impact point P_c will move along the surface and f_t will be generated. f_t is caused by friction on the contact surface as P_c moves tangentially. It only causes bruises and scrapes, rather than severe human injuries. Therefore, our focus is on f_c, the impact force.

In this study, we make the following assumptions:

1) The manipulator is moving with zero position error and zero acceleration just before the impact;
2) The manipulator joints are controlled by a position control algorithm such as PID control or proportional plus derivative (PD) plus feedforward (FF) control;
3) The gravity load and Coulomb friction (*i.e.* the Coulomb friction in the gearbox of the joint(s)) of the manipulator are balanced mechanically or compensated by the control system;

Regarding the first assumption, acceleration or deceleration of the manipulator during impact will influence the impact force (*i.e.* acceleration will increase the impact force and deceleration will decrease it). To simplify the problem, we assume the manipulator is in a steady-state condition just before the impact: moving with zero acceleration and zero position error. This can be achieved with the commonly used PD+FF or PID control algorithms.

For designing foam covering to enhance human safety, we should have an appropriate safety criterion. According to subsection 2.1.6 of Chapter 2, there are no suitable criteria for human-manipulator cooperation with foam-covered manipulators. To

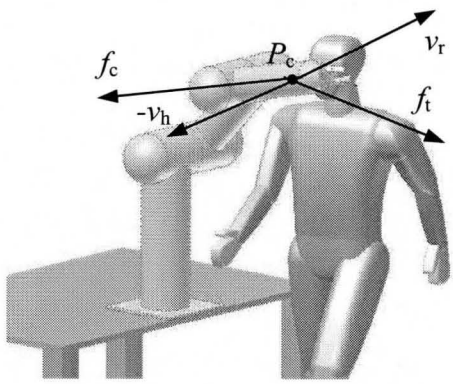
obtain a criterion, biomechanical tests are required. This is beyond the scope of this thesis. We can see those existing criteria are either impact-force-based or head-acceleration-based. Our model-based design procedure proposed in this chapter is suitable for both impact-force-based or head-acceleration-based criteria. Thus any further development of safety criteria can be incorporated into our design methodology. In this thesis, the safety criterion is arbitrarily chosen as a threshold, which is defined as the head acceleration is less than 10 g over the whole impact period (*i.e.* $a_{h,\max} < 10\,\mathrm{g}$ for $a_{h,\max}$ is the largest head acceleration during the impact).

3.3 The impact scenarios

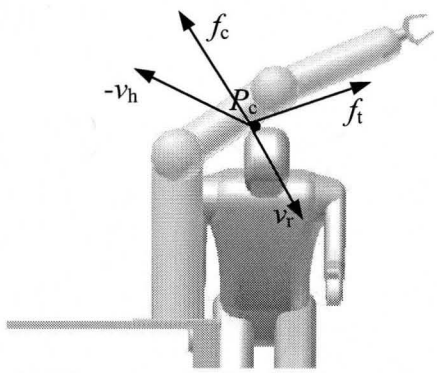
It is obvious that the impact force is correlated with the dynamic characteristics of the manipulator. The impact dynamics of the manipulator are determined by the relationship between the joint(s) and the impact point. The three main impact scenarios for an articulated manipulator are presented in Fig. 3.2. With the first impact scenario, shown in Fig. 3.2(a), the head comes in contact with the foam-covering manipulator on the side of the second or third links. The wrist joint of the manipulator is considered as an extension of the third link. This scenario normally happens with a standing or walking human. From the dynamics of the manipulator, $a_{h,\max}$ is mainly determined by the dynamics of the first joint. Furthermore, the position of the impact point also influences the effective impact dynamics of the manipulator. If the impact point is near the end-effector of the manipulator, the manipulator will be more compliant with respect to the impact, and this tends to result in a smaller $a_{h,\max}$. However, if the point is close to the

joint the manipulator will exhibit a greater stiffness, which could result in a larger $a_{h,\max}$.

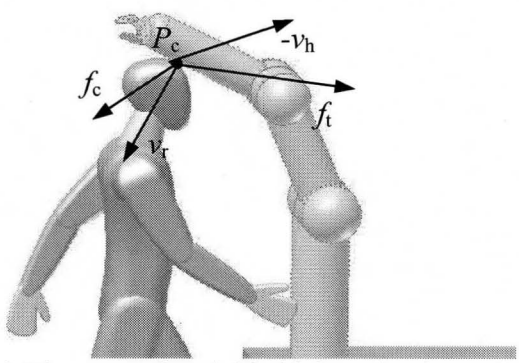
In the second impact scenario, the impact point is on the second link of the manipulator, as shown in Fig. 3.2(b). This impact scenario can occur when the human stands up from a sitting or squatting pose. The second joint is the dominant joint in this scenario. Although the third joint could also have an influence (*i.e.* move downwards slightly), it is much less important than the second joint. For both first and second scenarios, since only one joint dominates the impact, the dynamics of the manipulator can be simplified as a



(a) The first joint dominates the impact



(b) The second joint dominates the impact



(c) The second and third joints dominate the impact

Fig. 3.2. Illustration of the three human-manipulator impact scenarios.

single DOF manipulator when analyzing the impact. In the third impact scenario, the impact point is on the third link of the manipulator, as shown in Fig. 3.2(c). This impact scenario could occur with a human walking toward the manipulator or standing-up from below it. $a_{h,\max}$ is determined from the dynamics of the second and third joints combined. The manipulator is simplified as a two DOF planar manipulator. Furthermore, the impact could be a combination of the first and second scenarios or the first and third scenarios. Since $a_{h,\max}$ will lie in the horizontal plane with the first scenario, and in the vertical plane with the second or third scenarios, we can compute $a_{h,\max}$ in the horizontal and vertical planes and then sum them if one of these situations arises.

3.4 Dynamics of the manipulator during the impact

3.4.1 Introduction

The dynamics of a manipulator during the impact can be described as:

$$\boldsymbol{\tau} = \mathbf{M}\ddot{\boldsymbol{\theta}} + \mathbf{B} + \mathbf{J}^{\mathrm{T}}\mathbf{f} \tag{3.4.1}$$

where $\boldsymbol{\tau}$ is the vector of joint torques; $\boldsymbol{\theta}$ is the vector of joint positions; $\ddot{\boldsymbol{\theta}}$ is the vector of joint accelerations. $\mathbf{M}$ is the mass matrix; $\mathbf{B}$ is the vector of Coriolis and centripetal torques; $\mathbf{J}$ is the Jacobian matrix related to the impact point; and $\mathbf{f}$ is the force vector that includes the impact force and tangential force. For the first three joints, $\boldsymbol{\tau}$, $\boldsymbol{\theta}$, $\ddot{\boldsymbol{\theta}}$, $\mathbf{B}$ and $\mathbf{f}$ are 3×1 vectors and $\mathbf{M}$ and $\mathbf{J}$ are 3×3 matrices. Let us set:

$$\mathbf{M} = \begin{bmatrix} m_{11} & m_{12} & m_{13} \\ m_{21} & m_{22} & m_{23} \\ m_{31} & m_{32} & m_{33} \end{bmatrix},\ \mathbf{B} = \begin{bmatrix} B_1 \\ B_2 \\ B_3 \end{bmatrix} \text{ and } \mathbf{J} = \begin{bmatrix} J_{11} & J_{12} & J_{13} \\ J_{21} & J_{22} & J_{23} \\ J_{31} & J_{32} & J_{33} \end{bmatrix} \tag{3.4.2}$$

3.4.2 Joint Dynamics

Dynamics of a single revolute joint is shown in Fig. 3.3, whereby the parameters are described as follows:

- θ_d is the desired angular position of the joint,
- θ_{enc} is the sensed angular position of the joint typically measured with a rotary encoder,
- θ_J is the actual angular position of the joint,
- τ_{imp} is the torque acting on the joint caused by the impact force,
- I_J equals moment of inertia driven by the joint. Its value is related to the manipulator's mass matrix, $\mathbf{M}$,

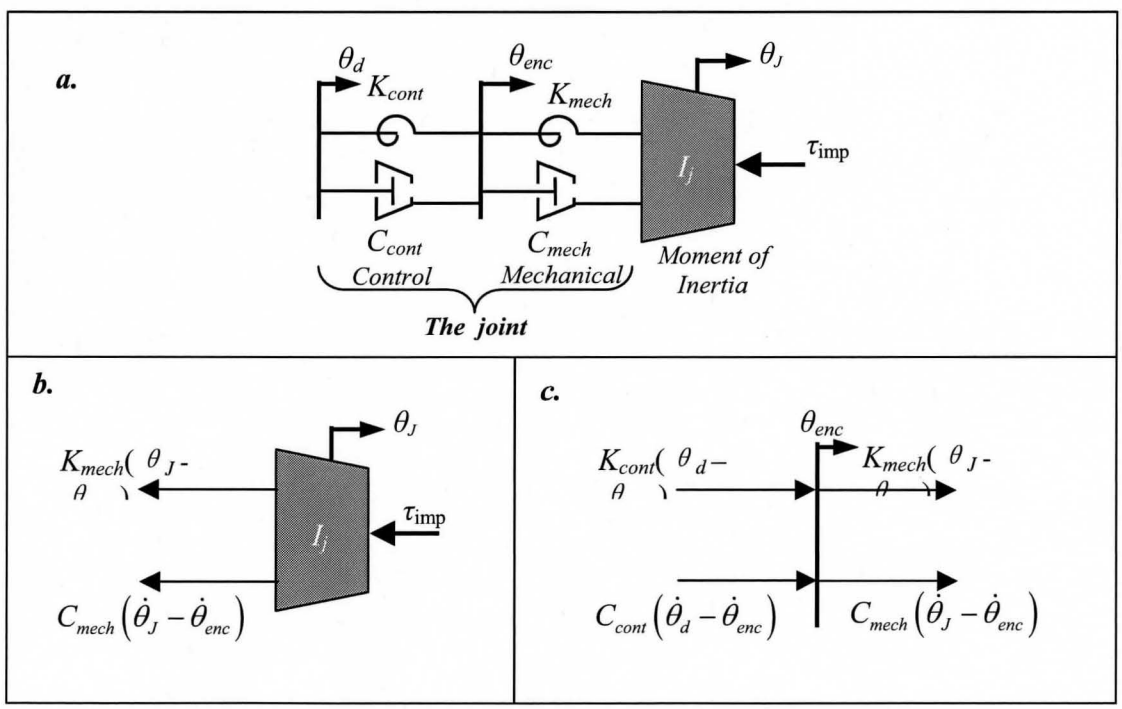


Fig. 3.3. The dynamics of a single revolute joint

- K_{mech} and C_{mech} are the mechanical stiffness and damping of the joint. K_{mech} is related to the deflections of gears in the gearbox of the joint, and also the deflections of the drive shafts. C_{mech} is due to the viscous friction between gears in the gearbox,
- K_{cont} and C_{cont} are the stiffness and damping of the position controller for the joint. When the joint is position controlled by a PD+FF controller, K_{cont} and C_{cont} equal the proportional and derivative gains of the controller, respectively. If a PID controller is used, K_{cont} and C_{cont} also equal the proportional and derivative gains[1].

From Fig. 3.3b, we have the following equation:

$$-\left(K_{mech}+C_{mech}s\right)\left[\theta_J(s)-\theta_{enc}(s)\right]-\tau_{imp}(s)=I_J\ddot{\theta}_J \tag{3.4.3}$$

Defining the joint torque: $\tau_J(s)=\tau_{imp}(s)+I_Js^2\theta_J(s)$:

$$-\left(K_{mech}+C_{mech}s\right)\left[\theta_J(s)-\theta_{enc}(s)\right]=\tau_J(s) \tag{3.4.4}$$

Defining the motor torque $\tau_m(s)=\left(K_{cont}+C_{cont}s\right)\left[\theta_d(s)-\theta_{enc}(s)\right]$, from Fig. 3.3c:

$$\left(K_{mech}+C_{mech}s\right)\left[\theta_J(s)-\theta_{enc}(s)\right]+\tau_m=0 \tag{3.4.5}$$

Eliminating $\theta_{enc}(s)$ from (3.4.4) to (3.4.5), and then re-organizing the result gives:

$$\frac{\tau_J(s)}{\left[\theta_J(s)-\theta_d(s)\right]}=\frac{\left(K_{mech}+C_{mech}s\right)\left(K_{cont}+C_{cont}s\right)}{\left(K_{mech}+K_{cont}\right)+\left(C_{mech}+C_{cont}\right)s} \tag{3.4.6}$$

In this research, to simplify the problem, we will use the following reduced order model

[1] Based on a simulation study, the integral term acts too slowly to have a significant influence on the human-manipulator impact.

to approximate (3.4.6):

$$\tau_J(s)/\tilde{\theta} \approx K_J + C_J s \tag{3.4.7}$$

where $\tilde{\theta} = \theta_J(s) - \theta_d(s)$; and K_J and C_J are the approximate stiffness and damping of the joint. Therefore we need to make:

$$(K_J + C_J s) \approx \frac{(K_{mech} + C_{mech}s)(K_{cont} + C_{cont}s)}{(K_{mech} + K_{cont}) + (C_{mech} + C_{cont})s} \tag{3.4.8}$$

Using long division, the approximated stiffness and damping are:

$$K_J = \frac{K_{mech}K_{cont}}{K_{mech} + K_{cont}} \text{ and} \tag{3.4.9a}$$

$$C_J = \frac{K_{cont}^2 C_{mech} + K_{mech}^2 C_{cont}}{(K_{cont} + K_{mech})^2} \tag{3.4.9b}$$

Therefore, the error of (3.4.7) relative to (3.4.6) is:

$$E_m(s) = \frac{\left[C_{mech}C_{cont} - (C_{mech} + C_{cont})C_r\right]s^2}{(K_{mech} + K_{cont}) + (C_{mech} + C_{cont})s} \tag{3.4.10}$$

From (3.4.9a) and (3.4.9b), if the transmission elements are very rigid, $K_{mech} \gg K_{cont}$, we have $K_J \approx K_{cont}$ and $C_J \approx C_{cont}$. This applies to the compliance control research presented in Heinzmann and Zelinsky (2003). They decreased the control stiffness to reduce K_J and f_c. Similarly, if $K_{cont} \gg K_{mech}$ then $K_J \approx K_{mech}$ and $C_J \approx C_{mech}$. This idea has been used in Lim and Tanie (2000) and Park *et al.* (2007). They decreased the mechanical stiffness to reduce K_J and f_c.

3.4.3 Dynamics of the manipulator for the first and second impact scenarios

For both the first and second impact scenarios, the manipulator can be modeled as a single revolute joint manipulator. These models will be derived and converted into a single DOF prismatic manipulator in this subsection. The first impact scenario is shown in Fig. 3.4. f_{c1} is the impact force acting on the manipulator. $\tilde{\theta}_1 = \theta_{d1} - \theta_{J1}$ where θ_{d1} is the desired angular position of joint 1 and θ_{J1} is the actual angular position of joint 1. Before the impact, as explained in Section 3.2, $\tilde{\theta}_1 = 0$. The moment of inertia driven by the first joint is I_{J1}; the angular stiffness and damping of the joint are K_{J1} and C_{J1}. The offset distance from the impact point P_c to the centre of rotation of the joint is l_1. The velocity of the manipulator normal to the contact surface and just prior to the impact is

$$v_r(0) = \dot{\theta}_{d1} l_1 \tag{3.4.11}$$

This is the initial contact velocity of the manipulator. Note the value of this velocity is not influenced by the radius of the cylindrical manipulator link. As in Fig. 3.5, we have:

$$v_t = v_{P_c} \cos(\gamma_{link}) = \dot{\theta}_{J1} \left(l_1 / \cos(\gamma_{link}) \right) \cos(\gamma_{link}) = \dot{\theta}_{J1} l_1 \tag{3.4.12}$$

The velocity of the human head is v_h and can be separated into two components, v_n and v_t, where v_n is normal to the contact surface. $v_n(0)$ is the initial contact velocity of the human head. The impact velocity is:

$$v_c = v_r(0) - v_h(0)\cos\gamma_c = v_{r1}(0) - v_n(0) \tag{3.4.13}$$

If $v_c > 0$, the impact happens and incurs a normal impact force f_{c1}, otherwise there is no impact. The tangential velocity, v_t, will cause the human head to slide along the surface

of the manipulator during the impact and change l_1. The effect of changing l_1 will be studied in Section 3.7.3. From (3.4.13), the impact velocity is largest when $\gamma_c = 0$, and thus $v_n = v_h$. A larger impact velocity will cause a larger impact force and $a_{h,\max}$. Therefore, the condition of $\gamma_c = 0$ is the worst case for this impact point since the largest impact velocity will be produced. During the impact, the angular position of the manipulator will change. Therefore, v_r will not remain normal to the contact surface (*i.e.* $\cos\gamma_c < 1$) and v_n will be reduced. This will also reduce the impact force. To simplify

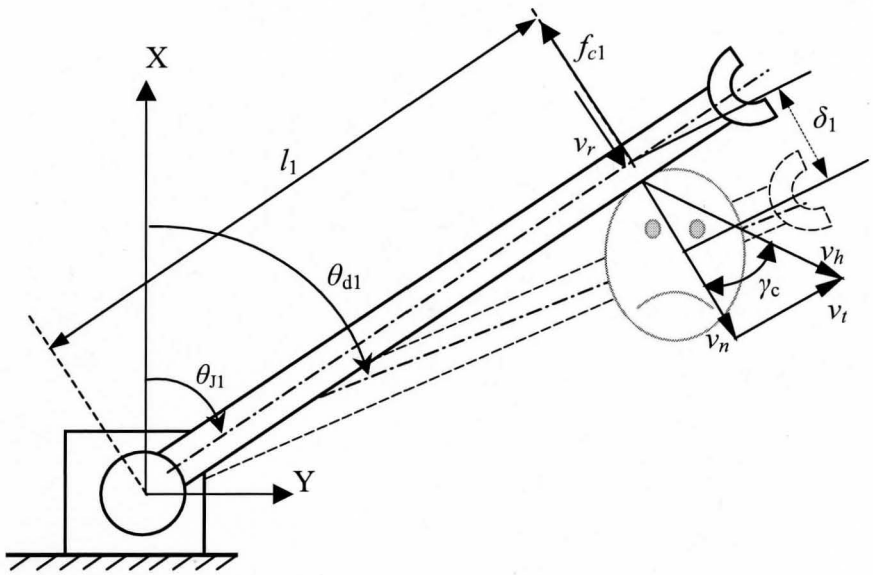


Fig. 3.4. The physics of the first impact scenario. Note that the position of v_r is slightly moved to show the linear velocity of the manipulator clearly.

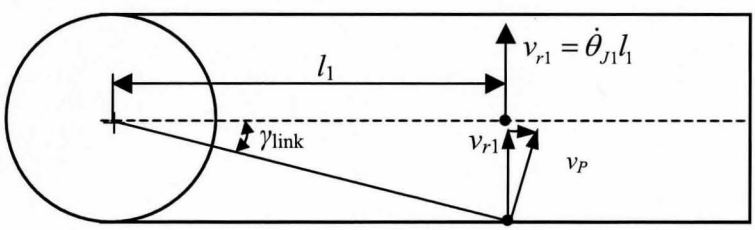


Fig. 3.5. The physics of the manipulator velocity with the large radius

the analysis, we make the conservative assumption that the velocity direction of the head is always normal to the contact surface (*i.e.* $\gamma_c = 0$) during the impact. Since only the first joint dominates the impact, from (3.4.2) and (3.4.3), the dynamics of the manipulator in this condition can be simply written as:

$$m_{11}\ddot{\theta}_{J1} + J_{11}f_{c1} = \tau_{J1} \tag{3.4.14}$$

where $J_{11} = l_1$, and τ_{J1} is the torque generated by the first joint. Note that $\mathbf{B} = B_1 = 0$ since only one joint dominates the impact. From (3.4.7), we have the joint dynamics equation:

$$\tau_{J1} = K_{J1}\tilde{\theta}_1 + C_{J1}\dot{\tilde{\theta}}_1 \tag{3.4.15}$$

From the kinematics of Fig. 3.5, and since $\ddot{\theta}_{d1} = 0$ from the third assumption in Section 3.2, we have: $\ddot{\theta}_{J1} = \ddot{\delta}_1 / l_1$, $\dot{\tilde{\theta}}_1 = \dot{\delta}_1 / l_1$ and $\tilde{\theta}_1 = \delta_1 / l_1$, where δ_1 is the linear displacement of P_c. Substituting those equations and (3.4.15) into (3.4.14), and dividing the result by l_1 gives:

$$f_{c1} = \frac{m_{11}}{l_1^2}\ddot{\delta}_1 + \frac{C_{J1}}{l_1^2}\dot{\delta}_1 + \frac{K_{J1}}{l_1^2}\delta_1 \tag{3.4.16}$$

Eq. (3.4.16) is the dynamics of the manipulator during the impact. It is identical to the contact dynamics of a single DOF manipulator with a prismatic joint. Therefore with (3.4.16) the manipulator with the revolute joint is emulated by a manipulator with a prismatic joint. The effective mass, stiffness and damping of the manipulator for the first impact scenario are:

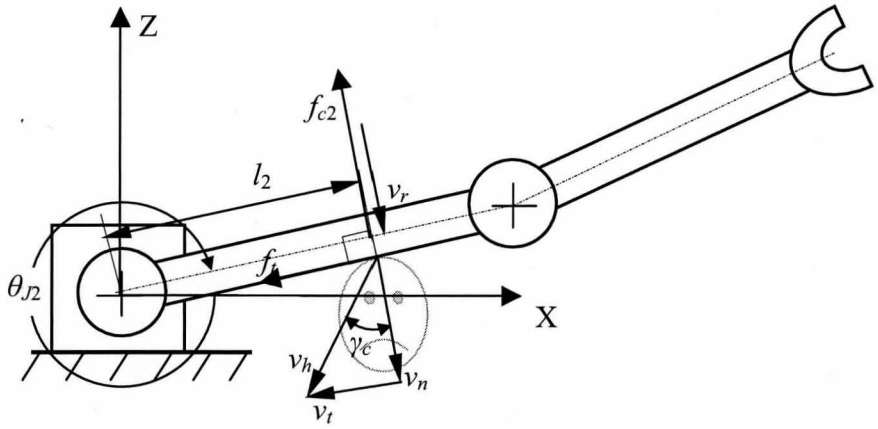


Fig. 3.6. The physics of the second impact scenario. Note that the position of v_r is slightly moved to show the linear velocity

$$M_{r1} = \frac{m_{11}}{l_1^2},\ C_{r1} = \frac{C_{J1}}{l_1^2} \text{ and } K_{r1} = \frac{K_{J1}}{l_1^2} \tag{3.4.17}$$

For the second scenario (shown in Fig. 3.6), assuming joint 3 is rigid, following the analysis from (3.4.11) to (3.4.17), the initial contact velocity, effective mass, stiffness and damping of the manipulator is:

$$v_r(0) = \dot{\theta}_{d2} l_2,\ M_{r2} = \frac{m_{22}}{l_2^2},\ C_{r2} = \frac{C_{J2}}{l_2^2} \text{ and } K_{r2} = \frac{K_{J2}}{l_2^2} \tag{3.4.18}$$

3.4.4 Dynamics of the manipulator for the third impact scenario

For the third impact scenario, as shown in Fig. 3.7, the impact dynamics are related to joints 2 and 3. The manipulator dynamics can be denoted as:

$$\mathbf{M}_{23}\begin{bmatrix}\ddot{\theta}_{J2}\\ \ddot{\theta}_{J3}\end{bmatrix} + \mathbf{B}_{23} + \mathbf{J}_{23}^{\mathrm{T}}\begin{bmatrix}f_t\\ f_{c3}\end{bmatrix} = \mathbf{K}_{23}\begin{bmatrix}\tilde{\theta}_2\\ \tilde{\theta}_3\end{bmatrix} + \mathbf{C}_{23}\begin{bmatrix}\dot{\tilde{\theta}}_2\\ \dot{\tilde{\theta}}_3\end{bmatrix} \tag{3.4.19}$$

where f_{c3} is the impact force; $\mathbf{M}_{23} = \begin{bmatrix} m_{22} & m_{23} \\ m_{32} & m_{33} \end{bmatrix}$ is the mass matrix of this two-DOF

manipulator; $\mathbf{B}_{23} = \begin{bmatrix} B_2 & B_3 \end{bmatrix}^T$ is the vector of Coriolis and centripetal torques; $\mathbf{K}_{23} = \begin{bmatrix} K_{J2} & 0 \\ 0 & K_{J3} \end{bmatrix}$ and $\mathbf{C}_{23} = \begin{bmatrix} C_{J2} & 0 \\ 0 & C_{J3} \end{bmatrix}$ are the stiffness and damping matrix, respectively; the elements of $\mathbf{K}_{23}$ and $\mathbf{C}_{23}$ can be calculated using (3.4.9). The Jacobian matrix in the u_n–u_t coordinate frame is

$$\mathbf{J}_{23} = \begin{bmatrix} J_{22} & J_{23} \\ J_{32} & J_{33} \end{bmatrix} = \begin{bmatrix} -l_3 \sin\theta_{J23} - a_2 \sin\theta_{J2} & -l_3 \sin\theta_{J23} \\ l_3 \cos\theta_{J23} + a_2 \cos\theta_{J2} & l_3 \cos\theta_{J23} \end{bmatrix} \tag{3.4.20}$$

where $\theta_{J23} = \theta_{J2} + \theta_{J3}$. The impact velocity is:

$$v_c = v_r(0)\cos\gamma_r - v_h(0)\cos\gamma_c \tag{3.4.21}$$

Since the velocity of the manipulator should be limited for the safety reason, $\mathbf{B}_{23}$ should not have a significant effect on the impact and will be ignored in remaining analysis. When $\theta_3 = 0$, every element in $\mathbf{M}_{23}$ will be at their maximum values, which will produce the largest impact force. For example, if the second and third links have uniformly distributed mass, their lengths are a_2 and a_3, and their masses are m_2 and m_3, $\mathbf{M}_{23}$ is:

$$\mathbf{M}_{23} = \begin{bmatrix} \frac{1}{3}m_2a_2^2 + m_3a_2^2 + \frac{1}{3}m_3a_3^2 + m_3a_2a_3\cos\theta_{J3} & \frac{1}{3}m_3a_3^2 + m_3a_2a_3\cos\theta_{J3} \\ +\frac{1}{3}m_3a_3^2 + m_3a_2a_3\cos\theta_{J3} & \frac{1}{3}m_3a_3^2 \end{bmatrix} \tag{3.4.22}$$

When $\theta_3 = 0$, $\cos\theta_3 = 1$ and every element in $\mathbf{M}_{23}$ is maximized. To be conservative, we will study this condition, as shown in Fig. 3.8. The initial contact velocity of the manipulator can be calculated with:

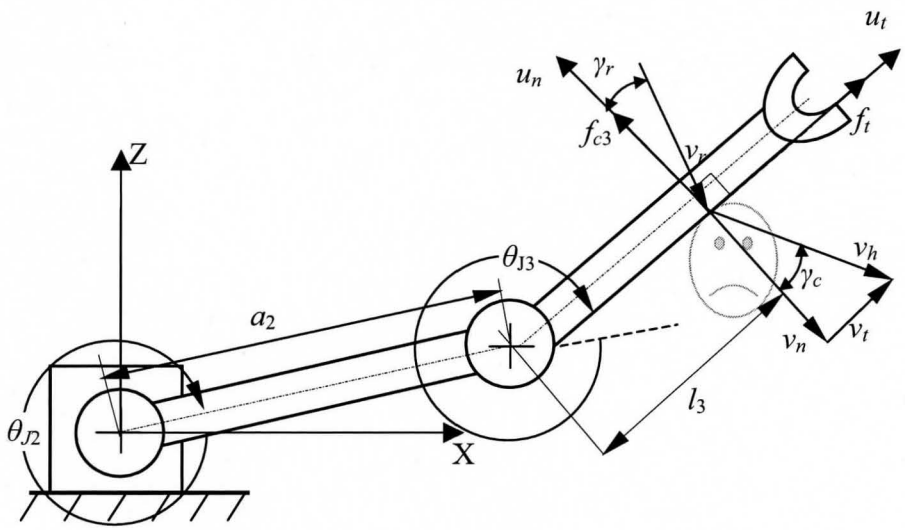


Fig. 3.7. The physics of the third impact scenario.

$$v_r(0) = \begin{bmatrix} 0 \\ 1 \end{bmatrix}^T \begin{bmatrix} 0 & 0 \\ a_2 + l_3 & l_3 \end{bmatrix} \begin{bmatrix} \dot{\theta}_{d2} \\ \dot{\theta}_{d3} \end{bmatrix} = (a_2 + l_3)\dot{\theta}_{d2} + l_3\dot{\theta}_{d3} \tag{3.4.23}$$

Since $\theta_{J3} = 0$, and the dynamics of the manipulator becomes:

$$-\begin{bmatrix} l_3 + a_2 \\ l_3 \end{bmatrix} f_{c3} = \mathbf{M}_{23} \begin{bmatrix} \ddot{\theta}_2 \\ \ddot{\theta}_3 \end{bmatrix} - \mathbf{K}_{23} \begin{bmatrix} \tilde{\theta}_2 \\ \tilde{\theta}_3 \end{bmatrix} - \mathbf{C}_{23} \begin{bmatrix} \dot{\tilde{\theta}}_2 \\ \dot{\tilde{\theta}}_3 \end{bmatrix} \tag{3.4.24}$$

To simplify the remaining analysis, we rewrite (3.4.24) as:

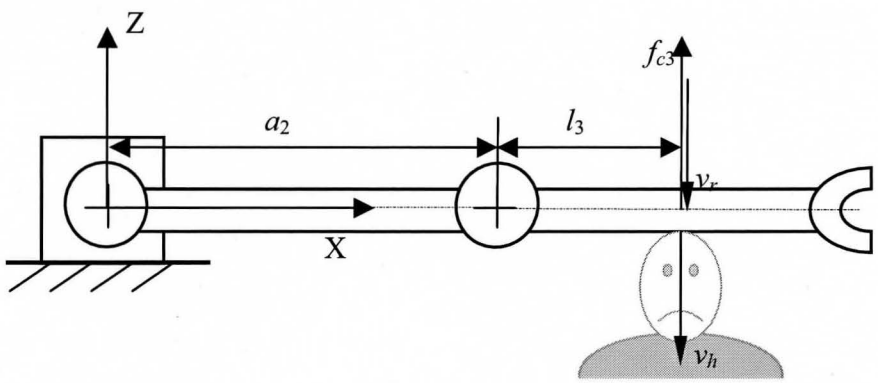


Fig. 3.8. The worst case for the third impact scenario. Note that the position of v_r is slightly moved to show the linear velocity of the manipulator clearly.

$$\begin{bmatrix} l_3 + a_2 \\ l_3 \end{bmatrix} f_{K3} = \mathbf{K}_{23} \begin{bmatrix} \tilde{\theta}_2 \\ \tilde{\theta}_3 \end{bmatrix} \tag{3.4.25}$$

where $f_{K3} = f_{c3} - \begin{bmatrix} 1/(l_3 + a_2) \\ 1/l_3 \end{bmatrix}^T \left(\mathbf{M}_{23} \begin{bmatrix} \ddot{\theta}_2 \\ \ddot{\theta}_3 \end{bmatrix} - \mathbf{C}_{23} \begin{bmatrix} \dot{\tilde{\theta}}_2 \\ \dot{\tilde{\theta}}_3 \end{bmatrix} \right)$ is the magnitude of the force caused by the stiffness. The displacement is:

$$\delta_3 = \begin{bmatrix} a_2 + l_3 & l_3 \end{bmatrix} \begin{bmatrix} \tilde{\theta}_2 \\ \tilde{\theta}_3 \end{bmatrix} \tag{3.4.26}$$

Multiplying (3.4.26) by $\begin{bmatrix} a_2 + l_3 & l_3 \end{bmatrix} \mathbf{K}_{23}^{-1}$ and substituting (3.4.25) gives:

$$\left[\frac{(l_3 + a_2)^2}{K_{J2}} + \frac{l_3^2}{K_{J3}} \right] f_{K3} = \delta_3 \tag{3.4.27}$$

Therefore, the effective stiffness of the manipulator at this impact point is:

$$K_{r3} = \frac{f_{K3}}{\delta_3} = \frac{K_{J2} K_{J3}}{(l_3 + a_2)^2 K_{J3} + l_3^2 K_{J2}} \tag{3.4.28}$$

Similarly, since $\dot{\delta}_3 = \begin{bmatrix} a_2 + l_3 & l_3 \end{bmatrix} \begin{bmatrix} \dot{\tilde{\theta}}_2 & \dot{\tilde{\theta}}_3 \end{bmatrix}^T$, the effective damping of the manipulator is:

$$C_{r3} = \frac{f_{c3}}{\dot{\delta}_3} = \frac{C_{J2} C_{J3}}{(l_3 + a_2)^2 C_{J3} + l_3^2 C_{J2}} \tag{3.4.29}$$

Since the manipulator moves slowly, $\dot{\mathbf{J}}_{23} \approx 0$, taking the second derivative of (3.4.26) gives:

$$\ddot{\delta}_3 = (l_3 + a_2) \ddot{\theta}_2 + l_3 \ddot{\theta}_3 \tag{3.4.30}$$

Defining $f_{M3} = f_{c3} - C_{r3} \dot{\delta}_3 - K_{r3} \delta_3$, we have:

$$\begin{bmatrix} l_3 + a_2 \\ l_3 \end{bmatrix} f_{M3} = \mathbf{M}_{23} \begin{bmatrix} \ddot{\theta}_2 \\ \ddot{\theta}_3 \end{bmatrix} \tag{3.4.31}$$

Multiplying (3.4.31) by $\begin{bmatrix} a_2 + l_3 & l_3 \end{bmatrix} \mathbf{M}_{23}^{-1}$ and substituting (3.4.30) gives:

$$\begin{bmatrix} l_3 + a_2 & l_3 \end{bmatrix} \mathbf{M}_{23}^{-1} \begin{bmatrix} l_3 + a_2 & l_3 \end{bmatrix}^T f_{M3} = \ddot{\delta}_3 \tag{3.4.32}$$

Therefore, the effective mass of the manipulator is:

$$M_{r3} = \frac{f_{M3}}{\ddot{\delta}_3} = \frac{1}{\begin{bmatrix} l_3 + a_2 & l_3 \end{bmatrix} \mathbf{M}_{23}^{-1} \begin{bmatrix} l_3 + a_2 & l_3 \end{bmatrix}^T} \tag{3.4.33}$$

3.5 Impact Dynamic Model

The structure of our model is shown in Fig. 3.9. x_d is the desired linear position of the impact point, and x_r is the actual linear position of the impact point. For the first and second impact scenarios: $x_d = \theta_{d1} l_1$, $x_r = \theta_{J1} l_1$; and $x_d = \theta_{d2} l_2$, $x_r = \theta_{J2} l_2$, respectively. For the worst case of the third impact scenario, $x_d = \theta_{d2} a_{l2} + \theta_{d3} l_3$ and $x_r = \theta_{J2} a_{l2} + \theta_{J2} l_3$. Before the impact, $x_d = x_r$. Note that the control action of the manipulator has been included in the spring (*i.e.* K_r) and damper (*i.e.* C_r) of the

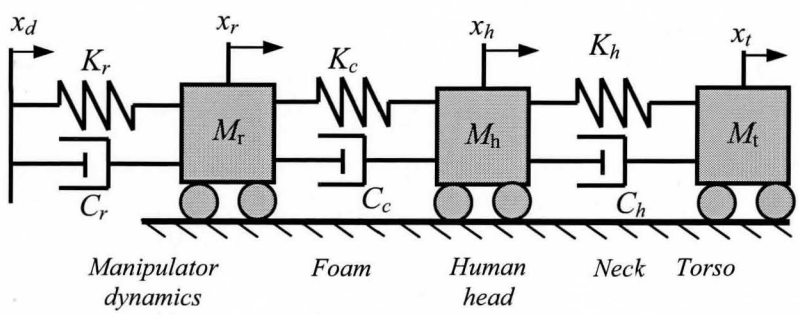


Fig. 3.9. The impact dynamics model

manipulator dynamics (see (3.4.8)). The impact begins when the foam covering attached to the manipulator first contacts the human's head. We set this time as $t = 0$ and define $x_d(0) = 0$ and $x_r(0) = 0$. From our first assumption, we have $\dot{x}_d(t) = v_r(0)$. The foam is modelled as a compressive spring and damper in parallel. The human head is modelled as a point mass. The dynamics of the human neck is modelled as a spring and a damper in parallel. As a conservative and simplifying assumption, since the mass of the human torso M_t is much larger than the head mass, M_h, M_t is assumed to be infinite. With this assumption, the human torso will keep its previous moving velocity without any deceleration during the impact period, *i.e.* $\dot{x}_t(t) = v_h(0)$. In reality, the torso would be decelerated by the impact force transmitted through the neck so the impact force and $a_{h,\max}$ would be less than that predicted by our model.

Based on our model, the impact force is:

$$\begin{aligned} f_c(t) &= K_c\varepsilon + C_c\dot{\varepsilon} \\ &= K_c\left(T_u - x_h(t) + x_r(t)\right) + C_c\left(\dot{x}_r(t) - \dot{x}_h(t)\right) \end{aligned} \tag{3.5.1}$$

where $\varepsilon = T_u - x_h(t) + x_r(t)$ is the compressed depth of the foam and T_u is the uncompressed thickness of the foam. The equation of the contact dynamics of the robotic manipulator is:

$$-f_c(t) = M_r\ddot{x}_r(t) + C_r\left[\dot{x}_r(t) - \dot{x}_d(t)\right] + K_r\left[x_r(t) - x_d(t)\right] \tag{3.5.2}$$

The values of M_r, C_r and K_r are calculated using either (3.4.17), (3.4.18), (3.4.28), (3.4.29) or (3.4.33), depending on the impact scenario. Note that $x_r(t) - x_d(t) = \delta(t)$. The initial conditions are $x_d(0) = x_r(0) = 0$, $\dot{x}_d(t) = v_r(0)$ and $x_d(0) = 0$, we have

$x_d(t) = v_r(0)t$. The value of the desired manipulator velocity, $\dot{x}_d(t)$ is dependent on the application. In this thesis, we select the maximum linear velocity of the manipulator to be smaller than 0.25 m/s. Regarding this value, when a human cooperates with a robotic manipulator, the maximum speed of the end-effector at full extension shall not exceed 0.25 m/s in accordance with current safety standards (ANSI/RIA R15.06-1999 and ISO10218-1-2006).

The dynamics of the head are:

$$f_c(t) = M_h\ddot{x}_h(t) + C_h\left(\dot{x}_h(t) - \dot{x}_t(t)\right) + K_h\left(x_h(t) - x_t(t)\right) \tag{3.5.3}$$

In (3.5.3), the initial condition of $x_h(0) = x_t(0) = T_u$ can be arbitrary. In this thesis, to simplify the derivation, we selected the initial conditions are: $x_h(0) = x_t(0) = T_u$ (*i.e.* the head is directly above the torso at $t = 0$ and then no elastic energy is stored in the spring of the human neck). Since $\dot{x}_h(0) = v_h$ and $x_t(t) = T_u$, we have $x_t(t) = v_h(0)t + T_u$. Based on the data presented by Viano (2003), the stiffness of the neck is $K_h \approx 3$ kN/m for females and 5 kN/m for males. The human head is modeled as a point mass, and M_h ranges from 4.4 to 5.3 kg (Willinger *et al.* 2005). The human neck damping ratio ξ_h ranges from 0.2 to 0.4. Since $C_h \approx 2\xi_h\sqrt{M_hK_h}$, C_h ranges from 46 to 130 Ns/m. The range of human velocity is application dependant. In this thesis, we will address the impact condition with a stationary or walking human whose velocity is limited to less than 1.0 m/s (van Emmerik and Wagenaar 1996). For the condition with a running human (v_h > 1.5 m/s), the thickness of elastomeric foam covering required for safe impact would be several centimetres which is impractical for most manipulators. Therefore, with the velocity

limits of the human(s) and manipulator(s), the maximum impact velocity equals 1.0 + 0.25 = 1.25 m/s in this thesis.

Taking the Laplace transform of (3.5.1) — (3.5.3), respectively, gives:

$$\begin{aligned} F_c(s) &= K_c\left(\frac{T_u}{s} - X_h(s) + X_r(s)\right) + C_c\left(-sX_h(s) - X_h(0) + sX_r(s) - X_r(0)\right) \\ &= \left(K_c + C_c s\right)\left(\frac{T_u}{s} - X_h(s) + X_r(s)\right), \end{aligned} \tag{3.5.4}$$

$$-F_c(s) = \left(M_r s^2 + C_r s + K_r\right)X_r(s) - \left(M_r s^2 + C_r s + K_r\right)\frac{v_r(0)}{s^2} \quad \text{and} \tag{3.5.5}$$

$$F_c(s) = \left(M_h s^2 + C_h s + K_h\right)X_h(s) - \left(M_h s^2 + C_h s + K_h\right)\left(\frac{v_h(0)}{s^2} + \frac{T_u}{s}\right). \tag{3.5.6}$$

Re-organizing (3.5.4), (3.5.5) and (3.5.6), we have

$$F_c(s) = \left(K_c + C_c s\right)\left(X_r(s) - X_h(s)\right) + \left(K_c + C_c s\right)\frac{T_u}{s}, \tag{3.5.7}$$

$$X_r(s) = \frac{-F_c(s)}{\left(M_r s^2 + C_r s + K_r\right)} + \frac{v_r(0)}{s^2} \quad \text{and} \tag{3.5.8}$$

$$X_h(s) = \frac{F_c(s)}{\left(M_h s^2 + C_h s + K_h\right)} + \left(\frac{v_h(0)}{s^2} + \frac{T_u}{s}\right) \tag{3.5.9}$$

Substituting (3.5.8) and (3.5.9) into (3.5.7),

$$\begin{aligned} F_c(s) = \left(K_C + C_c s\right)&\left(-\frac{F_c(s)}{M_r s^2 + C_r s + K_r} - \frac{F_c(s)}{M_h s^2 + C_h s + K_h}\right) \\ &+ \left(K_C + C_c s\right)\frac{v_r(0) - v_h(0)}{s^2} \end{aligned} \tag{3.5.10}$$

Since the impact velocity $v_c = v_r(0) - v_h(0)$, the impact force in the frequency domain is:

$$F_c(s) = \frac{(K_c + C_c s)v_c}{s^2\left(1 + \frac{K_c + C_c s}{M_r s^2 + C_r s + K_r} + \frac{K_c + C_c s}{M_h s^2 + C_h s + K_h}\right)} \tag{3.5.11}$$

Using the inverse Laplace transform, the force equation in the time domain can be analytically derived. Due to the complexity of the resulting equation, the inverse transform will be neglected and $f_c(t)$ will be computed numerically. From (3.5.11), it is obvious that the impact force is proportional to v_c. If $v_c \le 0$, the impact will not happen since the human head is moving away from the robotic manipulator. The impact force is zero in this condition. If $v_c > 0$, the head and the manipulator are approaching and the impact will occur. The impact force can be predicted with (3.5.11).

The head acceleration is

$$a_h(t) = \ddot{x}_h(t) \tag{3.5.12}$$

Its Laplace transform is

$$A_h(s) = X_h(s)s^2 - sx_h(0) - \dot{x}_h(0) \tag{3.5.13}$$

Substituting (3.5.11) into (3.5.9), and then into (3.5.13), gives

$$A_h(s) = \frac{(K_c + C_c s)v_c\left(s^2 + b_{1,r}s + b_{2,r}\right)}{M_h\left(s^4 + q_1 s^3 + q_2 s^2 + q_3 s + q_4\right)} \tag{3.5.14}$$

where $q_2 = \frac{K_r + K_c}{M_r} + \frac{K_h + K_c}{M_h} + \frac{C_c C_r + C_c C_h}{M_r M_h}$, $q_4 = \frac{K_r K_h + K_c K_h + K_r K_c}{M_r M_h}$, $b_{1,r} = \frac{C_r}{M_r}$,

$q_1 = \frac{C_r + C_c}{M_r} + \frac{C_c + C_h}{M_h}$, $q_3 = \frac{K_r(C_c + C_h) + K_c(C_r + C_h) + K_h(C_c + C_r)}{M_r M_h}$ and $b_{2,r} = \frac{K_r}{M_r}$.

Eq. (3.5.14) denotes the human head acceleration in the frequency domain. We can

clearly see that this acceleration is proportional to v_c. Therefore, reducing the impact velocity diminishes the head acceleration. It is not feasible to reduce the human's velocity, so if the head acceleration must be reduced, the robotic manipulator's velocity should be reduced. The influence of M_h, C_h, and K_h on the acceleration will be studied in Section 3.6. The influence of M_r, C_r, K_r, K_c and C_c will be presented in Section 3.7. Manipulator acceleration and deceleration will be studied in Section 3.8.

3.6 Influence of the human characteristics on the impact force and head acceleration

The effects of the human characteristics on f_c and a_h will be studied in this section. The impact force profiles computed from (3.5.11) with different values of M_h, C_h, and K_h and two manipulators are shown in Fig. 3.10. The parameters of the two manipulators are listed in Table 3.1. Note that only the first impact scenario will be investigated in this thesis. In Table 3.1, the distance between the impact point and the revolute joint, l_1 is also presented. The head-neck dynamic parameters are listed in Table 3.2. From this figure, each impact force profile possesses a peak when $t \approx 0.025$ s. The force increases very quickly when $t < 0.025$ s. The force profiles have a step at $t = 0$ and its value is $v_c C_c$. After 0.025 s, the force decreases quickly and then increases in a ramp-like fashion. This is due to the increasing compression of the foam from the approaching head and manipulator as the impact time increases. In reality, the impact force should not increase as much as shown in the figure. A fast impact detection system (for an example

see Deluca *et al.* (2006), where their system detected the impact in 0.004 s) can be installed on the manipulator. After the impact is detected, the manipulator would be commanded to either stop or move backwards. From Fig. 3.10, we can see that the manipulator should be decelerated to rest before 0.05 s to prevent the force further increasing. A simulation of this strategy will be provided in Section 3.8. Comparing the impact force profiles of Simulations 1 and 2, we can see the peak impact force with Manipulator 2 is 15% larger than with the Manipulator 1. This is because the stiffness of both manipulators are similar and the effective mass of Manipulator 2 is much larger (*i.e.* 410%) than that of Manipulator 1. Comparing Simulations 1 and 3, the larger K_h and C_h (70% for K_h and 180% for C_h) causes a 7% larger impact force. It is because a larger force from the neck causes a larger impact force. Comparing Simulations 2 and 4, the 20% larger M_h incurs a 10% larger impact force. The reason is a larger M_h carries higher energy and momentum to impact the foam, resulting in a larger impact force.

Table 3.1. Robotic manipulator parameters.

Manipulator No.	K_r (kN/m)	C_r (Ns/m)	M_r (kg)	l_1 (m)
1	20.0	250	9.6	0.078
2	14.4	111	49.4	0.5

Table 3.2. Head-neck parameters.

Simulation No.	**Manipulator No.**	M_h (kg)	C_h (Ns/m)	K_h (kN/m)
1	1	4.4	130	5.0
2	2	4.4	130	5.0
3	1	4.4	46	3.0
4	2	5.3	130	5.0

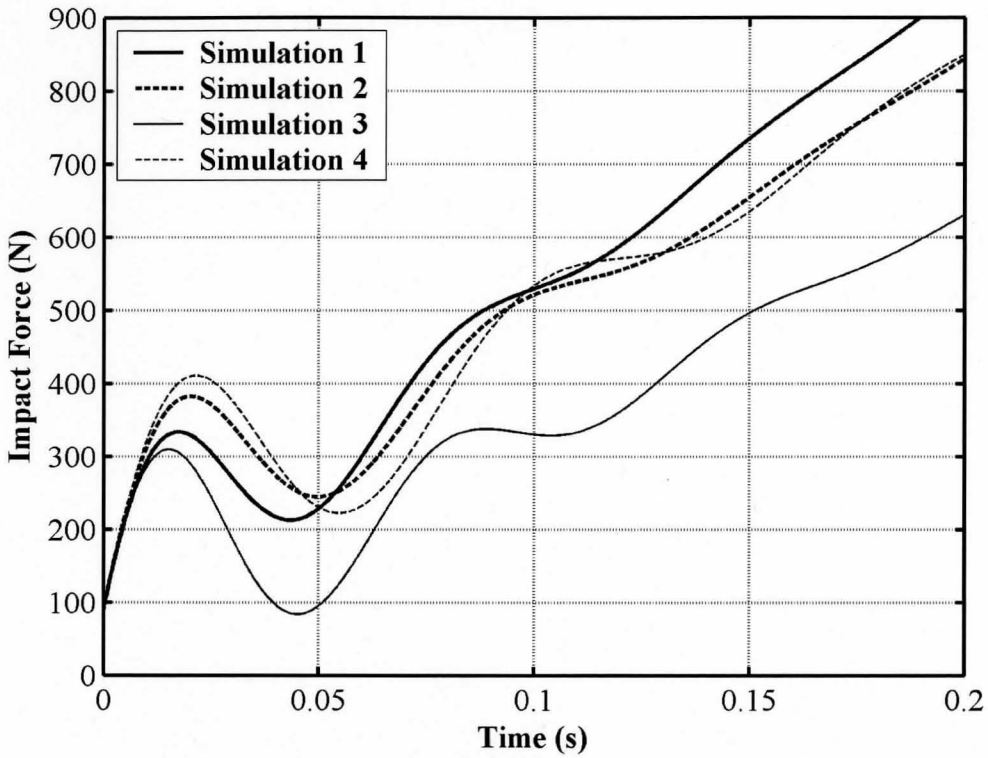


Fig. 3.10. The impact forces computed with (3.5.11) for v_c = 1.25 m/s, K_c=24.25 kN/m, C_c=71.9 Ns/m and the other parameters given in Tables 3.1 and 3.2.

The head acceleration computed from (3.5.14) is shown in Fig. 3.11 for the manipulators dynamic parameters from Table 1 and the head-neck parameters from Table 2. We can see that the head acceleration is bounded and the maximum head acceleration (denoted by an asterisk for Simulation 3 in the figure as an example) is the first peak value in each acceleration profile. Therefore, we will use rather than the entire acceleration profile to analyze the head accelerations. In this thesis, we define the impact period as Δt_{imp}. During this period, the head acceleration is positive. We also can see that the human head acceleration changes rapidly when s. After that, although is still increasing, the acceleration of the head approaches zero. This is because the impact force

is nearly balanced by the force from the neck. Comparing Simulations 1 and 2, the maximum head acceleration with Manipulator 2 is 10% larger than that with Manipulator 1 due to the larger effective mass of Manipulator 2. Comparing Simulations 1 and 3, since the force from resists the motion the head, although the larger causes a larger impact force, is 13% smaller. A 20% larger causes an 8% smaller in Simulation 2 vs. Simulation 4.

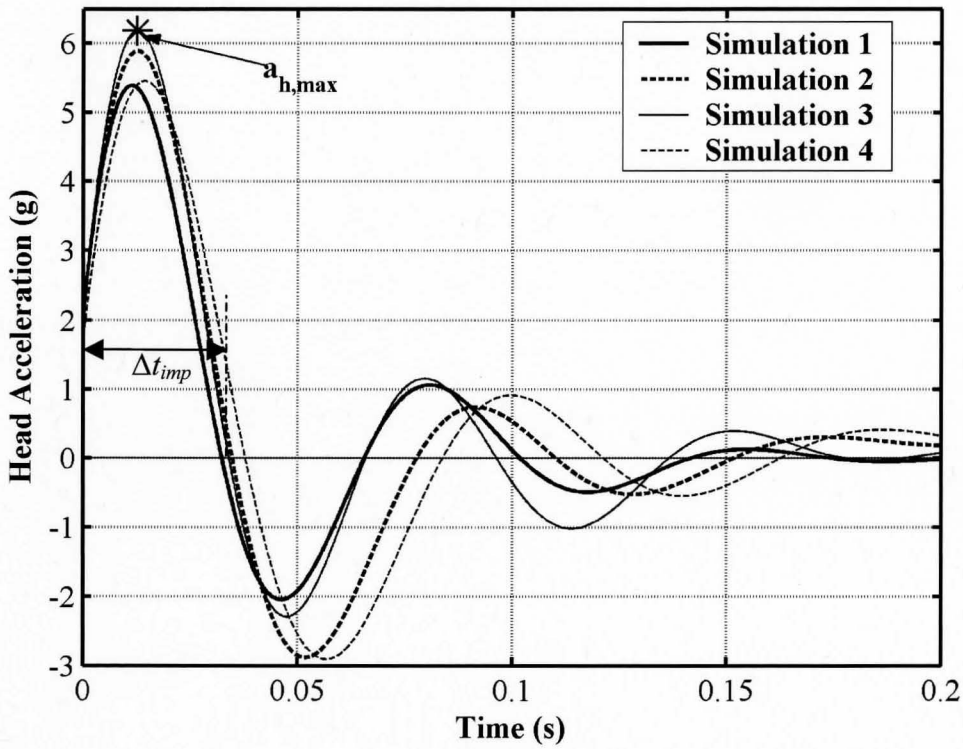


Fig. 3.11. The head acceleration computed with (3.5.14) for the same parameters as in Fig. 3.10.

From the above analysis, we can see that smaller head-neck parameters cause a smaller $a_{h,\max}$ and a larger impact force. If we increase the value of K_h to infinity, a *constrained impact* (a term introduced in Haddadin *et al.* 2009) will happen. A

constrained impact may occur when the head's motion is constrained (*i.e.* $v_h(t) = 0$ m/s), such as when a human stands against a wall. Since the motion of the head is constrained, the head acceleration will be zero. Letting $K_h \to \infty$ to model the constrained impact, (3.5.11) becomes:

$$F_{c,cst}(s) = \frac{(K_c + C_c s)v_c}{s^2\left(1 + \frac{K_c + C_c s}{M_r s^2 + C_r s + K_r}\right)} \tag{3.6.1}$$

where $F_{c,cst}$ is the constrained impact force. $F_{c,cst}$ is proportional to v_c. In Fig. 3.12, the impact force profile in the simulation of a constrained impact is compared with the simulation with the regular head-neck parameters (*i.e.* the head is unconstrained). We can see that the impact force in the constrained impact will be much larger. Normally the manipulator workcell is designed to minimize the possibility of constrained impacts with humans. For this reason, we will not concentrate this impact condition in this thesis.

For unconstrained impacts, the minimum head-neck parameters are $M_h = 4.4$ kg, $C_h = 46$ Ns/m and $K_h = 3$ kN/m. The maximum head-neck parameters are $M_h = 5.3$ kg, $C_h = 130.2$ Ns/m and $K_h = 5$ kN/m. To be conservative, the minimum head-neck parameters will be used for analyzing $a_{h,\max}$ and the maximum head-neck parameters will be used for analyzing f_c.

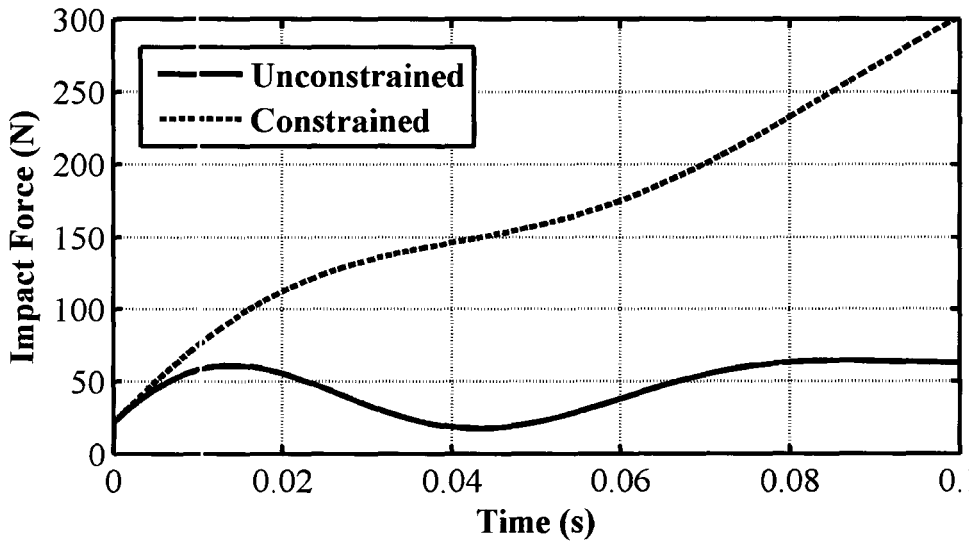


Fig. 3.12. Comparison between simulations of a constrained impact and an unconstrained impact. Parameters of Manipulator 1 and head-neck parameters in Simulation 3 are used for the unconstrained impact. Parameters of Manipulator 1 are used for the constrained impact. Other parameters: v_c = 0.25 m/s, K_c=24.25 kN/m, C_c=71.9 Ns/m.

3.7 Influence of the mechanical characteristics on the head acceleration

3.7.1 Influence of the manipulator's mechanical characteristics

From (3.5.14), the manipulator's mechanical characteristics, M_r, C_r and K_r influence $a_{h,\max}$ during the impact. A light and soft manipulator obviously reduces $a_{h,\max}$. The relationship between M_r and $a_{h,\max}$ is plotted in Fig. 3.13. The relationship between K_r and $a_{h,\max}$ is presented in Fig. 3.14. Note that in this section, the impact force f_c has a similar trend to $a_{h,\max}$, so we only analyze $a_{h,\max}$. Three foam coverings are used in those figures and their parameters are shown in Table 3.3. In the following simulations, $v_c = 1.25$ m/s, as mentioned in Section 3.5.

Table 3.3. Foam parameters.

Foam No.	K_c (kN/m)	C_c (Ns/m)	ξ_c
1	24.3	71.9	0.11
2	13.8	43.8	0.09
3	60.6	113.6	0.11

From Fig. 3.13, when $M_r < 20$ kg, $a_{h,\max}$ is significantly reduced by decreasing the mass. However when $M_r > 40$ kg, the influence of M_r is greatly reduced. This is because the head mass (*i.e.* 4.4 kg) is very small in comparison with the mass of the manipulator. This phenomena can also be seen from the experiment results in Haddadin, Albu-Schäffer, and Hirzinger (2008). When M_r is large a softer foam (*e.g.* Foam 1 and 2) will still make $a_{h,\max} < 10$ g. A stiffer foam (*e.g.* Foam 3) could make the human safety unattainable with a heavy manipulator.

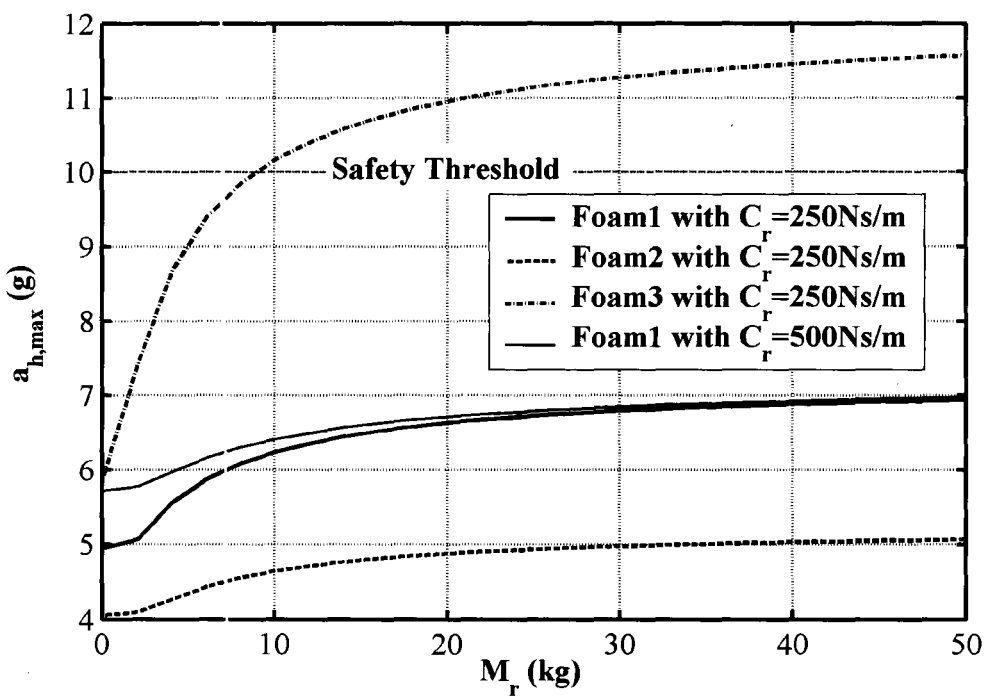


Fig. 3.13. $a_{h,\max}$ vs. M_r for the three foams with $K_r = 20$ kN/m, $M_h = 4.4$ kg, $C_h = 46$ Ns/m and $K_h = 3$ kN/m.

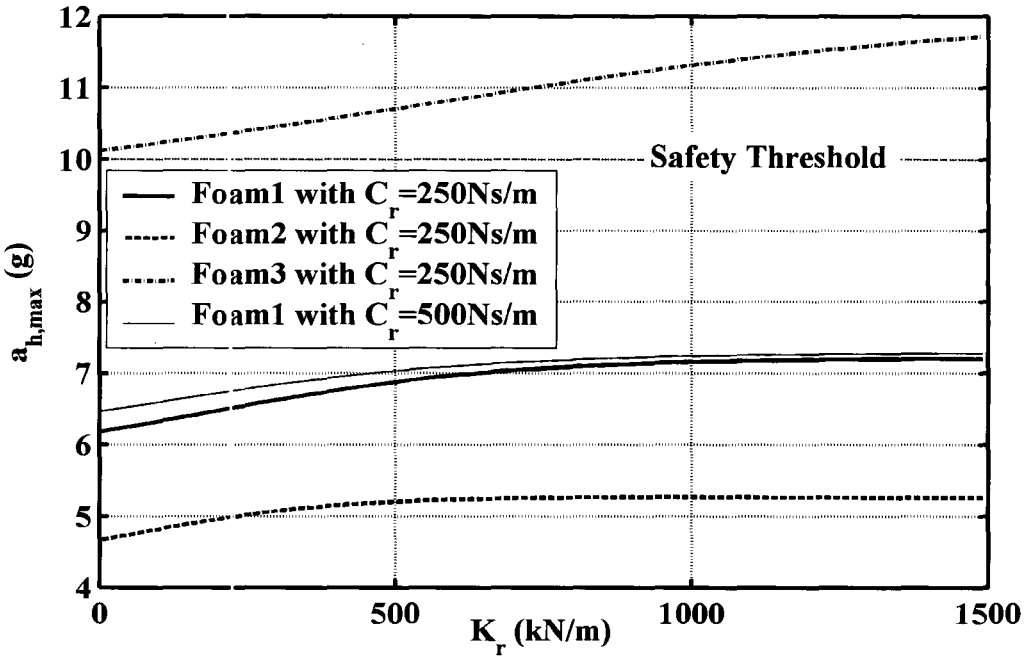


Fig. 3.14. $a_{h,\max}$ vs. K_r for the three foams with $M_r = 9.6$ kg, $M_h = 4.4$ kg, $C_h = 46$ Ns/m and $K_h = 3$ kN/m.

From Fig. 3.14, a small manipulator stiffness also reduces the head acceleration. The reduction effect is less significant than with the reduced M_r. This agrees with the conclusions of Haddadin et al. (2010). Reducing the joint stiffness does not significantly reduce the impact. The reduction effect nearly vanishes when K_r is very high (*i.e.* over 40 times K_c). The effect of C_r is studied by simulating with Foam 1 and $C_r = 500$ Ns/m. The curves in Figs. 3 13 and 3.14 show that this 100% increase in C_r has a minor effect on $a_{h,\max}$.

From the above analysis, to design a human-friendly robotic manipulator, we should prioritize reducing its mass, followed by its stiffness and damping.

3.7.2 The influence of the foam's mechanical characteristics

A soft foam covering will make $a_{h,\max}$ much smaller than the safety threshold. K_c and C_c have a greater influence on $a_{h,\max}$ than any of manipulator parameters as shown in Fig. 3.15. The two robotic manipulators' parameters are listed in Table 3.1. In accordance with p. 236 of Rivin (2003) the range of the damping ratio of the elastomeric foam, ξ_c, is from 0.05 to 0.2. A foam with a higher damping ratio possesses a larger C_c, since $C_c = 2\xi_c\sqrt{M_h K_c}$. The results show that $a_{h,\max}$ is small when K_c is small, regardless of the value of ξ_c. The foam with the large damping ratio has smaller $a_{h,\max}$ since the damping of the foam C_c absorbs the impact energy. Comparing the acceleration curves of Manipulators 1 and 2, Manipulator 1 produces a smaller $a_{h,\max}$. This is because

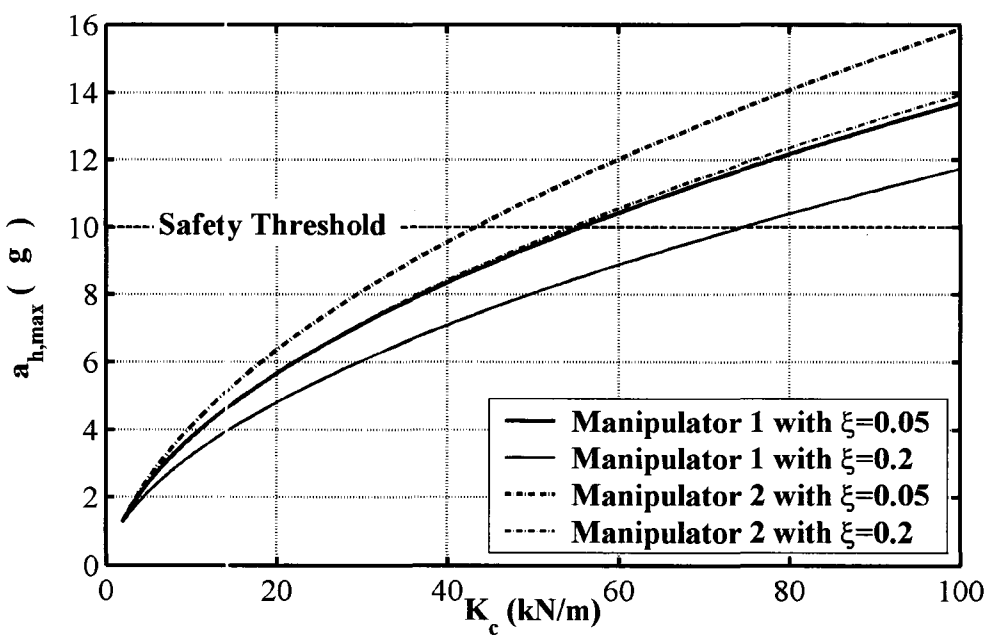


Fig. 3.15. $a_{h,\max}$ vs. K_c with ξ_c=0.05 or 0.2, M_h = 4.4 kg, C_h = 46 Ns/m and K_h = 3 kN/m for Manipulators 1 and 2.

Manipulator 1 has a smaller effective mass. The results also show that a large K_c can cause $a_{h,\max}$ to exceed the safety threshold. Therefore, the safety threshold will provide an upper bound for the stiffness of the foam.

3.7.3. Influence of the location of the impact point

According to the analysis in Section 3.4, when the impact point is near the joint, M_r, C_r and K_r will be large. This tends to increase the head acceleration. However, at the same time, the contact velocity of the manipulator $v_r(0)$ decreases which tends to decrease the acceleration. Therefore, the variation of $a_{h,\max}$ along the manipulator depends on which of these factors is dominant. The relationships between $a_{h,\max}$ and l_1 are plotted in Fig. 3.16. Foam 1 is used with both manipulators. For Manipulator 1, since M_r, C_r and K_r have a larger influence on the $a_{h,\max}$ than v_r does when M_r is small, $a_{h,\max}$ is reduced if l_1 is larger than 0.01 m. When $l_1 < 0.01$ m, since the effective mass is very large, the influence of the reduced v_r is the dominant factor and $a_{h,\max}$ is reduced. If $l_1 = 0$ then the direction of the impact velocity will pass through the joint, and the manipulator can be considered as a rigid body. In this situation, since $v_r = 0$, $a_{h,\max}$ is 2% smaller than that with $l_1 = 0.01$ m. Therefore, the maximum $a_{h,\max}$ along this manipulator appears at $l_1 = 0.01$ m. For Manipulator 2, since M_r is large, v_r always dominates $a_{h,\max}$. Therefore, the worst impact (i.e. with maximum of $a_{h,\max}$) occurs at the distal end of the manipulator. Since the mass of a typical industrial manipulator is relatively large, the

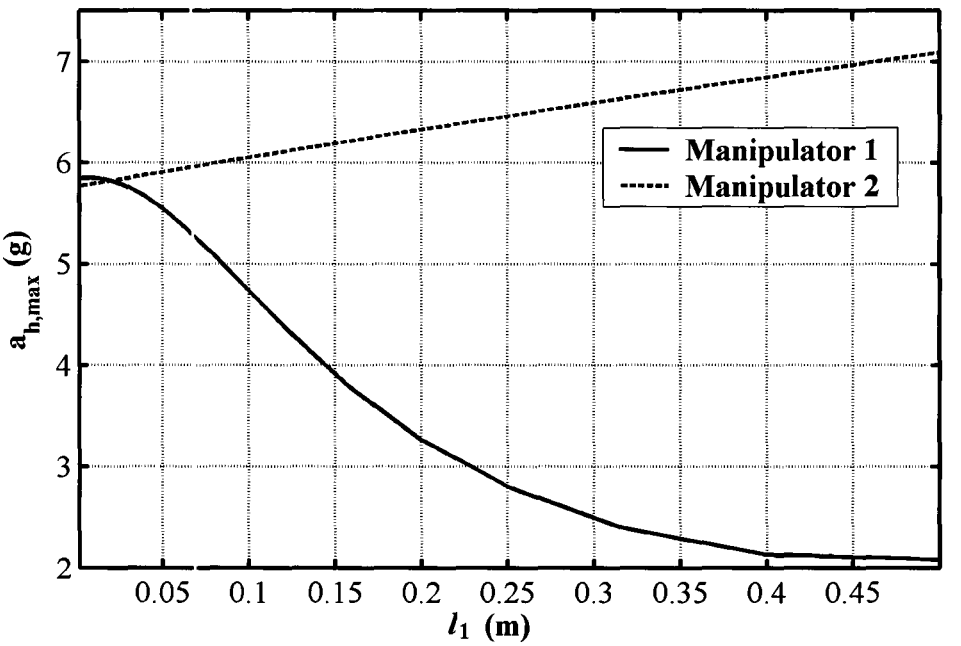


Fig. 3.16. l_1 vs. $a_{h,\max}$ for Manipulators 1 and 2 with K_c=24.25 *kN/m*, C_c=71.9 *Ns/m*, M_h = 4.4 kg, C_h = 46 Ns/m and K_h = 3 kN/m.

worst impact tends to occur at the distal end of the manipulator. For service manipulators, their masses are small, and the worst impacts tend to at the middle of the link of the manipulator. To design the foam covering to enhance safety, the impact point along the manipulator with the worst impact (*i.e.* with the maximum $a_{h,\max}$) must be studied.

3.8 Influence of acceleration and deceleration of the manipulator on the head acceleration and impact force

In Section 3.2, we assume that the manipulator is moving with zero acceleration. However, deceleration and acceleration of the manipulator during the impact also influence the impact force and head acceleration. The influences of deceleration and acceleration are studied with simulations. In Fig. 3.17, simulations with acceleration

(0.51 g from 0 s to 0.05 s and double the linear velocity from 0.25 m/s to 0.5 m/s) and deceleration (-0.51 g from 0 s to 0.05 s and decrease the linear velocity from 0.25 m/s to 0 m/s) are compared with simulations with zero acceleration. These acceleration/deceleration values are conservative since current manipulators can accelerate at several g. The other parameters are the same as Simulation 3 in Table 3.2. From this figure, acceleration increases f_c (5%) and $a_{h,\max}$ (4%), and deceleration decreases f_c (4%) and $a_{h,\max}$ (3%). On the other hand, we can see that the impact force can be effectively reduced by deceleration.

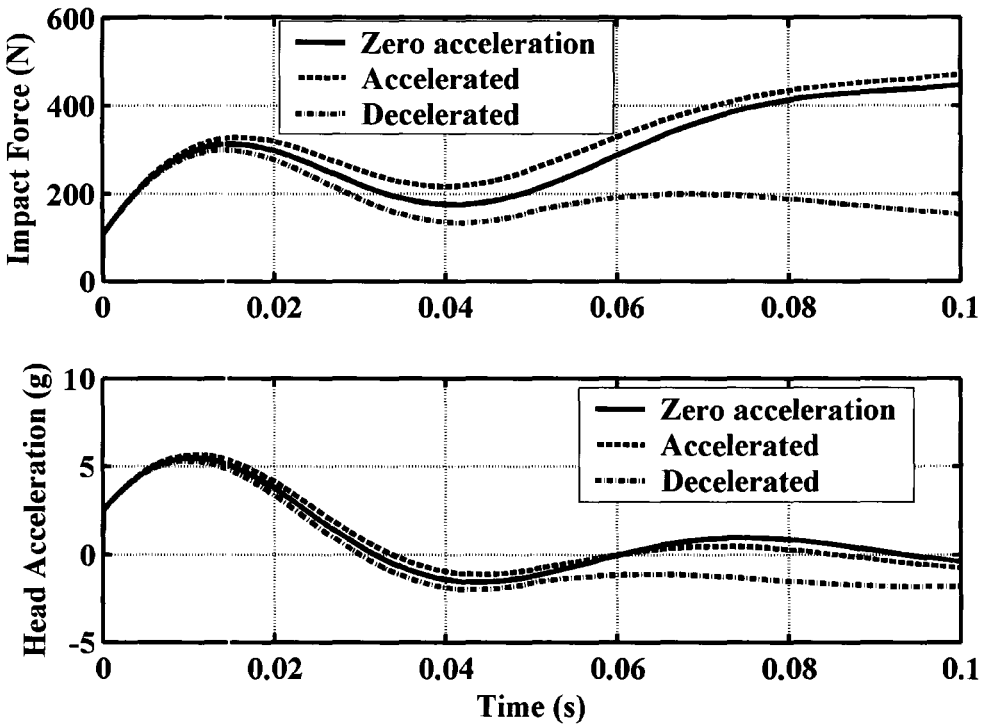


Fig. 3.17. The comparison of impact force and head acceleration for simulations with zero acceleration, accelerated and decelerated manipulator.

3.9 Foam covering design procedure

3.9.1 Upper stiffness-damping boundary curve based on the safety criteria

Satisfying the safe criteria will give an upper boundary for the foam stiffness and damping. As mentioned in Section 2, we will utilize the arbitrarily chosen acceleration-based safety threshold to explain how to obtain the upper stiffness-damping boundary curve. Note that this procedure is suitable for impact-force-based and/or head-acceleration-based criteria. Any further development of safety criteria can be incorporated into our design methodology. The upper stiffness-damping boundary curve must be satisfied for all possible human and manipulator velocities over their respective ranges. These ranges are application dependent. As mentioned in Section 3.5, we limit the impact velocity to $v_c \leq 1.25$ m/s. Given the values of the safety threshold, human velocity, manipulator velocity, and model parameters, the upper boundary of K_c and C_c can be computed rapidly using (3.5.14) and the secant method. Note that if an impact-force-based safety criterion is used, (3.5.11) will be used. In this section, the parameters of the two manipulators in Table 3.1 will be used to demonstrate the design process. To be conservative, the minimum head-neck parameters are used. The corresponding boundary for Manipulator 1 is the upper curve in Fig. 3.18. The boundary for Manipulator 2 is presented in Fig. 3.19. If the stiffness and damping of the foam covering is located below the curve, as it is for Foam 1 and Foam 2 for Manipulators 1 and 2, the safety of the human is enhanced.

3.9.2 Lower stiffness boundary curve due to thickness constraint

While a softer foam reduces $a_{h,\max}$, it can also more easily become fully compressed, thereby no longer providing a force reduction effect. This can be overcome by using a thicker layer of foam, but at some point the covering becomes too thick to be practical. Therefore the lower boundary of the foam stiffness and damping should satisfy a thickness constraint. Normally, the stiffness of the foam will be close to a constant if the compressed depth is smaller than a certain threshold defined here as λ. The remaining thickness is denoted as T_{rem}. The uncompressed foam thickness is then given by:

$$T_u = \lambda + T_{rem} \tag{3.9.1}$$

If the compressed depth is over λ, the actual stiffness will be greater than K_c and the results from our model will no longer be reliable. Therefore λ will determine the minimum bound for the foam stiffness. Substituting (3.5.11) into (3.5.7), and then dividing by $K_c + C_c s$, the compressed depth in the frequency domain is:

$$\varepsilon(s) = \frac{v_c\left(s^2 + b_{1,r}s + b_{2,r}\right)\left(s^2 + b_{1,h}s + b_{2,h}\right)}{s^2\left(s^4 + a_1 s^3 + a_2 s^2 + a_3 s + a_4\right)} \tag{3.9.2}$$

The thickness constraint must ensure that the compressed depth is smaller than λ. We denote the maximum compressed depth as $\varepsilon_{\max}$. The thickness constraint then is:

$$\varepsilon_{\max} < \lambda \tag{3.9.3}$$

This constraint must be satisfied for the maximum value of v_c. Since $\varepsilon_{\max}$ is mathematically intractable to obtain from (3.9.2) directly, the secant method is used to

find the lower stiffness and damping boundary. Since the influence of the head-neck parameters on the impact force and the compressed depth are nearly identical, the maximum head-neck parameters are used for a conservative result. In Fig. 3.18, the lower boundary for Manipulator 1 with $\lambda = 0.02$ m is presented as the lower curve. The boundary for Manipulator 2 with $\lambda = 0.02$ m is presented in Fig. 3.19. If the stiffness and damping of the foam covering are above the lower boundary, the thickness constraint is satisfied. As shown in the figures, Foam 1 satisfies the thickness constraint for Manipulators 1 and 2. Foam 2 does not satisfy the thickness constraint. It is almost on the boundary for Manipulator 1 and far below the boundary for Manipulator 2. Comparing Fig. 18 and 19, we can see that the stiffness and damping ranges of foam coverings between the upper and lower boundary for Manipulator 2 are smaller than with Manipulator 1.

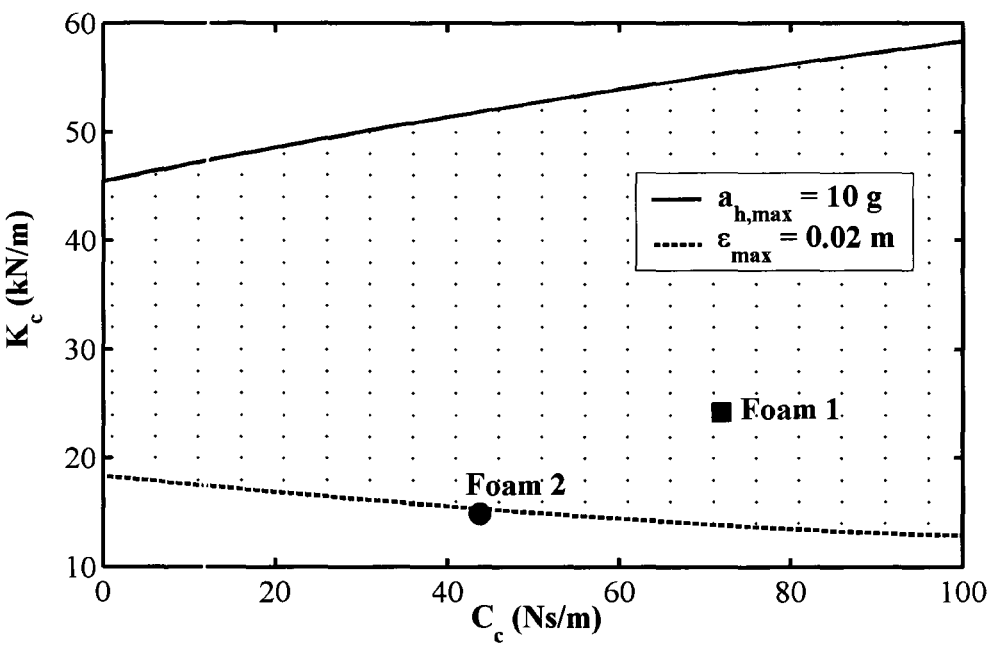


Fig. 3.18. The upper and lower boundaries of K_c and C_c for Manipulator 1 with safety threshold $a_{h,max} < 10$ g and the thickness constraint $\varepsilon_{max} < 0.02$ m.

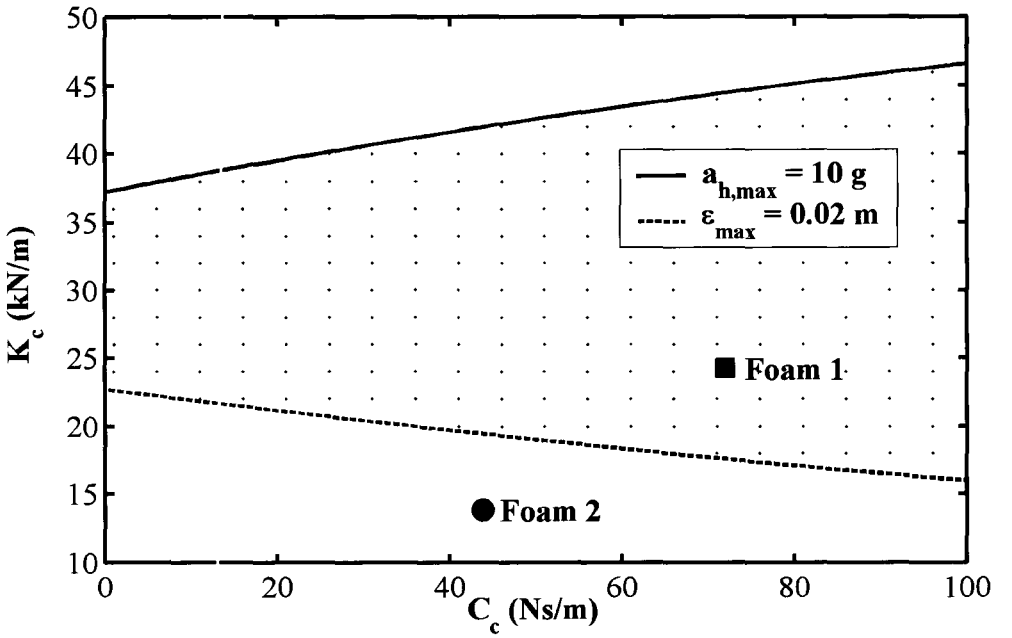


Fig. 3.19. The upper and lower boundaries of K_c and C_c for Manipulator 2 with safety threshold $a_{h,max} < 10$ g and the thickness constraint $\varepsilon_{max} < 0.02$ m.

Note that when the manipulator velocity is increased, the lower boundary will be raised. At the same time, the upper boundary will be lowered. Therefore, depending on the velocity and dynamic parameters of the manipulator, the stiffness-damping range can be reduced until there is no solution. If this occurs, the manipulator velocity must be reduced to make the range a nonempty set. Based on the above analysis, the procedure of designing foam coverings for manipulators is summarized in Fig. 3.20. If the safety criterion is impact-force-based, (3.5.11) and the maximum head-neck parameters are used to computer the upper boundary. If the safety criterion is head-acceleration-based, (3.5.14) and the minimum head-neck parameters are used for the upper boundary.

This procedure can also be used to consider the constrained impact. Then, (3.6.1) will be used to compute the upper boundary. The compressed depth in the constrained

impact can be obtained by dividing (3.6.1) with $K_c + C_c s$ and re-organizing the result:

$$\varepsilon_{cst}(s) = \frac{v_c\left(M_r s^2 + C_r s + K_r\right)}{s^2\left[M_r s^2 + \left(C_r + C_c\right)s + K_r + K_c\right]} \tag{3.9.4}$$

where $\varepsilon_{cst}(s)$ is the compressed depth in the constrained impact. With (3.9.4) and the secant method, the lower boundary for the constrained impact may be obtained.

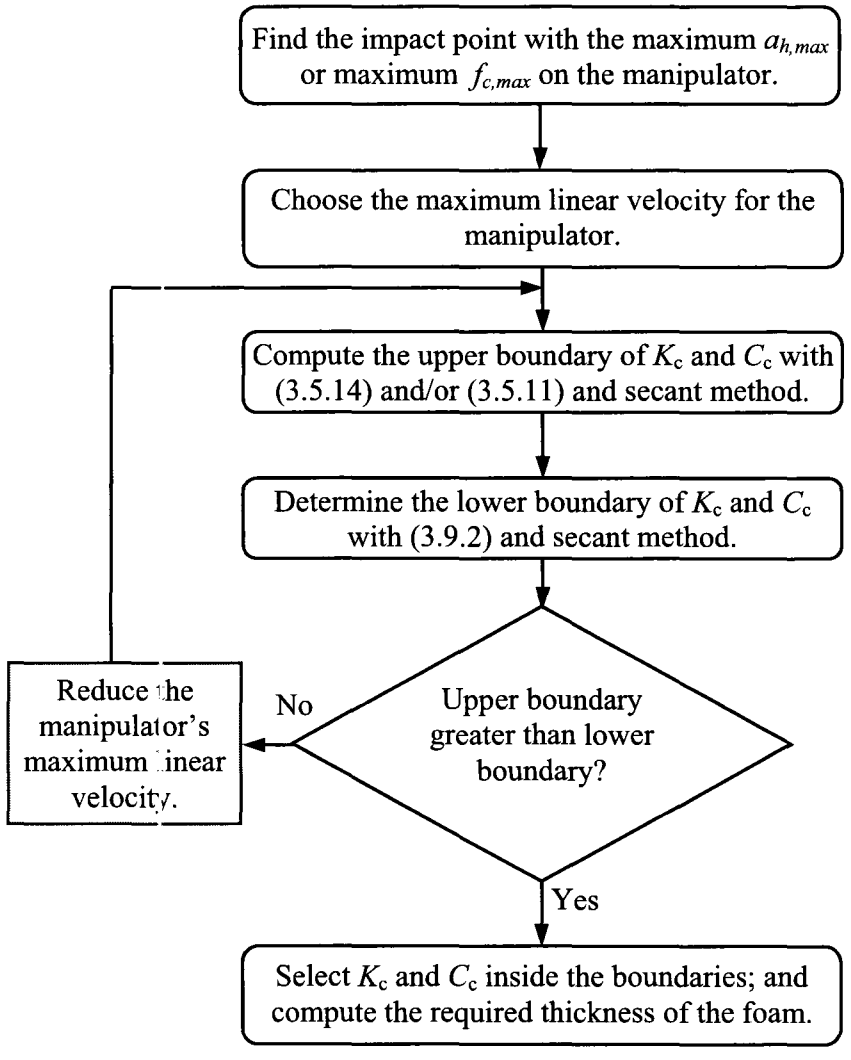


Fig. 3.20. Flowchart of the foam covering design procedure.

3.10 Conclusions

In this chapter, an improved impact force model and a novel model-based procedure for the design of elastomeric foam coverings for human-friendly robotic manipulators have been introduced. The novel impact model incorporates the configuration-dependent manipulator dynamics and the previously neglected coupling between the human head and torso. This model is also applicable for constrained impacts. Based on this model, the important parameters (*i.e.* K_h, C_h, M_h, K_r, C_r, M_r, C_c, K_c, and l_1) that influence the head acceleration and impact force were investigated. Influences of the manipulator acceleration/deceleration are also studied. Finally, we proposed a model-based procedure to properly select the foam in accordance with the safety criterion and the foam thickness constraint.

Chapter 4

Human-manipulator impact experiments

4.1 Introduction

In Chapter 3, the human-manipulator impact dynamics were modelled. With this model, the key parameters for designing an elastomeric foam-covered manipulator were studied. In this chapter, human-manipulator impact experiments will be presented to verify the impact model. An apparatus used to simulate the human head-neck-torso system will be introduced first. Next the impact experiments will be conducted with a direct-drive manipulator and the head-neck-torso apparatus. The experiments will be performed with different foam covering and different velocities of the head and manipulator. This is followed by impact experiments with a Puma 560 industrial manipulator. Finally, the conclusions from the impact experiments will be drawn.

4.2 Design of an apparatus to simulate the head-neck-torso system

During the human-manipulator impact experiments, an apparatus is required to simulate the dynamics of the human head-neck-torso system. The apparatus is comprised of a pneumatic system, a mass-spring mechanism and a linear slide, as shown in Fig. 4.1. The pneumatic system consists of a servo valve (Festo model MPYE-5-1/8), a rodless cylinder (Festo model DGPL-25-600) and a linear encoder (RSF Electronik model MSA6704 with a 0.0001 m resolution). This cylinder is closed-loop controlled by a PC and its moving range is from 0 to 0.32 m. The detailed circuit of the pneumatic system is

given in Chapter 3 of Ning (2004). In the impact experiments, the piston in this cylinder is controlled to move forward with a constant velocity. The control system is presented in Section A.3 of Appendix A. A linear slide is fixed on the piston to act as the moving human torso. The mechanism simulating the head and neck slides on the linear slide using a linear ball bearing (see Fig. 4.2). The mechanism has a 4.4 kg moving mass to simulate the human head. The half-cylindrical aluminum part with a 0.09 m radius is also employed to approximate the curvature of the front of the human head (Law 1993). The mechanism is connected to the linear slide with springs that emulate the human neck. The total stiffness of the springs is 4 kN/m (i.e. the mean value of K_h for females and males). During the experiments, an accelerometer (MMA6270 from Freescale Semiconductor Inc.) is fixed on the head mechanism and measures the head acceleration a_h. It can measure accelerations up to 8.5 g at the sampling frequency of 1000 Hz.

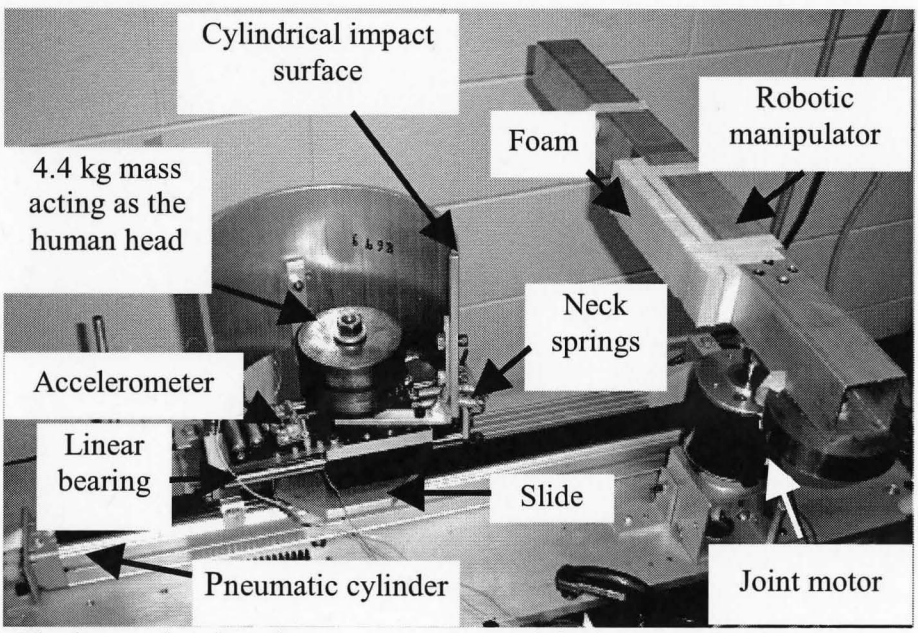


Fig. 4.1. The human head-neck torso apparatus and the human-robot impact experimental setup with the direct-drive manipulator.

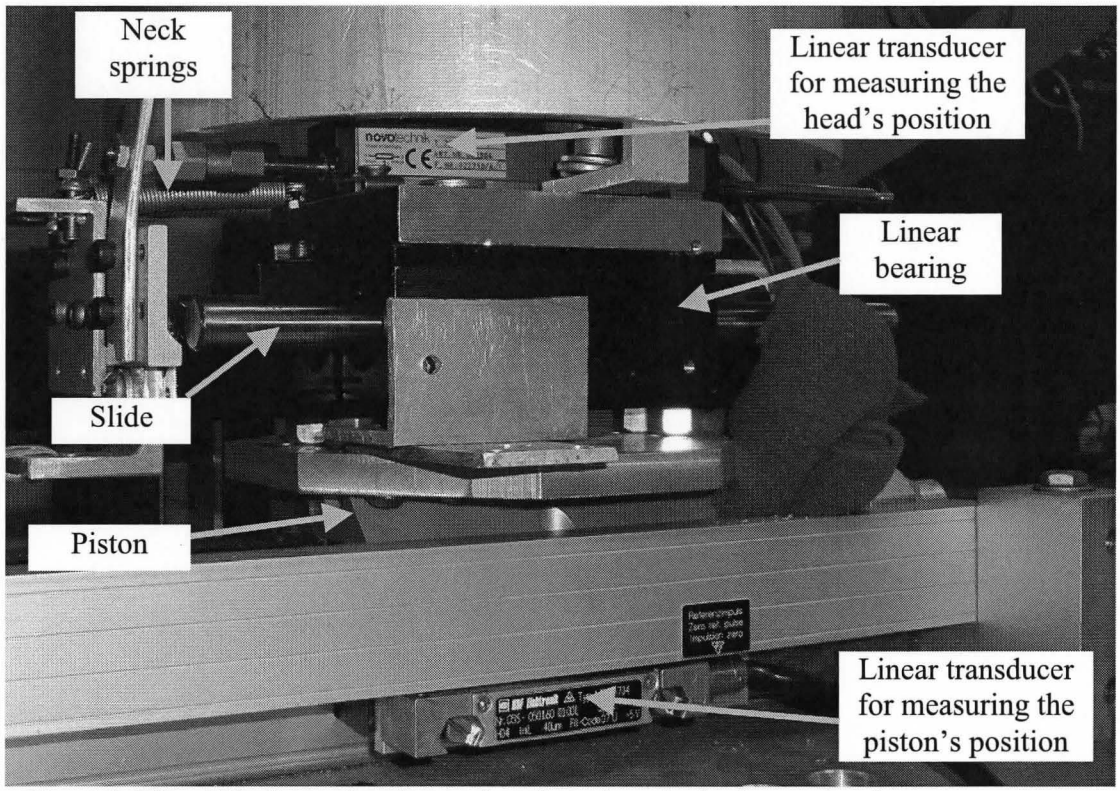


Fig. 4.2. The slide, linear bearing and neck springs in the human-neck-torso apparatus.

4.3 The impact experiments with the direct-drive manipulator

4.3.1 Experimental procedure

The first experimental setup is shown in Fig. 4.1 with the direct-drive manipulator and the apparatus simulating the head-neck-torso system. The manipulator's mechanical design and control system design (*i.e.* PD+FF control) are presented in Section A.2 of Appendix A. As discussed in Section 3.4, its effective mass, damping and stiffness can be computed with:

$$M_r = m_{11} / l_1^2, \tag{4.3.1}$$

$$K_r = \frac{K_{t,dd} K_{amp} K_{p,dd}}{l_1^2} \text{ and} \tag{4.3.2}$$

$$C_r = \frac{K_{t,dd} K_{amp} K_{d,dd}}{l_1^2}, \tag{4.3.3}$$

where $K_{p,dd}$ and $K_{d,dd}$ are the proportional and derivative gains in the direct-drive manipulator's controller; $K_{t,dd} = 0.28$ Nm/A is the summed motor torque constant; $K_{amp} = 1$ A/V is the amplifier gain; $m_{11} = 0.059$ kgm^2 is the moment of inertia of the manipulator and $l_1 = 0.078$ m is the distance between the impact point and the rotation centre of the joint. The detailed derivations of (4.3.2) and (4.3.3) are presented in Section A.2 of Appendix A. With (4.3.1), $M_r = m_{11}/l_1^2 = 0.059/0.078^2 = 9.6$ kg. From (4.3.2) and (4.3.3), with different $K_{p,dd}$ and $K_{d,dd}$, we can have the different C_r and K_r in the impact experiments. If we want $K_r = 20$ kN/m and $C_r = 250$ Ns/m, (4.3.2) is rewritten as:

$$K_{p,dd} = \frac{K_r l_1^2}{K_{t,dd} K_{amp}} = \frac{20{,}000 \times 0.078^2}{0.2807 \times 1} = 433.8 \text{ V}$$

and (4.3.3) is rewritten as

$$K_{d,dd} = \frac{C_r l_1^2}{K_{t,dd} K_{amp}} = \frac{250 \times 0.078^2}{0.2807 \times 1} = 5.4 \text{ V/s}.$$

Two foams (Foam 1 and Foam 2 in Table 3.2) will be attached to the surface of the manipulator. The thickness of both foams is $T_u = 0.0254$ m and $\lambda = 0.02$ m. A linear transducer (T50 with 0.0001 m resolution from Novotechnik. Inc.) is used to measure the

positions of the head during the impact. The angular positions of the manipulator, θ_{J1} can be measured with its rotary encoder. By setting $\theta_{J1,init} = \theta_{J1}$ when the head and the manipulator make first contact, the linear positions of the manipulator can be computed from its angular positions with:

$$x_r = l_1 \sin\left(\theta_{J1} - \theta_{J1,init}\right) \tag{4.3.4}$$

With the positions of the head and manipulator, the compressed depth can be calculated. The damping factor of the human neck is not included. In our experiments, the human head and/or the robotic manipulator will be driven to impact each other at different velocities.

4.3.2 Experiments results and discussion

Table 4.1 presents a comparison of the experiment results and simulation results obtained with the directly-drive manipulator. Two stiffness values (K_r = 20 kN/m and 10 kN/m) of the manipulator are used for the experiments with Foam 1. Two damping values (C_r = 250 Ns/m and 500 Ns/m) of the manipulator are used for the experiments with Foam 2. With those parameters the effects of the stiffness and damping of the manipulator on the impact can be revealed. From the table, the maximum error of $a_{h,\max}$ computed with (3.5.14) relative to the experimental results is 6% and the average relative error is 4%; while the maximum error of $\varepsilon_{\max}$ from (3.9.2) relative to the experimental results is 10% and the average relative error is 7%. Comparing the results of $v_c = 1.25$ m/s to that of $v_c = 0.4$ m/s, we can clearly see that $a_{h,\max}$ and $\varepsilon_{\max}$ are proportional to the impact velocity. The data also shows that identical impact velocities

cause nearly identical head accelerations if the same foam, head and manipulator's dynamics parameters are used. It is also apparent that the manipulator's parameters K_r and C_r have minor effects on $a_{h,max}$ and ε_{max}. The halving of K_r (from 20 kN/m to 10 kN/m) only reduces $a_{h,max}$ by 0.5% and has no obvious effect on ε_{max}. The halving value of C_r (from 500 Ns/m to 250 Ns/m) has no obvious effect on $a_{h,max}$ and ε_{max}. We also see the softer foam (Foam 2: K_c=13.8 kN/m and C_c=43.8 kN/m) reduces $a_{h,max}$ by 26% but increases ε_{max} by 41 % in comparison with Foam 1 (K_c=24.3 kN/m and C_c=71.9 kN/m). We can conclude that it is more effective to use foam coverings to reduce $a_{h,max}$ and enhance human safety than to decrease the manipulator's stiffness and damping. This agrees with observations from Section 3.7.

The time domain plots of six typical experiments from this table are also presented in Figs. 4.3 – 4.6. In these figures, the data are filtered by using a 4th order Butterworth low-pass filter with a 200 Hz cut-off frequency. The cut-off frequency was selected to prevent aliasing. The simulation and experimental results for the maximum head acceleration and compressed depth agree well. The most notable difference is that the impact periods in the experiments are longer than in the simulations. This was caused by the hysteresis effect of the foam. When the foam is releasing, the stiffness is reduced compared to its value when the foam is compressing (Van der Schuur *et al.* 2004). This lower stiffness produced the lower force. Furthermore, the lower force reduced the head acceleration causing the impact period to increase. In Fig. 4.3b and 4.4b, the compressed depth curve with v_r = 0.4 m/s and v_h = 0.0 m/s is much smoother than the curve with v_r =

0.0 m/s and v_h = -0.4 m/s. Since the head is connected to the linear slide with springs, when the head is moving, its acceleration is slightly oscillatory (see Section A.3 of Appendix A). This causes the oscillations in the acceleration. This oscillation phenomenon is not obvious in the figures with $v_c = 1.25$ m/s. This is because the acceleration oscillations are small in comparison with the large head acceleration and the scales of the figures are also larger.

Table 4.1. Impact experimental data for the direct-drive manipulator

Foam Parameters		Manipulator Parameters		Impact Velocities		Experimental Result			Simulation Result		
K_c (kN/m)	C_c (Ns/m)	K_r (kN/m)	C_r (Ns/m)	v_r (m/s)	v_h (m/s)	$a_{h,max}$ (g)	Δt_{imp} (s)	ε_{max} (m)	$a_{h,max}$ (g)	Δt_{imp} (s)	ε_{max} (m)
24.3	71.1	20.0	250	0.0	-0.4	2.05	0.038	0.0035	2.10	0.033	0.0039
						2.11	0.043	0.0038			
						1.98	0.039	0.0033			
				0.40	0.0	2.04	0.041	0.0033			
						2.11	0.041	0.0037			
						2.09	0.042	0.0038			
				0.25	-1.0	6.50	0.040	0.0112	6.56	0.033	0.0118
						6.57	0.036	0.0113			
						6.49	0.045	0.0115			
		10.0	250	0.0	-0.4	2.10	0.036	0.0035	2.09	0.034	0.0039
						2.00	0.047	0.0031			
						2.02	0.049	0.0033			
				0.40	0.0	2.00	0.037	0.0038			
						2.05	0.036	0.0038			
						2.00	0.033	0.0031			
				0.25	-1.0	6.39	0.042	0.0113	6.54	0.034	0.0118
						6.33	0.045	0.0112			
						6.44	0.038	0.0118			
13.8	43.8	20.0	250	0.0	-0.4	1.57	0.048	0.0050	1.55	0.042	0.0055
						1.55	0.050	0.0052			
						1.53	0.046	0.0053			
				0.40	0.0	1.46	0.049	0.0055			
						1.53	0.051	0.0051			
						1.51	0.051	0.0053			
				0.25	-1.0	4.74	0.048	0.0160	4.85	0.042	0.0171
						4.95	0.047	0.0167			
						4.82	0.047	0.0159			
		20.0	500	0.0	-0.4	1.48	0.048	0.0051	1.55	0.041	0.0055
						1.49	0.046	0.0051			
						1.50	0.049	0.0053			
				0.40	0.0	1.51	0.049	0.0051			
						1.58	0.048	0.0054			
						1.52	0.048	0.0050			
				0.25	-1.0	4.86	0.049	0.0168	4.87	0.041	0.0169
						4.78	0.047	0.0164			
						4.80	0.047	0.0166			

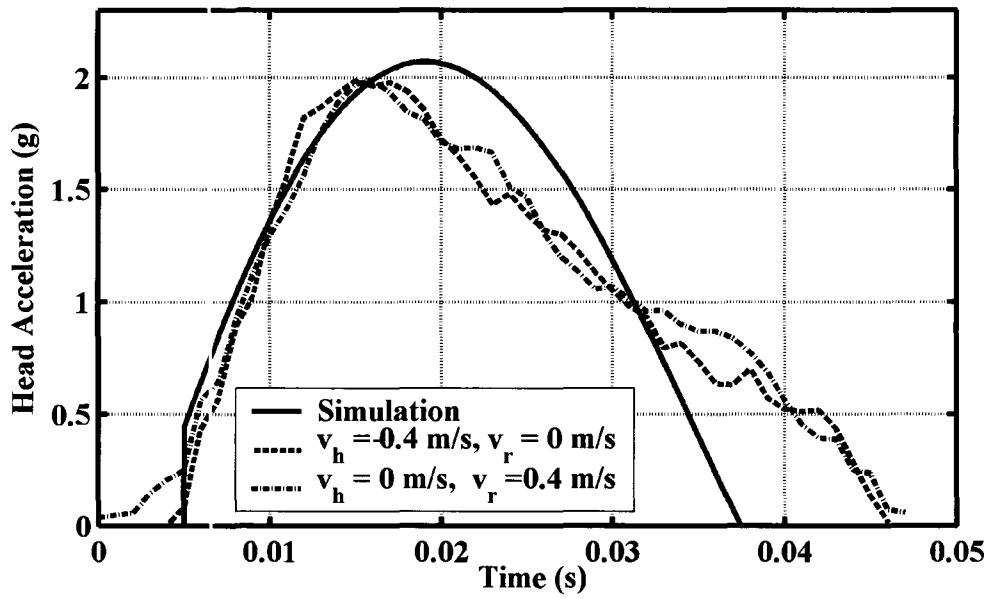


Fig. 4.3a. The head accelerations for the simulation and impact experiments with Foam 1 and with $v_c = 0.4$ m/s, K_r =20 kN/m and C_r=250 Ns/m.

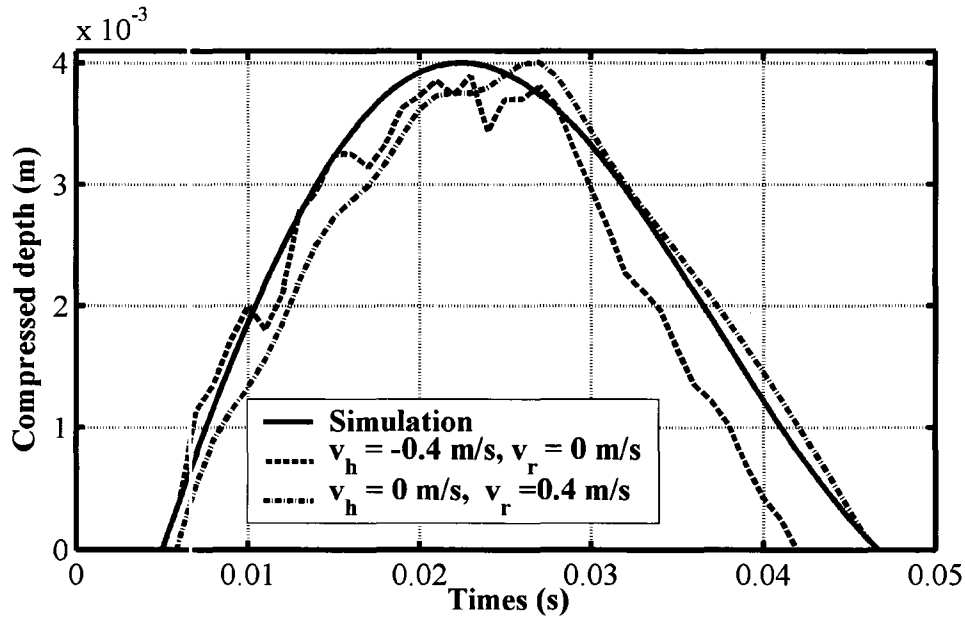


Fig. 4.3b. The compressed depth for the simulation and impact experiments with Foam 1 and with $v_c = 0.4$ m/s, K_r =20 kN/m and C_r=250 Ns/m.

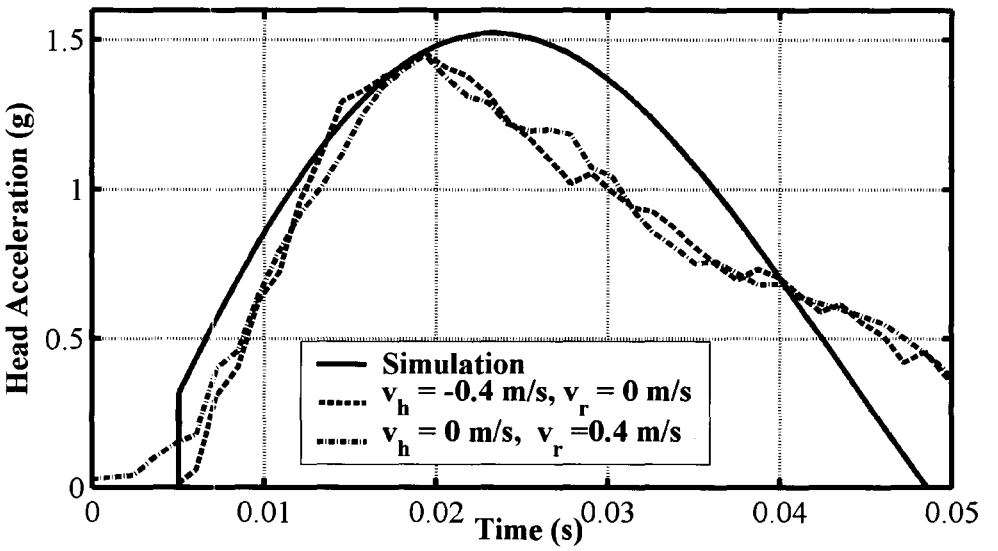


Fig. 4.4a. The head accelerations for the simulation and impact experiments with Foam 2 and with $v_c = 0.4$ m/s, K_r =20 kN/m and C_r=250 Ns/m.

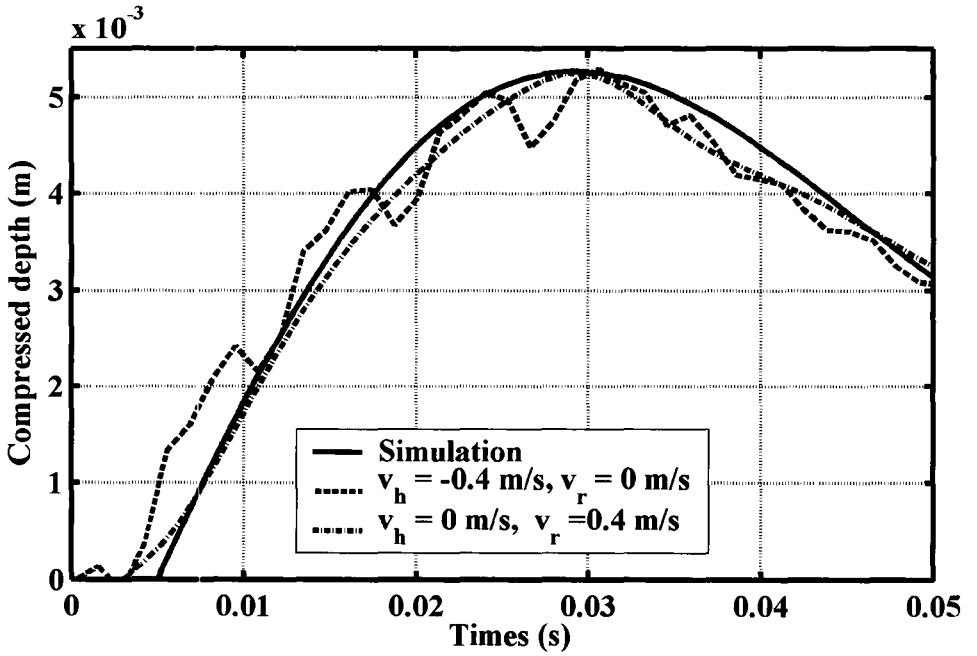


Fig. 4.4b. The compressed depth for the simulation and impact experiments with Foam 2 and with $v_c = 0.4$ m/s, K_r =20 kN/m and C_r =250 Ns/m.

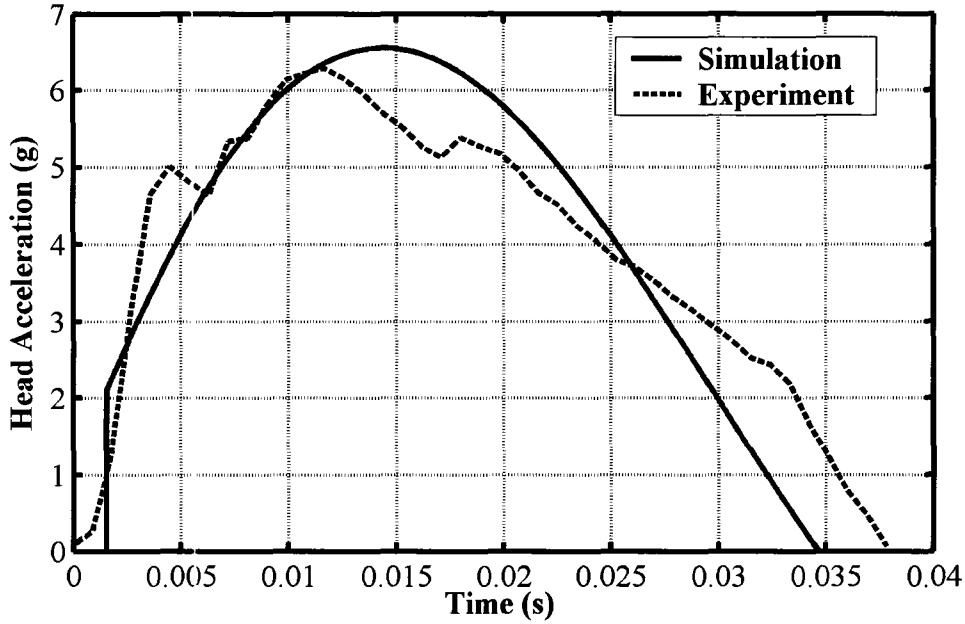


Fig. 4.5a. The head acceleration for the simulation and impact experiments with Foam 1 and with $v_c = 1.25$ m/s, K_r =20 kN/m and C_r=250 Ns/m.

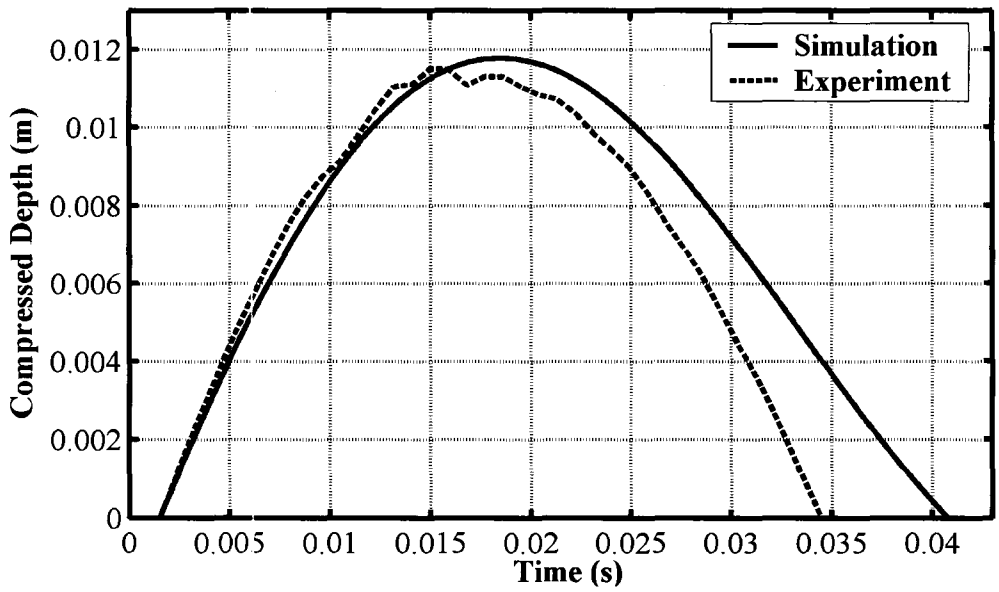


Fig. 4.5b. The compressed depth for the simulation and impact experiments with Foam 1 and with $v_c = 1.25$ m/s, K_r =20 kN/m and C_r=250 Ns/m.

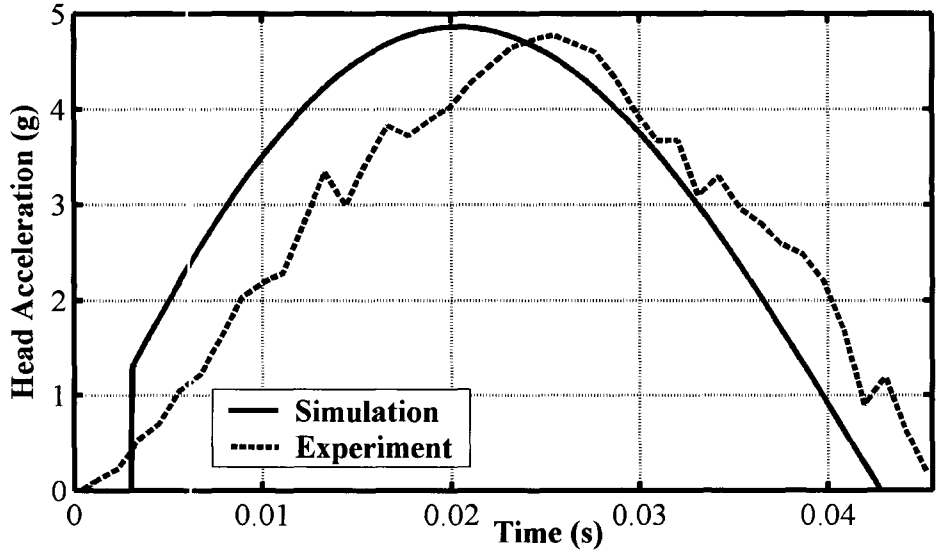


Fig. 4.6a. The head accelerations for the simulation and impact experiments with Foam 2 and with $v_c = 1.25$ m/s, K_r =20 kN/m and C_r=250 Ns/m.

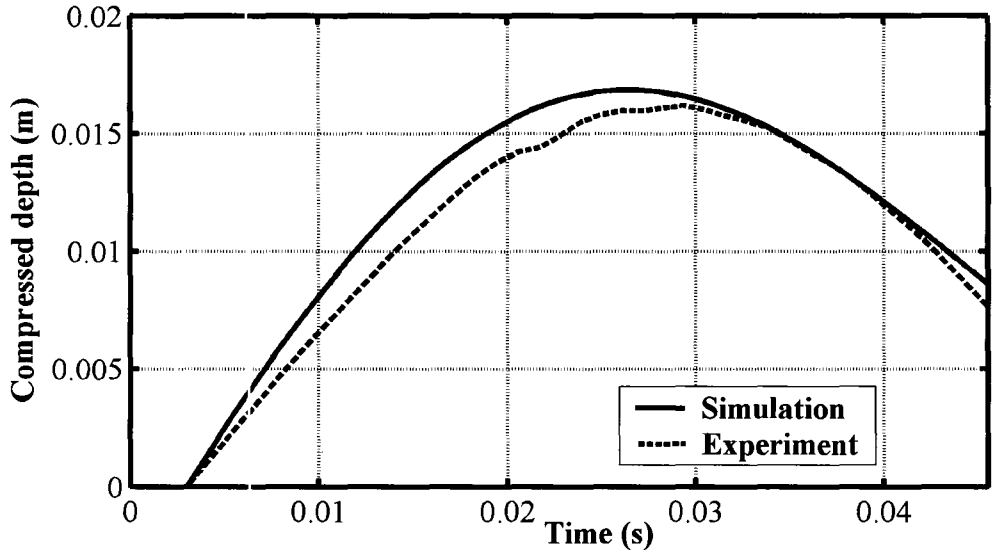


Fig. 4.6b. The compressed depth for the simulation and impact experiments with Foam 2 and with $v_c = 1.25$ m/s, K_r =20 kN/m and C_r=250 Ns/m.

4.4 The impact experiments with a Puma 560 manipulator

4.4.1 Experimental procedure

Experiments were also performed with a Puma 560 robotic manipulator. The setup is shown in Fig. 4.7. This manipulator's parameters are from Bone and Elbestawi (1991) and are listed under Manipulator 2 in Table 3.1. As in the foam design procedure presented in section 3.7, only Foam 1 is used in the experimental setup. During the experiments, the first joint of this manipulator is driven at a specified velocity to impact the head-neck-torso apparatus described in Section 4.1. A linear transducer (T100 from Novotechnik Inc. with a 0.0001 m resolution) is used to measure the displacement of the manipulator during the impact.

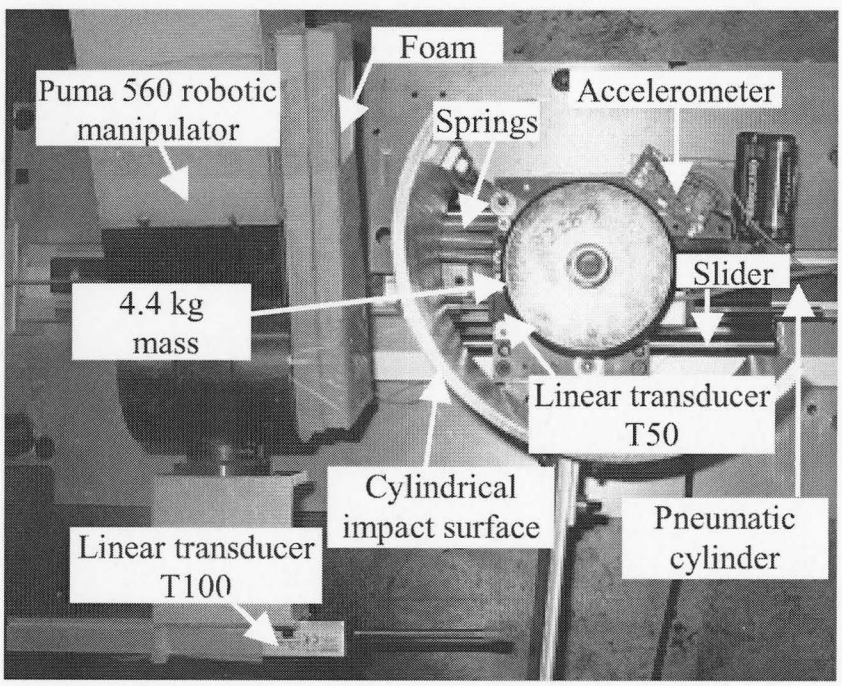


Fig. 4.7 The impact experimental setup with a Puma 560 manipulator

4.4.2 Experimental results and discussion

The experimental results with Foam 1 are presented in Table 4.2. From this table, the $a_{h,\max}$ computed from (3.5.14) are 6% larger than the experimental results on average and the maximum relative error is 8%. The $\varepsilon_{\max}$ from (3.9.2) are 9% smaller than the experimental results on average and the maximum relative error is 12%. Compared with the experimental results for the direct drive manipulator, we can see $a_{h,\max}$ are increased 18% for $v_c = 0.4$ m/s on average and 13% for $v_c = 1.25$ m/s on average. $\varepsilon_{\max}$ are increased 44% on average and 14% on average. This is because the effective mass of this manipulator is much larger than the direct drive manipulator. Therefore, we can conclude decreasing the mass of the manipulator is more effective to reduce $a_{h,\max}$ and $\varepsilon_{\max}$ than decreasing its stiffness and damping. The time domain plots of three typical experiment results from Table 4.2 are presented in Figs. 4.8a, 4.8b, 4.9a and 4.9b. The experiment results and the simulations agree well. The head acceleration oscillations in Fig. 4.8a and 4.9a are slightly more severe than observed with the direct-drive manipulator. This is because the Puma manipulator also oscillates slightly during its movement. This increases the head acceleration oscillations during the impact.

Table 4.2 Impact experimental data for the PUMA 560 manipulator

Impact Velocity		**Simulation Results**			**Experimental Results**		
v_r (m/s)	v_h (m/s)	$a_{h,max}$ (g)	Δt_{imp} (s)	ε_{max} (m)	$a_{h,max}$ (g)	Δt_{imp} (s)	ε_{max} (m)
0.0	-0.4	2.25	0.043	0.0042	2.47	0.035	0.0047
		2.27	0.042	0.0042			
		2.29	0.045	0.0044			
0.4	0.0	2.29	0.045	0.0044			
		2.25	0.044	0.0042			
		2.27	0.043	0.0043			
0.25	-1.0	7.18	0.041	0.0132	7.38	0.035	0.0137
		7.08	0.042	0.0128			
		7.20	0.045	0.0130			

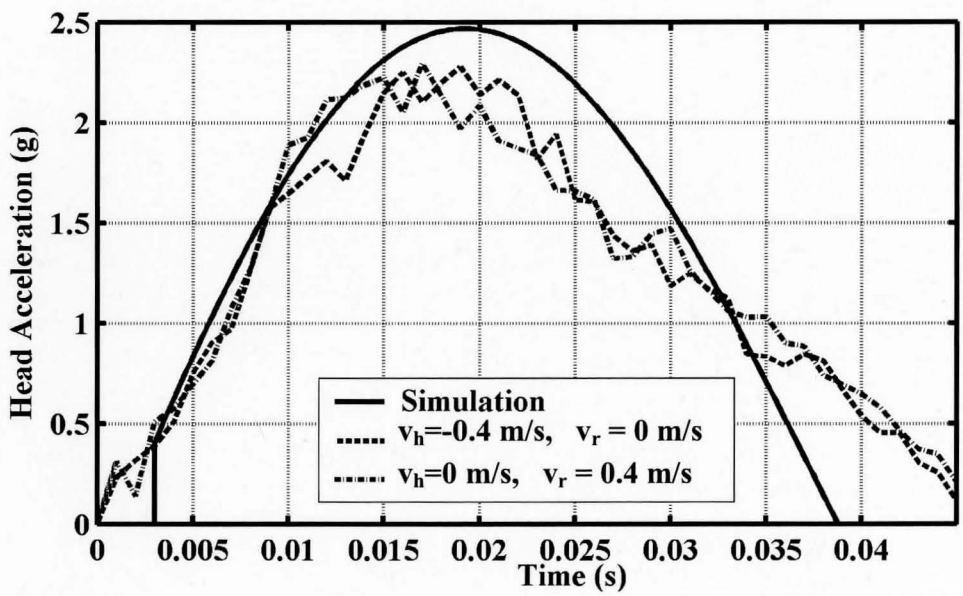


Fig. 4.8a. The head accelerations for the simulation and impact experiments with Foam 1, Puma 560 manipulator and $v_c = 0.4$ m/s

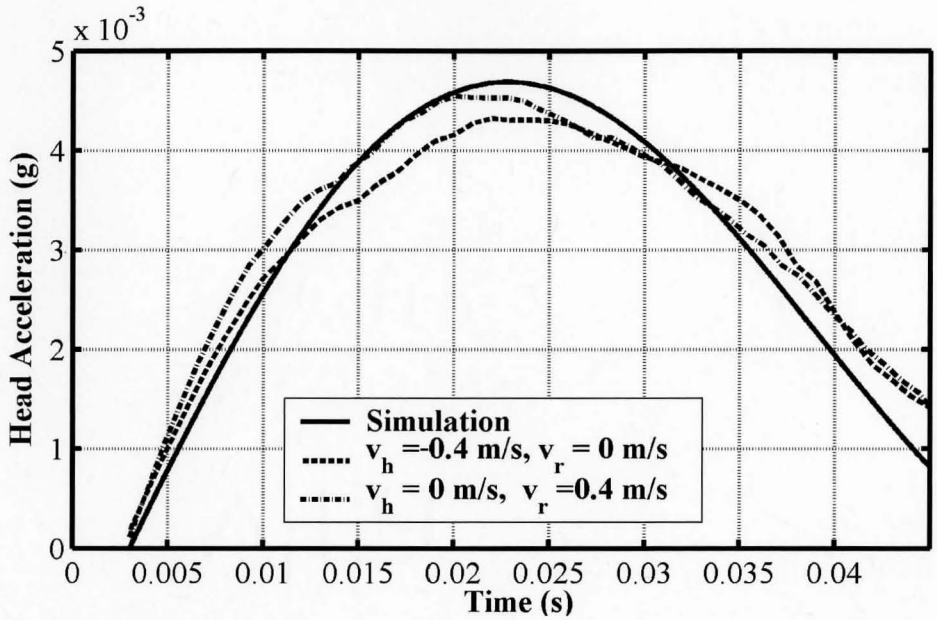


Fig. 4.8b. The compressed depth for the simulation and impact experiments with Foam 1, Puma 560 manipulator and $v_c = 0.4$ m/s

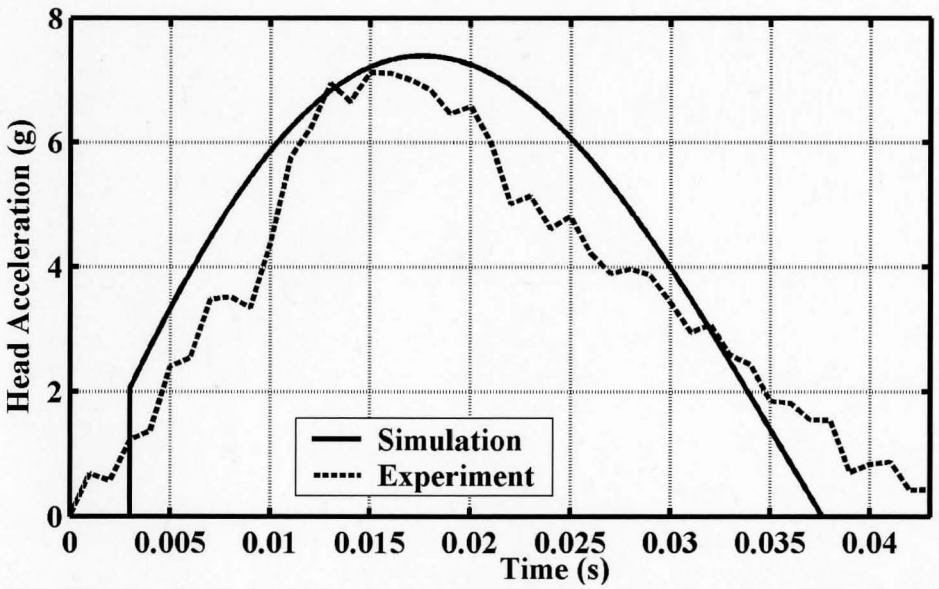


Fig. 4.9a. The head accelerations for the simulation and impact experiments with Foam 1, Puma 560 manipulator and $v_c = 1.25$ m/s

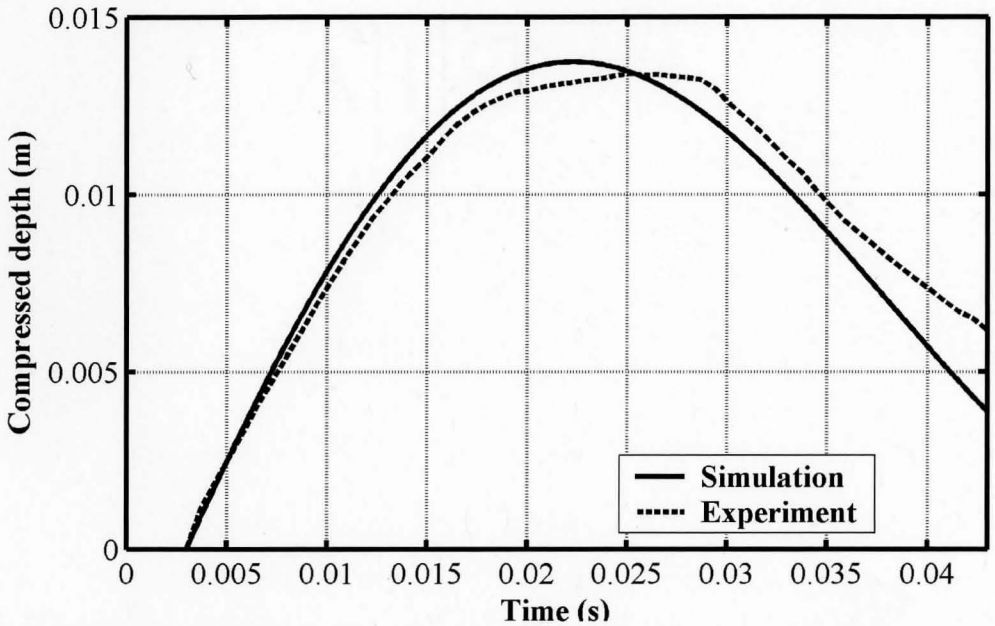


Fig. 4.9b. The compressed depth for the simulation and impact experiments with Foam 1, Puma 560 manipulator and $v_c = 1.25$ m/s

4.5 Conclusions

The impact model and model-based foam covering design procedure have been experimentally verified in this chapter. Impact experiments were performed with an apparatus simulating the human head, and two different manipulators and two foam coverings. Different impact velocities including the maximum impact velocity (v_c = 1.25 m/s) were also used in the experiments. The average error between the predicted and experimental $a_{h,\max}$ results is 7% and maximum error is less than 9%. The average error between the predicted and experimental $\varepsilon_{\max}$ results is 7% and maximum error is less than 12%.

The values of $a_{h,\max}$ in our experiments are always smaller than the predicted values from (3.5.14) and $\varepsilon_{\max}$ are always smaller than the values obtained from (3.9.2). In other words, our model gives a conservative result that is beneficial from the safety perspective.

Chapter 5

Design of a VFF-based mobile robot navigation algorithm for environments with humans

5.1 Introduction

As discussed in Chapter 2, VFF-based navigation algorithms are suitable for mobile robot navigation for environments with motion-unpredictable obstacles, such as humans. The following problems remain unsolved to date: path oscillation; stability; sizes of the active and critical regions. In this chapter we will present a novel VFF-based algorithm for holonomic mobile robot navigation for environments with humans. This will include a method to calculate the sizes of the active and critical regions from the humans and robot parameters, and the design of the improved virtual force functions for avoiding unpredictable obstacles and diminishing the path oscillation. The stability of the new VFF will be analyzed using Lyapunov's second method. Three new criteria will also be introduced to evaluate the navigation performance. Five well-known navigation algorithms will be compared with the new VFF-based algorithm by simulations with an unpredictable human path. In this chapter, we will design the new VFF-based algorithm for holonomic robots. The extensions required for the navigation of nonholonomic robots will be addressed in Chapter 7.

5.2 Assumptions and limitations

In this thesis, the following assumptions for mobile robot navigation must hold:

1) The human(s) future motion is unpredictable since it is both unknown and stochastic.
2) The human(s) motion is restricted to common walking with a velocity magnitude less than 1 m/s (Van Emmerik and Wagenaar 1996)
3) The robot operates on a horizontal plane so gravity does not influence the dynamics of the robot.
4) The robot dynamics are defined by:

$$\mathbf{F}_v = m_{mr}\mathbf{a}_{mr} \tag{5.2.1}$$

where $\mathbf{F}_v$ is the virtual force; m_{mr} is the mass of the holonomic robot and $\mathbf{a}_{mr}$ is the vector of acceleration of the robot.

5) The robot goal is stationary and outside of the active region(s) of the obstacle(s) for safety.
6) The region containing the goal is contiguous.

5.3 Sizes of active and critical regions

5.3.1 Active and critical regions for a human

Our approach for determining the sizes of the active region and the critical region for the case when the obstacle is a human will be explained first. The sizes of the two regions will be determined from the robot and human shapes; velocity limits and acceleration limits. The active region $\mathbb{C}_2$ is defined as the region near the human. If any mobile robot is in $\mathbb{C}_2$, the repulsive and detour virtual forces should be activated to make the robot avoid and detour around the human. Therefore, the size of $\mathbb{C}_2$ is a key issue in

the design of a navigation system. It determines how much space is available for the robot to avoid the human. If $\mathbb{C}_2$ is large, the space available for navigation is wide and the distance between the human and robot will increase, but the time to reach the goal will also increase. The critical region $\mathbb{C}_3$ is the region in close proximity to the human, and it is very dangerous if a robot intrudes into this region. Within this region, the robot cannot complete the avoidance and should stop to prevent or at least mitigate the collision. Any other action made by the robot could worsen the situation. Therefore, when the robot just enters $\mathbb{C}_3$, it should be fully decelerated until it is stopped. We define $\mathbb{C}_1$ as the region which excludes $\mathbb{C}_2$ and $\mathbb{C}_3$. $\mathbb{C}_1$ contains the goal. If the robot is in this region collision avoidance is unnecessary and the robot just moves towards its goal. Since the human's motion is arbitrary, it can change its moving velocity and direction any time subject to a limit in linear velocity magnitude. For safety, we need to consider the arbitrary motion made by the human, so the active and critical regions for humans are represented by circular regions to consider all directions (*i.e.* assuming the angular velocity of the human is infinite). In the following analysis, we will focus on the design of $\mathbb{C}_2$ and $\mathbb{C}_3$ for the human with consideration of the human's linear velocity limit.

5.3.2 Size of the critical region for a human

In this subsection, we will calculate the size of the critical region for a human. Since a mobile robot moves on the floor, when it avoids a human, the human will be modeled as the projected shape of its body on the floor. The projected shape is highly related the pose of the human. By modeling a human body as an enclosing cylinder, we

can neglect the difference poses and simplify the navigation system while still maintaining safety. The human cylinder shape projected onto the horizontal motion plane is a circle, as shown in Fig. 5.1. The average step length of a human is around 0.8 m (Martin and Marshi 1992). One half of this value will be used as the radius of the human model, *i.e.* $\rho_h = 0.4$ m.

To simplify the design of the critical region, the shape of the robot will be reduced to a point shape and the region will be appropriately enlarged. We will use the 2D Minkowski sum (Chapter 17 of Skiena 2008) to sum the circle of the human and the shape of the robot. In this thesis, the robot is modelled as a circle with a radius of ρ_r. The resultant Minkowski sum area (shown as the circular area enclosed by the dashed-line in Fig. 5.1) is a disk with a radius of $\rho_r + \rho_h$. Therefore when the centre position of the robot lies on the dashed-circle, the robot will just contact the human model. The shape of the robot then can be regarded as point shape in the following analysis.

The size of the critical region is related to the stopping distance, $\hat{r}_3$, that ensures the robot-human impact will not occur before the robot is stopped for the worst case; *i.e.* the human and robot approach in the collinear condition (defined in Chapter 2) with their maximum velocities $\bar{V}_h$ and $\bar{V}_r$, as shown in Fig. 5.1. In equation form:

$$\hat{r}_3 > \bar{\delta}_{h1} + \bar{\delta}_{r1} , \tag{5.3.1}$$

where $\bar{\delta}_{r1} = \bar{V}_r T_s - \iint_{\Delta t_1} a_{mrd} dt$ is the distance required to stop the robot; a_{mrd} is the magnitude of the robot's deceleration; Δt_1 is the time required for stopping the robot and

T_s is the sampling time of the navigation system. T_s must be considered due to the lack of position information of the human and robot between sampling instants. $\overline{\delta}_{h1} = \overline{V}_h \times (\Delta t_1 + T_s)$ is the maximum distance that the human could travel towards the robot during $\Delta t_1 + T_s$. When the inequality (5.3.1) is satisfied, the robot will have enough distance to stop and prior to contact with the human. With the kinematic relationship between the velocity and deceleration, the robot stopping time, Δt_1 must satisfy the following relationship:

$$\overline{V}_r = \int_{\Delta t_1} a_{mrd} dt \tag{5.3.2}$$

When the robot enters $\mathbb{C}_3$ of the human, it should decelerate with its maximum deceleration magnitude, $a_{mrd} = \overline{a}_{mr}$, until it stops. Substituting this deceleration into (5.3.2), $\overline{V}_r = \overline{a}_{mr} \Delta t_1$. We then obtain $\Delta t_1 = \overline{V}_r / \overline{a}_{mr}$, and $\overline{\delta}_{r1} = \overline{V}_r T_s + \frac{1}{2} \overline{a}_{mr} \Delta t_1^2$ for use in (5.3.1). $\hat{r}_3$ also defines a circular area, as shown in Fig. 5.1. The Minkowski sum is used

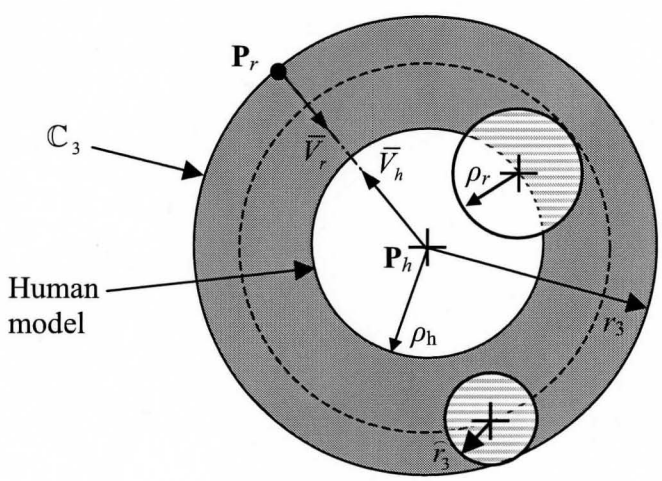


Fig. 5.1. The critical region of a human

to sum this circular area with the area of the sum of the human and robot shapes. From Fig. 5.1, a larger circular area considered the stopping distance and the shapes of the human and robot is then obtained, and its size is defined with a radius of r_3. We will use this area as the critical region of the human. Accordingly, the size of $\mathbb{C}_3$ is defined as:

$$r_3 = \hat{r}_3 + \rho_h + \rho_r \tag{5.3.3}$$

5.3.3 Size of the active region for a human

The active region of a human should provide the robot(s) enough room to avoid the human. As previously mentioned, the worst case is the collinear condition with the human and robot approaching at their maximum speeds, as shown in greater detail from Fig. 5.2. The initial conditions are as follows: the position and velocity of the robot centre are $\mathbf{P}_{r0} = \begin{bmatrix} P_{rx0} & P_{ry0} \end{bmatrix}^{\mathrm{T}}$ and $\mathbf{V}_{mr} = \begin{bmatrix} \overline{V}_{mr} & 0 \end{bmatrix}^{\mathrm{T}}$, respectively; and the position and velocity of the human centre are $\mathbf{P}_{h0} = \begin{bmatrix} P_{hx0} & P_{hy0} \end{bmatrix}^{\mathrm{T}}$, and $\mathbf{V}_h = \begin{bmatrix} -\overline{V}_h & 0 \end{bmatrix}^{\mathrm{T}}$, respectively. The human's velocity here is assumed constant. Due to the detour force the robot's velocity will change direction, and the robot will move to the point $\mathbf{P}_r$. The displacement of the robot in the Y-direction must be larger than r_3, the radius of $\mathbb{C}_3$. Otherwise, the robot will enter $\mathbb{C}_3$. Therefore, from Fig. 5.2, the radius of $\mathbb{C}_2$ must be larger than the moving distance of the human and the robot in the X-direction before the robot moves to $\mathbf{P}_r$ in this worst case. The radius of $\mathbb{C}_2$ is then constrained to be

$$r_2 > \overline{\delta}_{hx2} + \delta_{rx2} \tag{5.3.4}$$

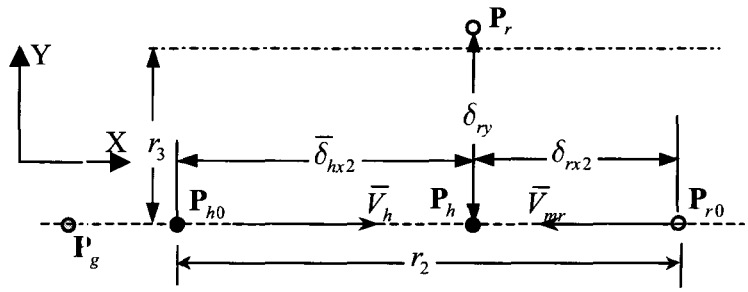


Fig. 5.2. The collinear condition

where $\delta_{rx2} = P_{rx} - P_{r0} = \bar{V}_{mr} \times (\Delta t_2 + T_s) + \iint_{\Delta t_2 + T_s} a_{rx} d^2t$; a_{mrx} is the acceleration component of the robot in the X-direction; $\bar{\delta}_{hx2} = P_{hx} - P_{hx0} = \bar{V}_h \times (\Delta t_2 + T_s)$; and Δt_2 is the time required for the robot to move by r_3 in the Y direction. Since in this case the robot needs to move in both X and Y directions, the robot is required to decelerate in the X-direction (*i.e.* $a_{rx} < 0$) and accelerate to reach the dashed-dot line in the Y-direction. Noting that the velocity of the robot is also limited, we have:

$$\delta_{ry2} = \bar{V}_{mr} \times (\Delta t_2 - t_a) + \iint_{t_a} a_{mry} dt > r_3 \tag{5.3.5}$$

where a_{mry} is the acceleration of the robot in the Y-direction; t_a is the time spent by the robot accelerating from zero velocity to $\bar{V}_{mr}$ in the Y-direction. It is impossible to obtain the acceleration in the Y-direction analytically since it will be different for different algorithms, so Δt_2 cannot be obtained directly. We will estimate Δt_2 by first assuming the robot is driven with maximum acceleration magnitude[1], $a_{mry} = \bar{a}_{mr}$ in the Y-direction.

[1] It is assumed that the maximum acceleration magnitude of the mobile robot equals the maximum deceleration magnitude.

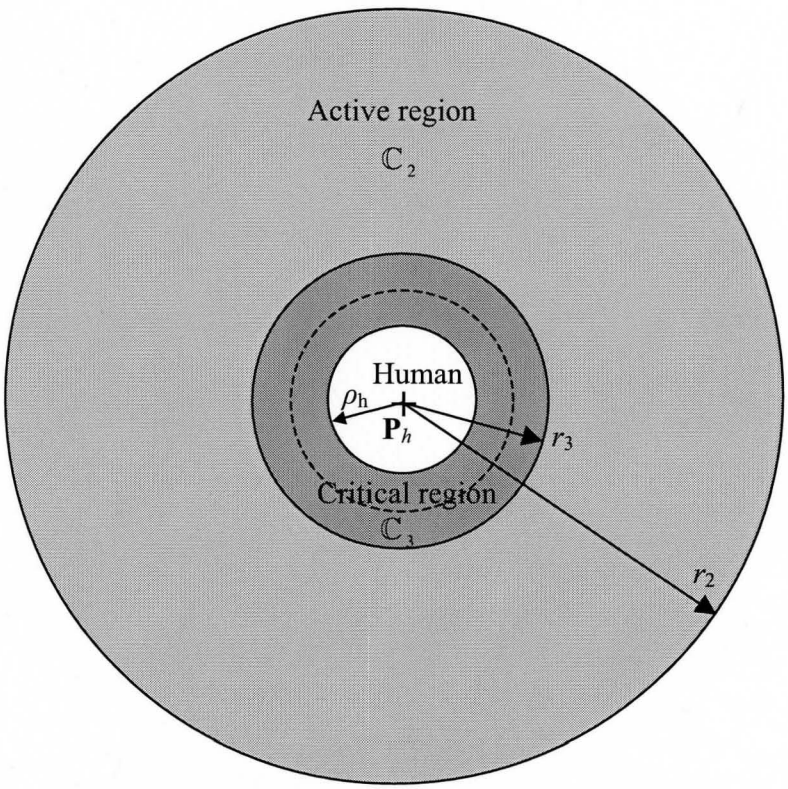


Fig. 5.3. The active and critical regions for a human.

Therefore, (5.3.5) is rewritten as:

$$\overline{V}_{mr} \times (\Delta t_2 - t_a) + \tfrac{1}{2}\overline{a}_{mr} {t_a}^2 = \delta_{ry2} > r_3 \tag{5.3.6a}$$

So Δt_2 can be estimated with:

$$\Delta t_2 > \frac{r_3 - \frac{1}{2}\overline{a}_{mr} t_a^2}{\overline{V}_{mr}} + t_a \tag{5.3.6b}$$

where $t_a = \overline{V}_{mr} / \overline{a}_{mr}$. The robot acceleration magnitude in the Y-direction will actually be less than $\overline{a}_{mr}$ since the robot must also decelerate in the X-direction, so the actual Δt_2 will be longer than the computed values, and (5.3.4) will underestimate the value of r_2. To compensate for this error, we can assume the worst case of $a_{mrx} \approx 0$, and then with

$\overline{\delta}_{rx2} = \overline{V}_{mr} \times (\Delta t_2 + T_s)$, the conservative estimate for r_2 is given by:

$$r_2 > \overline{\delta}_{hx2} + \overline{\delta}_{rx2} \tag{5.3.7}$$

The active and critical regions for the human are in Fig. 5.3.

5.3.4 Sizes of active and critical regions for stationary circular obstacles

The procedure for determining r_2 and r_3 for humans can be used for other circular obstacles, such as stationary circular obstacles. Since the velocity of a stationary obstacle is zero, according to (5.3.3) and (5.3.7), avoiding the obstacle requires

$$r_{3,cir} > \overline{\delta}_{r1} + \rho_{circular} + \rho_r \text{ and} \tag{5.3.8a}$$

$$r_{2,cir} > \overline{\delta}_{rx2,circular} \tag{5.3.8b}$$

where $r_{2,cir}$ and $r_{3,cir}$ are the radii of the active and critical regions for the stationary circular obstacle, respectively; $\rho_{circular}$ is the radius of the circular obstacle model; and $\overline{\delta}_{rx2,circular} = \overline{V}_{mr} \times (\Delta t_{2\,circular} + T_s)$. The value of $\Delta t_{2,circular}$ is related to $r_{3,cir}$. Similar to (5.3.6a) and (5.3.6b), $\Delta t_{2,circular}$ can be obtained by solving the following relationship:

$$\overline{V}_{mr} \times (\Delta t_{2,circular} - t_a) + \tfrac{1}{2}\overline{a}_{mr} t_a^{\,2} > r_{3,cir} \tag{5.3.9a}$$

So

$$\Delta t_{2,circular} > \frac{r_{3,cir} - \frac{1}{2}\overline{a}_{mr} t_a^{\,2}}{\overline{V}_{mr}} + t_a \tag{5.3.9b}$$

5.3.5 Sizes of the active and critical regions for a stationary rectangular obstacle

For a stationary rectangular obstacle, as shown in Fig. 5.4, too much floor area

will be wasted by modeling it as a circle. The sizes of its active and critical regions should be determined by modeling it as a rectangle. As we did in subsection 5.3.2, we will use the Minkowski sum to sum the rectangle and the shape of the robot to allow us to reduce the shape of the robot to a point shape. The resultant area after the Minkowski sum expands the rectangle by ρ_r, and is enclosed by the dashed line in Fig. 5.5.

To obtain the region sizes, we need to know the worst case for a robot avoiding this obstacle. As shown in Fig. 5.4, the robot approaches the obstacle with a velocity, $\mathbf{V}_{mr}$. It is clear that its velocity component $V_{mr,t}$ (*i.e.* in tangential direction along the boundary of the Minkowski sum area) will not cause the collision, and it will help the robot to detour around the obstacle. The other velocity component $V_{mr,n}$ will cause a collision, and:

$$V_{mr,n} = \|\mathbf{V}_{mr}\|_2 \cos\gamma_{mr} \tag{5.3.10}$$

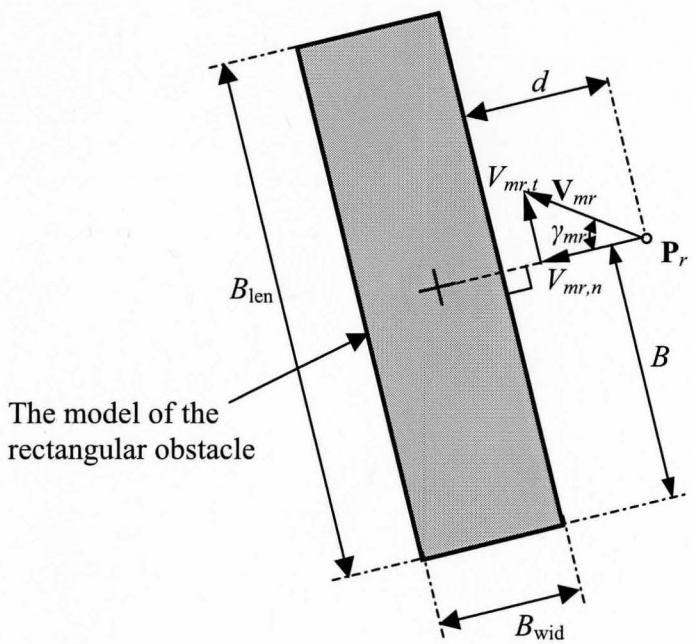


Fig. 5.4 The geometry of the robot avoiding the rectangular obstacle.

where $\|\bullet\|_2$ is the Euclidean norm. The worst case occurs when $\gamma_{mr} = 0$ and $V_{mr,n} = \overline{V}_{mr}$. For this case, to stop the robot before contact occurs requires:

$$\hat{r}_{3,rect} > \overline{\delta}_{r1} \tag{5.3.11a}$$

where $\overline{\delta}_{r1} = \overline{V}_{mr} T_s + \frac{1}{2}\overline{a}_{mr}\Delta t_1^2$ and $\hat{r}_{3,rect}$ is the stopping distance. This distance also defines a circular area, as shown in Fig. 5.5. The Minkowski sum may be used to sum this area with the shapes of the obstacle and robot. This summed area is defined as the critical region of the rectangular obstacle and can be regarded simply as the Minkowski sum of the shape of the obstacle and a circular area with a radius of $\hat{r}_{3,rect} + \rho_r$, as shown in Fig. 5.5. So the size of $\mathbb{C}_{3,rect}$ is determined by:

$$r_{3,rect} = \hat{r}_{3,rect} + \rho_r . \tag{5.3.11b}$$

For the active region, similar to (5.3.8b), we require:

$$\hat{r}_{2,rect} > \overline{\delta}_{rx2,rect} \tag{5.3.12}$$

where $\overline{\delta}_{rx2,rect} = \overline{V}_{mr} \times \left(\Delta t_{2,rect} + T_s\right)$, and $\Delta t_{2,rect}$ must satisfy:

$$\overline{V}_{mr} \times \left(\Delta t_{2,rect} - t_a\right) + \frac{1}{2}\overline{a}_{mr} t_a^2 > r_{3,rect} + B \tag{5.3.13}$$

where B is the distance between the robot's position and the boundary of the obstacle perpendicular to $V_{mr,n}$, as shown in Fig. 5.4. $\hat{r}_{2,rect}$ defines a circular area too. So, $\Delta t_{2,rect}$ in (5.3.12) and (5.3.13) will be obtained by solving the following inequalities:

$$\Delta t_{2,rect} > \frac{r_{3,rect} + B - \frac{1}{2}\overline{a}_{mr} t_a^2}{\overline{V}_{mr}} + t_a \tag{5.3.14}$$

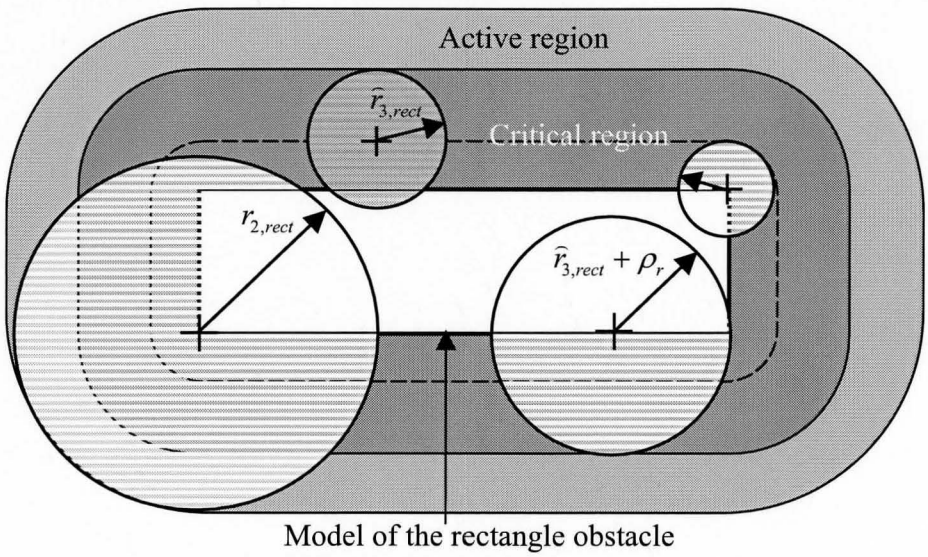


Fig. 5.5 The active and critical regions of the rectangular obstacle.

Since different B values will be obtained for different robot positions, $\Delta t_{2,rect}$ will be different with different robot positions according to (5.3.14). A larger B causes a larger $\Delta t_{2,rect}$, and a larger $\Delta t_{2,rect}$ makes a larger $\widehat{r}_{2,rect}$. Therefore, we should use the largest B to obtain the largest $\widehat{r}_{2,rect}$ value, termed $r_{2,rect}$. We will use $r_{2,rect}$ to define the active region of the rectangle obstacle rather than the position dependent $\widehat{r}_{2,rect}$. For the worst case of $\gamma_{mr} = 0$, the largest B equals $\frac{1}{2}B_{len}$ (half the length of the longer side of the rectangle). This occurs when the direction of $\mathbf{V}_{mr}$ is through the centroid of the rectangle (*i.e.* the + symbol in Fig. 5.4). Note that if the goal is behind the obstacle and $\mathbf{V}_{mr}$ also passes through the goal, this is the collinear condition for the rectangle obstacle. $r_{2,rect}$ also defines a circular area, as shown in Fig. 5.5. The outer boundary of $\mathbb{C}_{2,rect}$ can be

obtained by using Minkowski sum to sum this area and the shape of the obstacle. The resultant active region is illustrated in Fig. 5.5.

5.4 Design of the virtual force functions

This section begins by discussing the case of a robot avoiding a human as an example of an unpredictable obstacle. The geometric configuration of the goal, the robot and the human for a navigation system is presented in Fig. 5.6. As in Section 5.3, $\mathbf{P}_g = \begin{bmatrix} P_{gx} & P_{gy} \end{bmatrix}^{\mathrm{T}}$, $\mathbf{P}_r = \begin{bmatrix} P_{rx} & P_{ry} \end{bmatrix}^{\mathrm{T}}$ and $\mathbf{P}_h = \begin{bmatrix} P_{hx} & P_{hy} \end{bmatrix}^{\mathrm{T}}$ are the current position vectors of the goal, the robot centre and the human centre, respectively. $\mathbf{V}_{mr}$ and $\mathbf{V}_h$ are the velocity vectors of the robot and the human. We have:

$$\mathbf{E} = \mathbf{P}_g - \mathbf{P}_r \tag{5.4.1a}$$

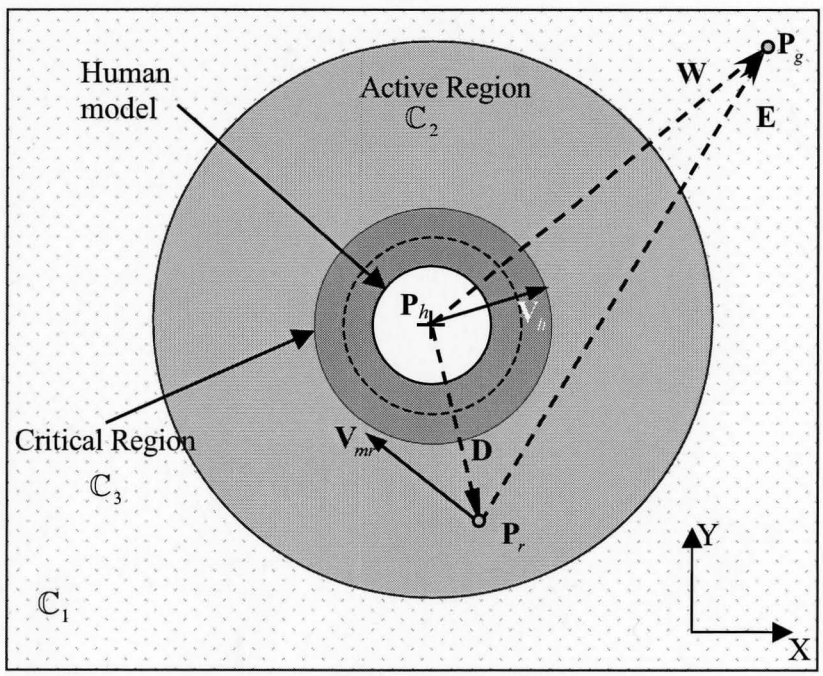


Fig. 5.6. Configuration of collision avoidance for a mobile robot and a human obstacle.

$$\mathbf{D} = \mathbf{P}_r - \mathbf{P}_h \tag{5.4.1b}$$

$$\mathbf{W} = \mathbf{P}_g - \mathbf{P}_h \tag{5.4.1c}$$

where $\mathbf{E}$ is the vector from the robot to its goal, $\mathbf{D}$ is the vector from the obstacle to the robot, and $\mathbf{W}$ is the vector from the obstacle to the goal.

An attractive virtual force is used to guide the robot to the goal, as shown in Fig. 5.7. It is activated when the robot is within $\mathbb{C}_1$ or $\mathbb{C}_2$. As in Khatib (1986); Ge and Cui (2002), its force function is defined as:

$$\mathbf{F}_a = \begin{cases} K_1\mathbf{E} + K_2\dot{\mathbf{E}} & \text{if } \mathbf{P}_r \in \mathbb{C}_2 \cup \mathbb{C}_1 \\ \text{undefined} & \text{if } \mathbf{P}_r \in \mathbb{C}_3 \end{cases} \tag{5.4.2}$$

where K_1 and K_2 are positive attractive virtual force gains. $\mathbf{E} = \begin{bmatrix} P_{gx} - P_{rx} & P_{gy} - P_{ry} \end{bmatrix}^T$ and $\dot{\mathbf{E}} = \begin{bmatrix} -\dot{P}_{rx} & -\dot{P}_{ry} \end{bmatrix}$ since the goal is stationary.

A repulsive virtual force is used to keep the robot away from the human and is activated when the robot is within $\mathbb{C}_2$, as shown in Fig. 5.7. In the existing literature, the repulsive force is normally a function of the distance between the human and robot, $d = \|\mathbf{D}\|_2$, and its rate of change, $\dot{d}$. Its direction is along $\mathbf{D}$. We propose the improved repulsive force function as follow:

$$\mathbf{F}_\Lambda = \begin{cases} \left(K_3\Lambda + K_4\dot{\Lambda}^*\right)\mathbf{u}_\Lambda & \text{if } \mathbf{P}_r \in \mathbb{C}_2 \\ \text{undefined} & \text{if } \mathbf{P}_r \in \mathbb{C}_1 \cup \mathbb{C}_3 \end{cases} \tag{5.4.3}$$

where K_3 and K_4 are positive repulsive force gains; $\mathbf{u}_\Lambda$ is a unit vector along $\mathbf{D}$, pointing

away from the human; $\Lambda = \dfrac{(r_2 - d)^2}{d - r_3}$; and $\dot{\Lambda}^* = -\dot{d}\dfrac{(r_2 - d)^2}{(d - r_3)^2} = -K_\Lambda \dot{d}$. Note that $\mathbf{P}_r \in \mathbb{C}_2 \Leftrightarrow r_3 < d < r_2$. Unlike the existing functions, the numerator term of Λ, $(r_2 - d)^2$, causes a gentle increase of the repulsive force when the robot enters the active region. The denominator term, $1/(d - r_3)$, causes the force to increase greatly when the robot is near the boundary of $\mathbb{C}_3$. This tends to push the robot away from the critical region. $\mathbf{F}_\Lambda$ is designed to make the VFF continuous as will be proven below. Note that velocities of the human and robot are included in $\dot{\Lambda}^*$ of (5.4.3). When the human and robot approach each other, $\dot{d} < 0$, so $\dot{\Lambda}^* > 0$ and the repulsive force will increase to push the robot away.

The detour force, $\mathbf{F}_\psi$, is a virtual force perpendicular to $\mathbf{u}_\Lambda$, as shown in Fig. 5.7.

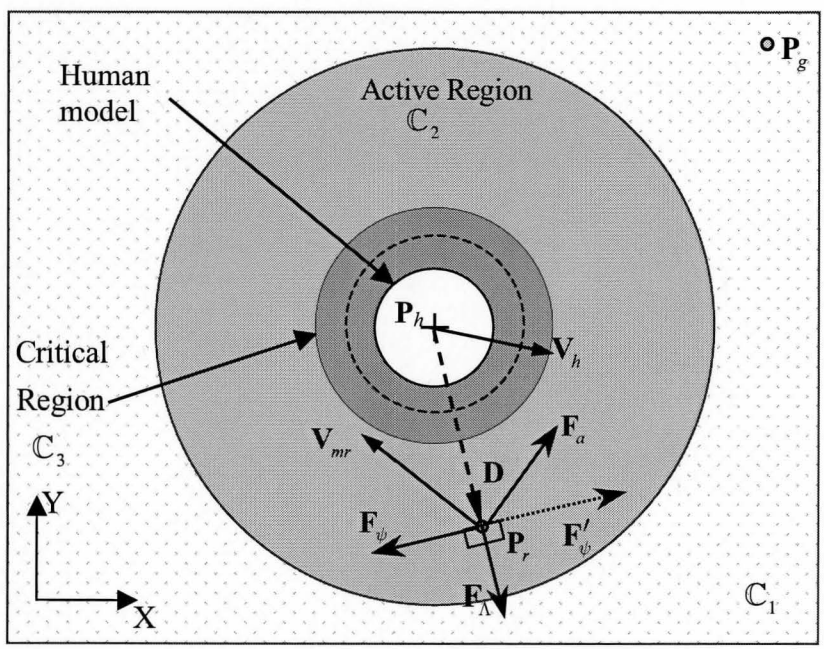


Fig. 5.7. The attractive, repulsive and detour force when a robot avoiding a human.

Its main purpose is to push the robot to detour around the human while still moving to the goal. Its other purpose is dealing with the collinear condition. The first issue when designing this force is its direction. This issue has not been adequately addressed in the existing literature. If the force direction were $\mathbf{F}'_\psi$ (see Fig. 5.7), the robot would have to pass in front of the obstacle. This is acceptable for static obstacles, but is not desirable with humans for physical and emotional reasons. Physically, although the current human velocity may be slow, the human could accelerate at any instant. Therefore, there may not be enough time for the robot to pass in front. In this case, the robot could block the motion of the human possibly resulting in tripping and/or a severe impact. Emotionally, it may be considered rude. The choices for the force direction are illustrated in Fig. 5.8. In Fig. 5.8a, the goal and robot are on different sides of the human's velocity line. In this case, the detour force should cause the robot to detour around behind the human to prevent blocking the human. In Fig 5.8b, since the goal and robot are on the same side of the human's velocity line, the robot will not block the human. Then, the direction of $\mathbf{F}_\psi$, $\mathbf{u}_\psi$, should be towards the goal to help the robot reach its goal faster. For the collinear condition, since the robot, its goal and the obstacle are collinear and their velocity lines are also along this line, the condition for avoidance is symmetrical to this line. When there are other obstacles, we should choose the force direction to satisfy

$$\mathbf{u}_\psi \cdot \sum \mathbf{F}_\psi > 0 \tag{5.4.4}$$

where $\sum \mathbf{F}_\psi$ is the sum of the detour forces from the other obstacles. There is no velocity line for stationary obstacles. Therefore, to reach the goal early, the force direction should

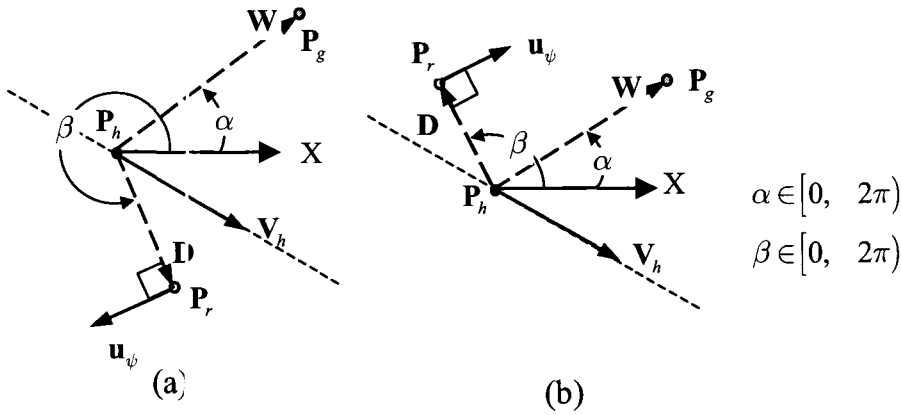


Fig.5.8. The direction of the detour force

be chosen towards the goal with stationary obstacles (Fig. 5.8b). This force should be zero when the robot is between the goal and the human (on the same line), or $\alpha = \beta$ (α is the angle of the vector **W** to the positive X-axis, and β is the angle of **D** relative to the positive X-axis). Note that $\alpha = \pi$ and $\beta = 0$ in the collinear condition of Fig. 5.2. So, the proposed new detour force function is:

$$\mathbf{F}_\psi = \begin{cases} \left(K_5\psi + K_6\dot{\psi}^*\right)\mathbf{u}_\psi & \text{if } \mathbf{P}_r \in \mathbb{C}_2 \\ \text{undefined} & \text{if } \mathbf{P}_r \in \mathbb{C}_1 \cup \mathbb{C}_3 \end{cases} \tag{5.4.5}$$

where K_5 and K_6 are the detour force gains; $\psi = (r_2 - d)^2 \Phi$; $\Phi = \alpha - \beta$; $\dot{\psi}^* = (r_2 - d)^2 \dot{\Phi}$; and $\mathbf{u}_\psi$ is the unit vector in the direction of the detour force.

For circular obstacles, (5.4.3) and (5.4.5) are also applicable by changing r_2 and r_3 to the corresponding values for those obstacles. For a rectangular obstacle, as shown in Fig. 5.9, the repulsive force direction is also along **D**, and $\mathbf{D} = \mathbf{P}_r - \mathbf{P}_{rect}$ where $\mathbf{P}_{rect}$ is the point on the external contour of the rectangle with the shortest distance to $\mathbf{P}_r$.

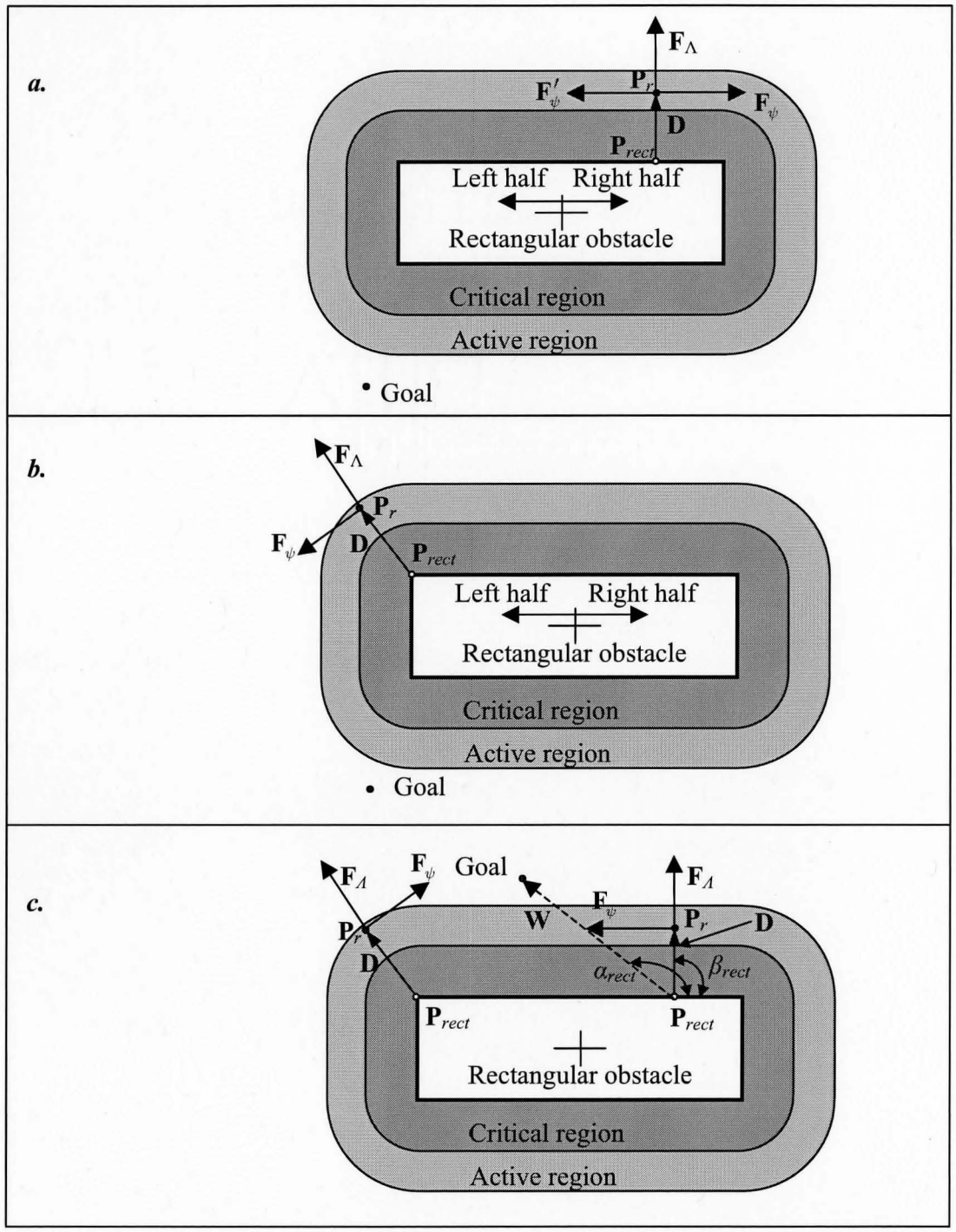


Fig. 5.9. The directions of the repulsive and detour forces when avoiding a rectangular obstacle. '+' is used to denote the centroid of the rectangle.

$\Lambda_{rect} = \frac{(r_{2,rect} - d_{rect})}{d_{rect} - r_{3,rect}}$; and $\dot{\Lambda}^*_{rect} = -\dot{d}_{rect} \frac{(r_{2,rect} - d_{rect})^2}{(d_{rect} - r_{3,rect})^2}$ for the repulsive forces. When the robot and its goal are separated by the rectangle, if $\mathbf{P}_r$ is in the left half of $\mathbb{C}_{2,rect}$, the detour force direction should point to the left, as shown in Fig. 5.9a. If $\mathbf{P}_r$ is in the right half of $\mathbb{C}_{2,rect}$, the force should point to the right, as shown in Fig. 5.9b. The reason will now be explained. In Fig. 5.9a, if the direction of the detour force is along $\mathbf{F}'_\psi$, the robot will travel a shorter path than when it is along $\mathbf{F}_\psi$. However, the value of B will be larger than $\frac{1}{2}B_{len}$. According to the analysis in subsection 5.3.5, this will cause a larger $r^*_{2,rect}$, and then (5.14c) will not provide an enough active region for this obstacle. When the robot and its goal is in the same side of the rectangle, as shown in Fig. 5.9c, the direction of the detour force should be chosen towards the goal to help the robot reach its goal faster. In the detour force function, $\psi_{rect} = (r^*_{2,rect} - d_{rect})^2 (\alpha_{rect} - \beta_{rect})$ where α_{rect} is the angle of the vector $\mathbf{W}$ and β_{rect} is the angle of the vector $\mathbf{D}$, as shown in Fig. 5.9.

To ensure stability, we add the following *stabilizing virtual force* into the VFF:

$$\mathbf{F}_L = \begin{cases} \frac{K_L \mathbf{u}_L}{\|\mathbf{v}_L\|_2} & \text{if } \mathbf{P}_r \in \mathbb{C}_2 \\ \text{undefined} & \text{if } \mathbf{P}_r \in \mathbb{C}_1 \cup \mathbb{C}_3 \end{cases} \tag{5.4.6}$$

where $\mathbf{v}_L = K_1 \dot{\mathbf{E}} + K_3 \dot{\Lambda}^* \mathbf{u}_\Lambda + K_5 \dot{\psi}^* \mathbf{u}_\psi$ and $K_L \geq 0$ is a time-varying parameter; and $\mathbf{u}_L$ is the unit vector along the direction of $\mathbf{v}_L$. The details of this force will be given in the stability analysis presented in the next section.

The VFF is the combination of the four forces. Thus, the VFF for the robot with a single human/obstacle is:

$$\mathbf{F}_V := \begin{cases} \mathbf{F}_a & \text{if } \mathbf{P}_r \in \mathbb{C}_1 \\ \mathbf{F}_a + \mathbf{F}_\Lambda + \mathbf{F}_\psi + \mathbf{F}_L & \text{if } \mathbf{P}_r \in \mathbb{C}_2 \\ \text{undefined} & \text{if } \mathbf{P}_r \in \mathbb{C}_3 \end{cases} \tag{5.4.7}$$

If the robot intrudes into $\mathbb{C}_3$, since it may not have enough space for collision avoidance the VFF is not used, and the robot is decelerated to a stop. This stopping action should prevent or at least mitigate the collision. Any other action made by the robot could worsen the situation.

A VFF should be continuous to reduce path oscillations. For our VFF to be continuous the virtual forces at the boundary between $\mathbb{C}_1$ and $\mathbb{C}_2$ must be equal. From (5.4.7) at the boundary of $\mathbb{C}_1$:

$$\mathbf{F}_{V,\mathbf{P}_r \in \mathbb{C}_1} = \mathbf{F}_a \tag{5.4.9}$$

At the boundary of $\mathbb{C}_2$, we have

$$\lim_{d \to r_2} \Lambda = \lim_{d \to r_2} (r_2 - d)^2 / (d - r_3) = 0, \tag{5.4.10a}$$

$$\lim_{d \to r_2} \dot{\Lambda}^* = \lim_{d \to r_2} -\dot{d}(r_2 - d)^2 / (d - r_3)^2 = 0, \tag{5.4.10b}$$

$$\lim_{d \to r_2} \mathbf{F}_\Lambda = \lim_{d \to r_2} K_3 \Lambda \mathbf{u}_\Lambda + K_4 \dot{\Lambda}^* \mathbf{u}_\Lambda = \mathbf{0}, \tag{5.4.11}$$

$$\lim_{d \to r_2} \psi = \lim_{d \to r_2} (r_2 - d)^2 \Phi = 0, \tag{5.4.12a}$$

$$\lim_{d \to r_2} \dot{\psi}^* = \lim_{d \to r_2} (r_2 - d)^2 \dot{\Phi} = 0 \text{ and} \tag{5.4.12b}$$

$$\lim_{d\to r_2}\mathbf{F}_{\psi}=\lim_{d\to r_2}K_5\psi\mathbf{u}_{\psi}+K_6\dot{\psi}^{*}\mathbf{u}_{\psi}=\mathbf{0}, \tag{5.4.13}$$

$$\lim_{d\to r_2}\mathbf{F}_{L}=0 \tag{5.4.14}$$

The proof of (5.4.14) is given in the next section. Therefore:

$$\lim_{d\to r_2}\mathbf{F}_{V,P_r\in\mathbb{C}_2}=\lim_{d\to r_2}\mathbf{F}_a+\mathbf{F}_{\Lambda}+\mathbf{F}_{\psi}+\mathbf{F}_{L}=\mathbf{F}_a=\mathbf{F}_{V,P_r\in\mathbb{C}_1} \tag{5.4.15}$$

From (5.4.15) the piecewise VFF is continuous.

According to the above analysis, if the robot is sharing its work area with N obstacles (which could include other robots), the force field will be:

$$\mathbf{F}_V=\begin{cases}\mathbf{F}_a & \text{if } \mathbf{P}_r\in\mathbb{C}_1\\ \text{undefined} & \text{if } \mathbf{P}_r\in\mathbb{C}_{3,n}\ i\in[1,...,N]\\ \mathbf{F}_a+\sum_{n=1}^{N}c_n\left(\mathbf{F}_{\Lambda,n}+\mathbf{F}_{\psi,n}\right)+\mathbf{F}_{L,N} & \text{otherwise}\end{cases} \tag{5.4.16}$$

where $c_n=\begin{cases}1 & \text{if } \mathbf{P}_r\in\mathbb{C}_{2,n}\\ 0 & \text{otherwise}\end{cases}$; $\mathbf{F}_{\Lambda,i}$ and $\mathbf{F}_{\psi,i}$ are the repulsive and detour forces for the i^{th} obstacle, respectively; and $\mathbf{v}_{L,N}=K_1\dot{\mathbf{E}}+\sum_{n=1}^{N}c_n\left(K_{3,n}\dot{\Lambda}_n^{*}\mathbf{u}_{\Lambda,n}+K_{5,n}\dot{\psi}_n^{*}\mathbf{u}_{\psi,n}\right)$. With (5.4.16), the force field is piecewise. The VFF for multiply obstacles (5.4.16) is also continuous at boundaries of every $\mathbb{C}_2$. The proof is presented in Appendix B:

5.5 Stability analysis for the new VFF

5.5.1 The piecewise Lyapunov function for the new VFF

Since the robot is modeled as a point mass, stability of our VFF implies stability of the robot motion. Our stability analysis is based on Lyapunov's second method

(Lyapunov 1892). We will analyze the stability for the case of a single obstacle in this section. The stability analysis for the case of multiple obstacles is presented in Appendix B. The following form of Lyapunov function candidate will be used:

$$V_i(\mathbf{X}_i) = \tfrac{1}{2}\mathbf{X}_i^{\mathrm{T}}\mathbf{Q}_i\mathbf{X}_i \tag{5.5.1}$$

where $\mathbf{X}_i$ is the state vector of the VFF inside $\mathbb{C}_i$ and $\mathbf{Q}_i$ is a positive definite matrix. Recall that our VFF is a piecewise force field. When $\mathbf{P}_r \in \mathbb{C}_1$, $\mathbf{F}_v = K_1\mathbf{E} + K_2\dot{\mathbf{E}}$ only depends on $\mathbf{E}$ and $\dot{\mathbf{E}}$ so the state vector is $\mathbf{X}_1 = \begin{bmatrix}\mathbf{E}^{\mathrm{T}} & \dot{\mathbf{E}}^{\mathrm{T}}\end{bmatrix}^{\mathrm{T}}$. The only equilibrium point is the origin. This is proven in Appendix B. Note that being at the origin of the state space is equivalent to the robot being stopped at its goal. When $\mathbf{P}_r \in \mathbb{C}_2$, the VFF state vector is $\mathbf{X}_2 = \begin{bmatrix}\mathbf{E}^{\mathrm{T}} & \dot{\mathbf{E}}^{\mathrm{T}} & \Lambda & \dot{\Lambda}^* & \psi & \dot{\psi}^*\end{bmatrix}^{\mathrm{T}}$. The piecewise Lyapunov function candidate for our VFF is:

$$V = \begin{cases} V_1 = \tfrac{1}{2}\mathbf{X}_1^{\mathrm{T}}\mathbf{Q}_1\mathbf{X}_1, & \text{if } P_r \in \mathbb{C}_1 \\ V_2 = \tfrac{1}{2}\mathbf{X}_2^{\mathrm{T}}\mathbf{Q}_2\mathbf{X}_2, & \text{if } P_r \in \mathbb{C}_2 \end{cases} \tag{5.5.2}$$

where $\mathbf{Q}_2 = \begin{bmatrix}K_1^2 & K_1^2 & M_r K_1 & M_r K_1 & K_3^2 & h_\Lambda & K_5^2 & h_\psi\end{bmatrix}\cdot\mathbf{I}_8$; $h_\Lambda = M_r K_3 \dfrac{(d-r_3)^2}{(r_2-d)^2}$ and $h_\psi = M_r K_5 \dfrac{d}{(r_2-d)^2}$ are positive time varying coefficients since $d > r_3$; and $\mathbf{Q}_1 = \begin{bmatrix}K_1^2 & K_1^2 & M_r K_1 & M_r K_1\end{bmatrix}\cdot\mathbf{I}_4$. Note that when $\mathbf{P}_r \in \mathbb{C}_3$ the robot is commanded to stop, and the VFF is not used. Johansson (2002) stated in chapter 4 of his book that a piecewise Lyapunov function candidate must be continuous to analyze the stability of the

piecewise system. Eq. (5.5.2) is continuous at the boundary between $\mathbb{C}_1$ and $\mathbb{C}_2$. This will now be proven:

When $\mathbf{P}_r \in \mathbb{C}_1$

$$\lim_{d\to r_2, P_r\in\mathbb{C}_1} V = \lim_{d\to r_2} V_1 = \lim_{d\to r_2} \tfrac{1}{2}K_1^2\mathbf{E}^{\mathrm{T}}\mathbf{E} + \tfrac{1}{2}M_r K_1 \dot{\mathbf{E}}^{\mathrm{T}}\dot{\mathbf{E}} = \tfrac{1}{2}K_1^2\mathbf{E}^{\mathrm{T}}\mathbf{E} + \tfrac{1}{2}M_r K_1 \dot{\mathbf{E}}^{\mathrm{T}}\dot{\mathbf{E}} \quad (5.5.3)$$

When $\mathbf{P}_r \in \mathbb{C}_2$

$$\lim_{d\to r_2, P_r\in\mathbb{C}_2} V = \lim_{d\to r_2} V_2 = \lim_{d\to r_2}\left(\tfrac{1}{2}K_1^2\mathbf{E}^{\mathrm{T}}\mathbf{E} + \tfrac{1}{2}M_r K_1 \dot{\mathbf{E}}^{\mathrm{T}}\dot{\mathbf{E}} + \tfrac{1}{2}K_3^2\Lambda^2 + \tfrac{1}{2}M_r K_3 h_\Lambda \dot{\Lambda}^{*2} + \tfrac{1}{2}K_5^2\psi^2 + \tfrac{1}{2}M_r K_5 h_\psi \dot{\psi}^{*2}\right) \quad (5.5.4)$$

Substituting (5.4.10a), (5.4.10b), (5.4.12a) and (5.4.12b) into (5.5.4) gives:

$$\lim_{d\to r_2, P_r\in\mathbb{C}_2} V = \tfrac{1}{2}K_1^2\mathbf{E}^{\mathrm{T}}\mathbf{E} + \tfrac{1}{2}M_r K_1 \dot{\mathbf{E}}^{\mathrm{T}}\dot{\mathbf{E}} = \lim_{d\to r_2, P_r\in\mathbb{C}_1} V \quad (5.5.5)$$

Eq. (5.5.5) proves the continuity of (5.5.2).

5.5.2 The stability analysis for the piecewise Lyapunov function candidate

In this subsection, the stability analysis results for (5.5.2) will be summarized. The detailed steps are given in Appendix B. The first derivative of (5.5.2) is

$$\dot{V} = \begin{cases} \dot{V}_1 = -bK_1^2\dot{\mathbf{E}}^{\mathrm{T}}\dot{\mathbf{E}} & \text{if } P_r \in \mathbb{C}_1 \\ \dot{V}_2 = -b\left\|\mathbf{v}_L\right\|_2^2 - \mathbf{v}_L{}^{\mathrm{T}}\mathbf{F}_L + \Omega & \text{if } P_r \in \mathbb{C}_2 \end{cases} \quad (5.5.6)$$

where

$$b = \frac{K_2}{K_1} = \frac{K_4}{K_3} = \frac{K_6}{K_5}; \quad (5.5.7)$$

and

$$\Omega = -K_1\dot{\mathbf{E}}^{\mathrm{T}}\left(K_3\Lambda\mathbf{u}_\Lambda + K_5\psi\mathbf{u}_\psi\right) - K_3^2\frac{2\dot{d}(r_2-d)}{d-r_3}\Lambda + \tfrac{1}{2}M_r K_3\dot{K}_\Lambda\dot{d}^2 + K_5\dot{\psi}^*\left(\mathbf{a}_o^{\mathrm{T}}\mathbf{u}_\psi - K_1\mathbf{E}^{\mathrm{T}}\mathbf{u}_\psi\right) - K_3\dot{\Lambda}\left(K_1\mathbf{E}^{\mathrm{T}}\mathbf{u}_\Lambda - \mathbf{a}_o^{\mathrm{T}}\mathbf{u}_\Lambda\right) + \tfrac{1}{2}M_r K_5(r_2-d)^2\dot{d}\dot{\Phi}^2 - M_r K_5(r_2-d)d\dot{d}\dot{\Phi}^2 - K_5^2(r_2-d)\dot{d}\Phi^2 \quad (5.5.8)$$

When $\mathbf{P}_r \in \mathbb{C}_1$, from (5.5.6), the function $\dot{V}_1$ is negative semi-definite. Applying Lasalle's theorem (Spong and Vidyasagar 1989), if $\dot{V}_1 \equiv 0$ then $\dot{\mathbf{E}} \equiv 0$ and hence $\ddot{\mathbf{E}} \equiv 0$. From the robot's dynamics, if $\ddot{\mathbf{E}} = 0$ then $\mathbf{F}_v = 0$. Since $\dot{\mathbf{E}} = 0$ and $\mathbf{F}_v = 0$, (5.4.2) and (5.4.7) imply that $\mathbf{E} = 0$. Hence we can conclude the new VFF is asymptotically stable at the origin subject to the condition of the robot being inside $\mathbb{C}_1$.

To make $\dot{V}_2$ negative semi-definite, we need to set:

$$\mathbf{F}_L = \begin{cases} \dfrac{\Omega\mathbf{u}_L}{\|\mathbf{v}_L\|_2} & \Omega > 0 \\ 0 & \Omega \le 0 \end{cases} \quad (5.5.9)$$

Comparing to (5.4.6), we have:

$$K_L = \begin{cases} \Omega & \Omega > 0 \\ 0 & \Omega \le 0 \end{cases} \quad (5.5.10)$$

Then the first derivative of V_2 is:

$$\dot{V}_2 = \begin{cases} -b\|\mathbf{v}_L\|_2^2 & \Omega > 0 \\ -b\|\mathbf{v}_L\|_2^2 + \Omega & \Omega \le 0 \end{cases} \quad (5.5.11)$$

We have:

$$\dot{V}_2 \le -b\|\mathbf{v}_L\|_2^2 \le 0. \quad (5.5.12)$$

When $\mathbf{P}_r \in \mathbb{C}_2$, according to (5.5.12), $\dot{V}_2 \le 0$ and $\dot{V}_2 = 0$ only when $\mathbf{v}_L = 0$. It will now

be explained how the stabilizing virtual force, $\mathbf{F}_L$, prevents $\mathbf{v}_L = \mathbf{0}$ except for the case $\dot{\mathbf{E}} = \mathbf{0}$, $\dot{\Lambda} = 0$ and $\dot{\psi}^* = 0$. When $\mathbf{v}_L \to 0$, and $\dot{\mathbf{E}} \neq \mathbf{0}$, $\dot{\Lambda}^* \neq 0$ and $\dot{\psi}^* \neq 0$, the magnitude of $\mathbf{F}_L$ will increase significantly since $K_L \neq 0$ and the denominator of (5.5.6) will approach zero. The force will drive the robot such that $\|\mathbf{v}_L\|_2 \neq 0$. The proof is presented in Appendix B. If $\dot{\mathbf{E}} = \mathbf{0}$, $\dot{\Lambda} = 0$ and $\dot{\psi}^* = 0$, this means that the robot is stopped and the obstacle is stationary or stopped. Therefore, $\dot{d} = 0$ and $\dot{\Phi} = 0$. In this case, we can see that $\Omega = 0$ since every element in (5.5.8) is a function of either $\dot{\mathbf{E}}$, $\dot{\Lambda}^*$, $\dot{\psi}^*$, $\dot{d}$ or $\dot{\Phi}$. Then $K_L = 0$ from (5.5.9) and $\mathbf{F}_L = \mathbf{0}$. At this condition, if $\mathbf{F}_v = \mathbf{F}_a + \mathbf{F}_\Lambda + \mathbf{F}_\psi = \mathbf{0}$ is also true, then the robot is stopped and will not be restarted. Since $\dot{\mathbf{E}}$, $\dot{\Lambda}^*$, $\dot{\psi}^*$ are zeros, according to the force functions of $\mathbf{F}_a$, $\mathbf{F}_\Lambda$ and $\mathbf{F}_\psi$, $K_1\mathbf{E}+K_3\Lambda\mathbf{u}_\Lambda+K_5\psi\mathbf{u}_\psi = \mathbf{0}$. This condition is rare and only appears when the vectors $K_1\mathbf{E}$, $K_3\Lambda\mathbf{u}_\Lambda$ and $K_5\psi\mathbf{u}_\psi$ make up a right triangle, as shown in Fig. 5.10. This rare situation is an equilibrium point and is termed a local minima point in the mobile robot navigation literature (for examples of local minima see: Deng *et al.* 2010). From the above analysis, we can conclude $\dot{V}_2$ is negative semi-definite, and that $\dot{V}_2 = 0$ will rarely occur.

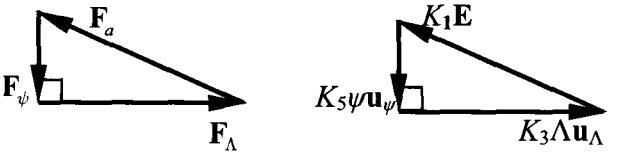


Fig. 5.10. The conditions of $\dot{\mathbf{E}}=0$, $\dot{\Lambda}^*=0$, $\dot{\psi}^*=0$ and $K_1\mathbf{E}+K_3\Lambda\mathbf{u}_\Lambda+K_5\psi\mathbf{u}_\psi = \mathbf{0}$

Recall that it is still necessary to prove (5.4.14). Note that every element in (5.5.8) is a function of either Λ, $\dot{\Lambda}^*$, ψ, or $\dot{\psi}^*$. Therefore, from (5.4.10a), (5.4.10b), (5.4.12a) and (5.4.12b), we have:

$$\lim_{d \to r_2} \Omega = 0 \tag{5.5.13}$$

According to (5.5.9) and (5.5.6),

$$\lim_{d \to r_2} \mathbf{F}_L = \mathbf{0} \tag{5.5.14}$$

Therefore, (5.4.14) is proven.

From the above analysis, we can conclude that the VFF is asymptotically stable at the origin when $\mathbf{P}_r \in \mathbb{C}_1$ and stable in sense of Lyapunov when $\mathbf{P}_r \in \mathbb{C}_2$. The Lyapunov function candidate (5.24) is a Lyapunov function of the new VFF. The stability analysis can be also expanded to include multiple obstacles. If the robot is in the active regions of N obstacles, the VFF state is

$$\left[\mathbf{E}^{\mathrm{T}},\ \dot{\mathbf{E}}^{\mathrm{T}},\ \Lambda_1,\ \dot{\Lambda}_1,\ \psi_1,\ \dot{\psi}_1,\ \ldots\ ,\Lambda_N\ \ \dot{\Lambda}_N\ \ \psi_N\ \ \dot{\psi}_N\right]^{\mathrm{T}}.$$

Similar to (5.5.2), the Lyapunov function for the VFF with N obstacles can be built. An example for $N = 2$ is presented in Appendix B.

5.6 Performance criterion for mobile robot navigation

5.6.1 Arrival time criterion

The arrival time T_{arrive} is the time consumed by the robot to reach its goal. This time is a measure of the efficiency of the navigation algorithm when avoiding obstacles. The normalized arrival time criterion is defined as:

$$H_T = \frac{T^*_{arrive}}{T_{arrive}} \tag{5.6.1}$$

where T^*_{arrive} is the nominal arrival time and T^*_{arrive} is defined as the time consumed by the mobile robot reaching its goal without obstacles. Larger H_T implies the better performance.

5.6.2 Avoidance criterion

The second criterion is the avoidance criterion. It is based on the shortest distance between the external profiles of the robot model, $d_{s,\min}$ and the models of the obstacles during the avoidance. A larger value of $d_{s,\min}$ means the robot stays farther from the obstacle and had a better avoidance performance. Note that we here use the shortest distance d_s rather than the centre-centre distance d used in Section 5.3. The reason is some researchers (for example see: Borenstien and Koren 1991) used d_s, and the centre-centre distance can be easily transformed to this distance. If the obstacle is a human, we have:

$$d_s = d - \rho_r - \rho_h . \tag{5.6.2}$$

For the case of N obstacles, the avoidance criterion is defined as:

$$H_D = \frac{1}{N}\sum_{i=1}^{N} \frac{d_{s,\min,i}}{d^*_{s,\min,i}} \qquad d_{s,\min,i} > 0 \tag{5.6.3}$$

where $d_{s,\min,i}$ is the shortest distance between the robot and the i^{th} obstacles; $d^*_{s,\min,i}$ is the nominal distance to normalize the index and can be computed as $d^*_{s,\min,i} = r_{2,i} - \rho_r - \rho_{obs,i}$

for the i^{th} obstacle. In (5.6.3), we require $d_{s,\min,i} > 0$. If $d_{s,\min,i} \le 0$, the robot collides with the obstacle. The avoidance criterion should be zero. If the robot does not enter the active region of any obstacle $d_{s,\min,i} = d^*_{s,\min,i}$ and this index equals one.

5.6.3.Energy saving criterion

The third criterion is the energy saving criterion. The energy wasted by the motors used to drive the robot is:

$$L_{energy} = \int_0^{T_{arrive}} \mathbf{i}^{\mathrm{T}} \mathbf{R}_a \mathbf{i} dt \tag{5.6.4}$$

where $\mathbf{i}$ is the vector of the motor currents, and $\mathbf{R}_a$ is a diagonal matrix of the motor winding resistances. If we assume the same type of motors are used, the winding resistances are identical and equal to R_a. Then the normalized energy saving index is:

$$H_E = \frac{L^*_{energy}}{L_{energy}} = \frac{\int_0^{T^*_{arrive}} (\mathbf{i}^*)^{\mathrm{T}} \mathbf{R}_a \mathbf{i}^* dt}{\int_0^{T_{arrive}} \mathbf{i}^{\mathrm{T}} \mathbf{R}_a \mathbf{i} dt} = \frac{R_a \int_0^{T^*_{arrive}} (\mathbf{i}^*)^{\mathrm{T}} \mathbf{i}^* dt}{R_a \int_0^{T_{arrive}} \mathbf{i}^{\mathrm{T}} \mathbf{i} dt} = \frac{\int_0^{T^*_{arrive}} (\mathbf{i}^*)^{\mathrm{T}} \mathbf{i}^* dt}{\int_0^{T_{arrive}} \mathbf{i}^{\mathrm{T}} \mathbf{i} dt} \tag{5.6.5}$$

where $\mathbf{i}^*$ is the current that the robot uses to reach its goal without obstacles. Larger H_E means less energy is wasted by the motors.

5.6.4 Total performance criterion for mobile robot navigation

The total performance criterion for mobile robot navigation includes the arrival time, the avoidance performance and the energy saving performance as follows:

$$\begin{aligned} H &= k\left(w_1 H_T + w_2 H_D + w_3 H_E\right) \\ s.t.\ & w_1 + w_2 + w_3 = 1 \end{aligned} \tag{5.6.6}$$

where $k = \begin{cases} 1 & d_{s,\min,i} > 0 \ \ \forall i = 1,2,...,N \\ 0 & d_{s,\min,i} \le 0 \ \ \exists i = 1,2,...,N \end{cases}$ and w_1, w_2 and w_3 are the criterion weights. Their values are dependent on the application of the robot. In this thesis, we used: $w_1 = w_2 = w_3 = \frac{1}{3}$ to make the three criteria have equal weight. Note from (5.6.1), (5.6.3) and (5.6.5), the criterion (5.6.6) has no units. A higher value of H means a better performance. The value of H will be less than or equal to one. Only when the robot does not avoid any obstacles, H will be equal to one. Using this criterion, the sensitivity analysis for the VFF gains on the navigation performance has been studied. The results are presented in Appendix C. This criterion will be also used in the next chapter to compare the navigation performances of the new VFF-based algorithm with two important conventional VFF-based algorithms.

5.7 Simulations with an unpredictable human path

In this section, simulations with five well-known navigation algorithms from the two categories (described in Chapter 2) are compared with the new VFF-based algorithm. An unpredictable human path is used. The simulation paths and d are presented in Fig. 5.11 and Fig. 5.12, respectively. In simulations, a holonomic robot starts from [4, 0] m and navigates past the walking human to reach its goal at [0, 0] m. The walking human initially starts from [0.6, 0] m with a [1, 0] m/s velocity. After 1 s, the human slows down with a [-1, 0] m/s^2 acceleration until stopped, and then moves sideways with a [0, 1] m/s^2 acceleration. After the human's velocity reaches [0, 1] m/s, the human keeps this velocity. In Fig. 5. 11 and Fig. 5.12, VO denotes the *velocity obstacle* navigation

algorithm (Large, Laugier, and Shiller 2005). RRT denotes the *rapidly-exploring random trees* algorithm (Bruce and Veloso 2006). D* denotes the *Field D** algorithm (Ferguson and Stentz 2006). Those three algorithms belong to the first category. Masoud denotes the APF-based algorithm of Masoud (2007). Ge & Cui denotes the VFF-based algorithm presented by Ge and Cui (2002). New VFF denotes the new VFF introduced in Section 5.4. Those three algorithms belong to the second category. During the simulations, the future motion of the human is unknown for all six algorithms. The walking direction change of the human is unpredictable. From Fig. 5.11 and Fig. 5.12, we can see that the three algorithms in the first category incur collisions (*i.e.* $d < \rho_r + \rho_h = 0.2 + 0.4$ m)

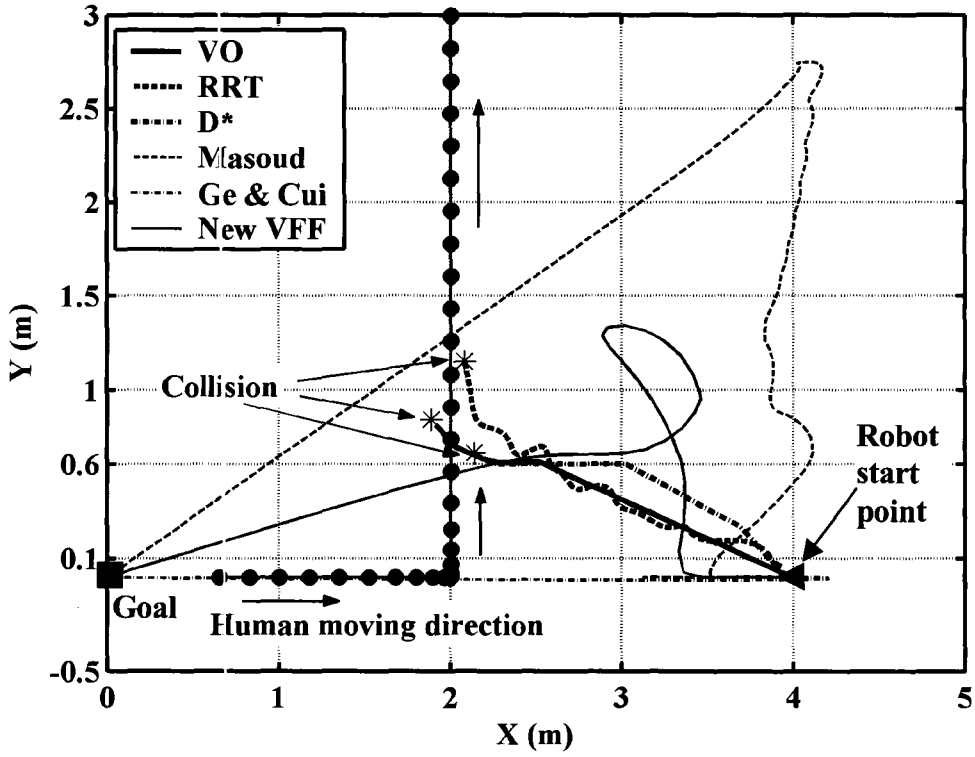


Fig. 5.11. Comparison of the simulation results for the six algorithms with a motion-unpredictable human.

after the human changes its direction. Since the future position information of the human is not required for the algorithms in the second algorithms, they successfully complete the navigation. Therefore, the algorithms in the second category are suitable to avoid the motion-unpredictable obstacles, such as humans. The performance criteria are listed in Table 5.1. The new VFF-based algorithm possesses a 2% better performance. To further evaluate the new VFF-based algorithm, more simulations will be performed and compared in the next chapter.

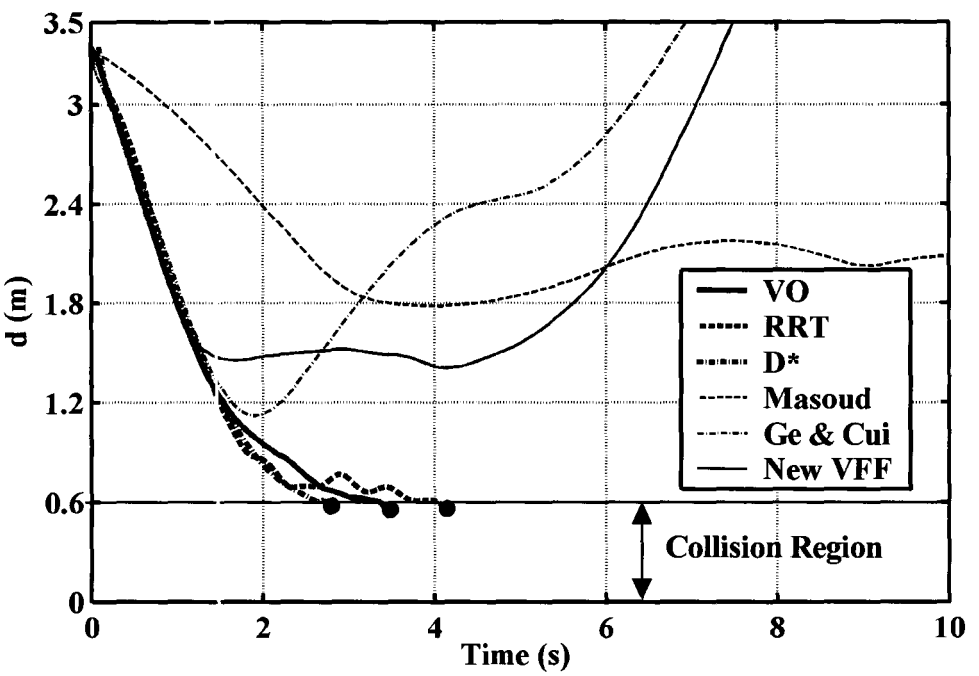


Fig. 5.12. Comparison of the simulation results for distances between the robot and the human, *d*.

Table 5.1. The performance criteria for three algorithms in the second category

Methods	H	L_{energy} (J)	$d_{cir.min}$ (m)	T_{arrive} (s)
Masoud	0.48	13.1	1.8	14.0
Ge & Cui	0.48	5.0	1.1	10.5
New VFF	0.49	7.7	1.4	9.9

5.8 Conclusions

In this chapter a novel VFF-based mobile robot navigation algorithm was proposed. It features improved functions for the repulsive and detour virtual forces, and a new stabilizing virtual force. It is continuous (which should diminish path oscillations), will work in the collinear condition, and has proven stability. In particular, it is asymptotically stable at the goal position when the robot is inside region $\mathbb{C}_1$. To address another unsolved problem, methods were presented for sizing the active and critical regions for human, circular and rectangular obstacles. Using an unpredictable human path, simulations with five well-known navigation algorithms are compared with the simulation of the new VFF-based algorithm. Since the future motions of the obstacles are not required for algorithms in the second category, they are suitable for avoiding motion-unpredictable obstacles. In the next chapter, we will conduct more simulations to compare with the two existing second-category algorithms. Finally, three criteria were introduced for quantifying the navigation performance. The performance of the proposed navigation algorithm will be evaluated via simulations and experiments in the next two chapters.

Chapter 6

Navigation simulations and experiments with a holonomic mobile robot

6.1 Introduction

In this chapter, navigation simulations and experiments with a holonomic mobile robot will be presented. Since the performance of a VFF-based navigation system is influenced by sensor errors and tracking errors of the robot (*e.g.* due to sliding of the wheels), experiments are required to evaluate the performance of the new VFF. The experimental setup built for this purpose will be described in the next section. Four different configurations with two stationary obstacles and a walking human will be used. The simulation results with the new VFF-based algorithm will be compared with two important conventional VFF-based navigation algorithms. The performance criteria introduced in Section 5.7 of Chapter 5 will be utilized for the comparisons. The experiment results with the four obstacle configurations will be compared with the simulation results. Another simulation will be also performed to reveal the limit of the conventional algorithms. The chapter ends with a conclusions section.

6.2 Experimental setup and procedure

6.2.1 Design of a holonomic robot

To study the navigation performance of the new VFF-based algorithm, a three-wheel holonomic robot was built, as shown in Fig. 6.1. The total mass of the robot is

M_r = 4.4 kg. The shape of the robot is modeled as a circle with a 0.2 m radius, *i.e.* $\rho_r = 0.2\,\text{m}$. Three omni-directional wheels are used (Kornylak model FXA308 CAT-TRAK). Each wheel is driven by a DC brushed motor (Maxon model RE40-148877) through a 10:1 ratio worm gear set. The advantages for using the worm gear sets rather than planetary gears are to reduce the robot's mass and to reduce the size of the robot since the wheel shaft and the motor shaft can be aligned vertically. The disadvantage is at least two wheels must be rotating at same time when operating the robot; otherwise significant sliding of the wheels occurs since the gears are not back-drivable. As an example, when only one wheel is running, the robot will rotate. Since the rotations of the other two wheels are blocked by their worm gear sets, they will slide as the robot rotates. This design creates more driving power but is less energy efficient than a design employing back-drivable gears. The linear velocity limit of the robot is $\overline{V}_{mr} = 0.7\,\text{m/s}$, and its acceleration limit is $\overline{a}_{mr} = 10\,\text{m/s}^2$. These limits were selected based on the speed

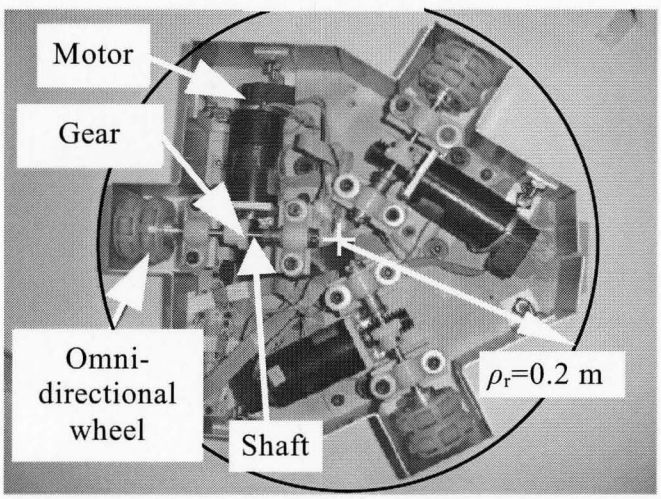


Fig. 6.1. The holonomic robot used in the navigation experiments

and torque limits of the DC motors driving the robot wheels. Since the kinematics of the robot and design of the robot wheel-motor controllers are not key components of this thesis, they are presented in Section D.3 of Appendix D.

6.2.2 Experimental setup

As illustrated in Fig. 6.2a, the experimental setup includes a colour video camera (PGR model DR2 HICOL) employed to capture images of the human and robot. Colour patches are attached to the human's shoulder and the robot at known Z heights, as shown in Fig. 6.2b. With a standard PC, the colour patches are segmented from the captured image. The geometric centroids of the segmented patches are computed from their segmented pixels. The X-Y positions of the human and robot are reconstructed from the centroids of the colour patches using the camera calibration matrix and the known patch

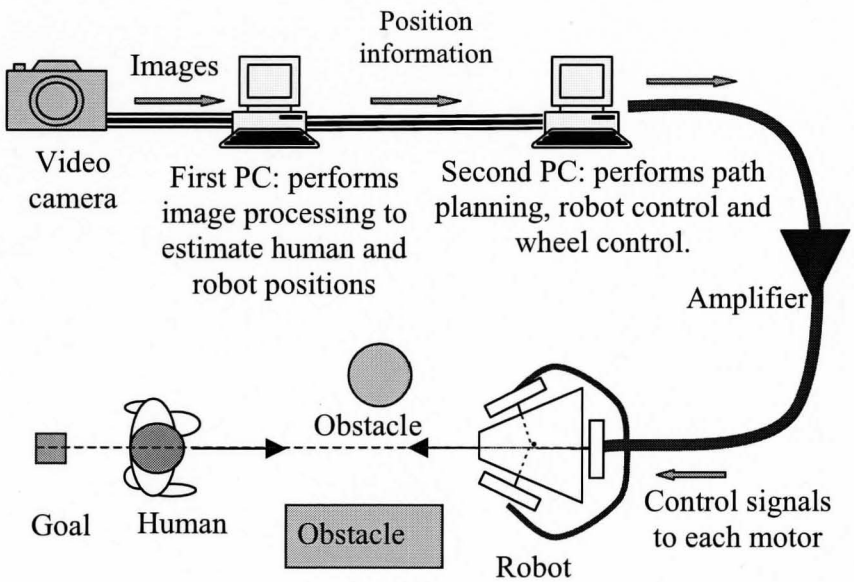


Fig. 6.2a. The experimental setup

heights (see Section D.4 of Appendix D). The Tsai camera calibration method (Tsai 1987) is used to obtain the calibration parameters. The sampling frequency of the image processing is 16.7 Hz. The maximum error of the position measurements is 0.025 m. The detailed calibration procedure and results are presented in Appendix D. To reduce the number of the colour patches and the computing burden of the image processing, the position(s) of stationary obstacle(s) are predefined. This is also applicable in manufacturing and office environments; the positions of stationary obstacles, such as machines and tables are fixed and can be predefined. The position data are transferred via serial communication to a second PC that performs the robot control. The second PC computes the virtual forces and controls the three wheels of the mobile robot. The amplified motor currents and motor position encoder signals are transmitted to/from the robot using an umbilical cable.

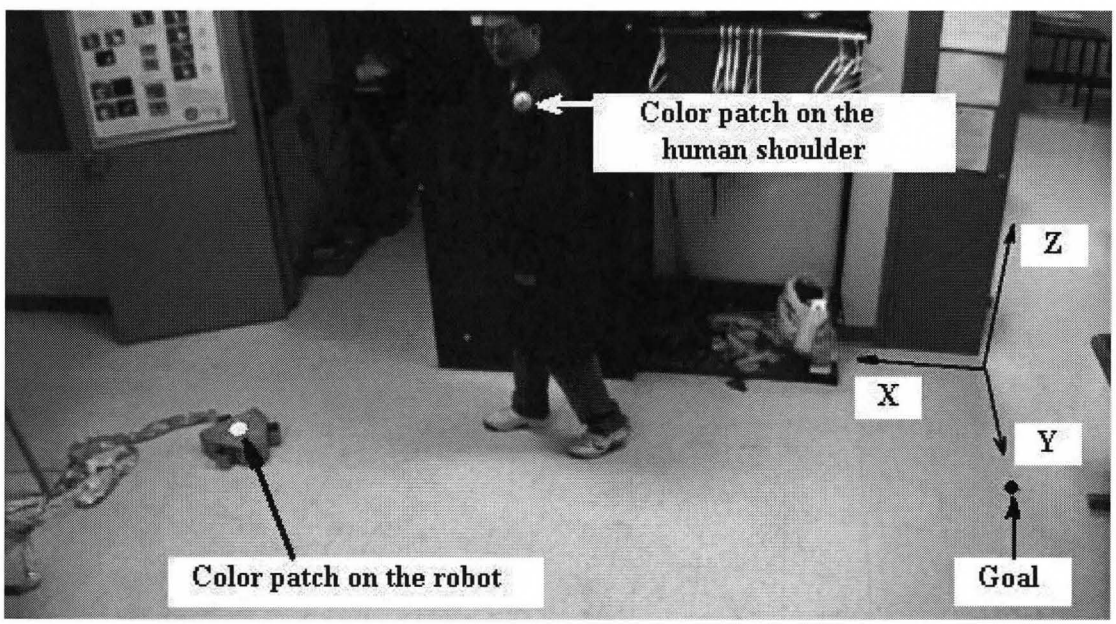


Fig. 6.2b. Colour patches on human shoulder and robot for detecting the human and robot positions.

6.2.3 Experimental procedure

Four different obstacle configurations will be used in the simulations and experiments, including the basic configurations: avoiding one stationary obstacle (termed Configuration 1), avoiding a walking human in the collinear condition (termed Configuration 2), avoiding two obstacles through a narrow passage (termed Configuration 3), and the more challenging configuration: avoiding two stationary obstacles and a walking human (termed Configuration 4). The simulation and experiment results with the new VFF-based algorithm for those four configurations will be compared with simulations with two important existing VFF-based algorithms (Ge and Cui 2002; and Masoud 2007).

In Configuration 1 (as shown in Fig. 6.3a), a stationary obstacle is used. This obstacle is modeled as a circular shape with a 0.1 m radius. The centre of this obstacle is located at [2.45, 0.595] m. Its active and critical regions are shown in Fig. 3.3a as $\mathbb{C}_{2,circular}$ and $\mathbb{C}_{3,circular}$, respectively. Their sizes will be derived in Section 6.3. The robot will start moving from position coordinates [3.9, 0.6] m and navigate past the obstacle to reach its goal at [0, 0.6] m. This obstacle configuration is near to collinear condition. The collinear condition is the worst case as discussed in Section 5.3. We chose the nearly collinear condition to allow a comparison to be made with the algorithms of Ge and Cui's algorithm since they cannot handle the exactly collinear condition. In Configuration 2 (Fig. 6.3b), the human starts from position coordinates [0.6, 0.6] m, moves from left to right and then stops at [4.0, 0.6] m. At the same time, the robot starts from [3.9, 0.6] m and moves from right to left and towards its goal at [0.0, 0.6] m. This configuration is in

the collinear condition. Since the active and critical regions of the human (defined as $\mathbb{C}_2$ and $\mathbb{C}_3$) are moving, they are not drawn in this and the following figures. The sizes of those regions will be also defined in Section 6.3. In Configuration 3 (Fig. 6.3c), rectangular and circular stationary obstacles are used. The circular obstacle is centered at [2.45, 1.4] m. The rectangular stationary obstacle's corners are at [1.6, -0.3], [3, -0.3], [3, 0.3] and [1.6, 0.3] m. B_{len} = 1.4 m and B_{wid} = 0.6 m. Its active and critical regions are presented as $\mathbb{C}_{2,rect}$ and $\mathbb{C}_{3,rect}$ and their sizes will be calculated in Section 6.3. The distance between this obstacle and the robot used for computing the repulsive and detour forces must be the shortest distance between the geometric centre of the robot model and the exterior profile of the rectangle, as described in Section 5.4. In this figure, we can see that the two active regions $\mathbb{C}_{2,rect}$ and $\mathbb{C}_{2,circular}$ have an overlap. A narrow passage is created with the overlap. In Configuration 4 (Fig. 6.3d), three obstacles, the walking human, the circular and rectangular obstacles, are employed. The positions of the

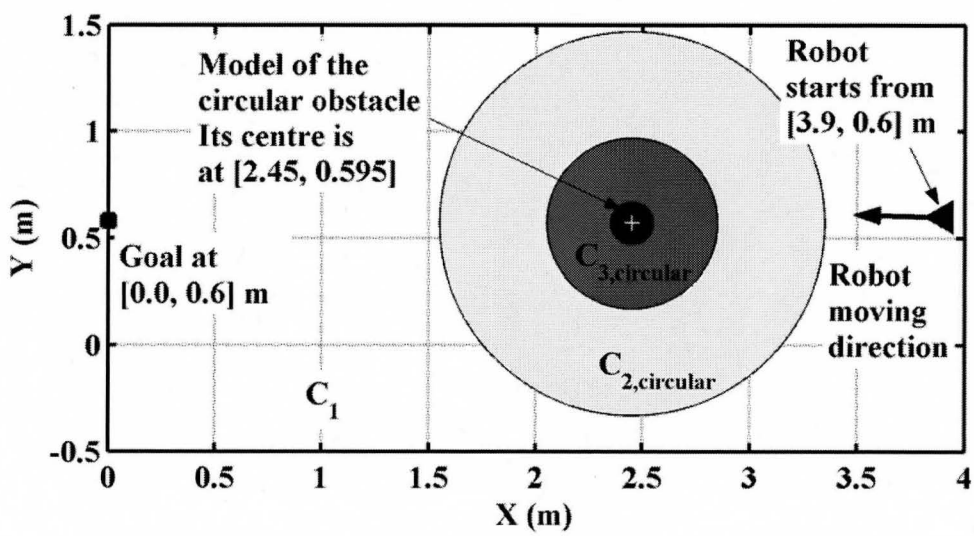


Fig. 6. 3a. Obstacle configuration 1: avoiding a stationary obstacle.

rectangular and circular obstacles are same as in Configuration 3. The human starts from position [0.6, 0.6] m, moves from left to right and then stops at [4.0, 0.6] m.

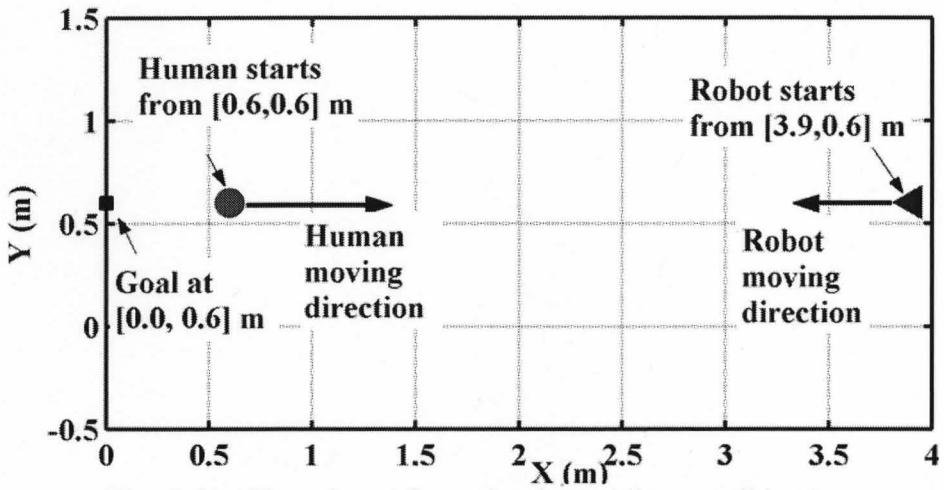


Fig. 6. 3b. Obstacle configuration 2: avoiding a walking human.

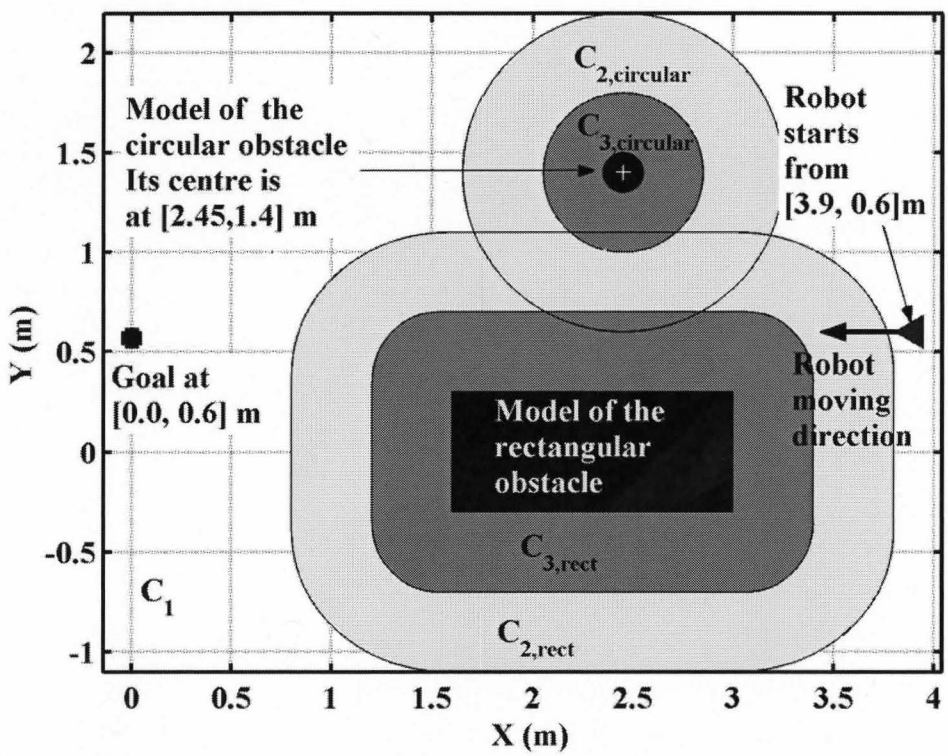


Fig. 6. 3c. Obstacle configuration 3: avoiding two stationary obstacles.

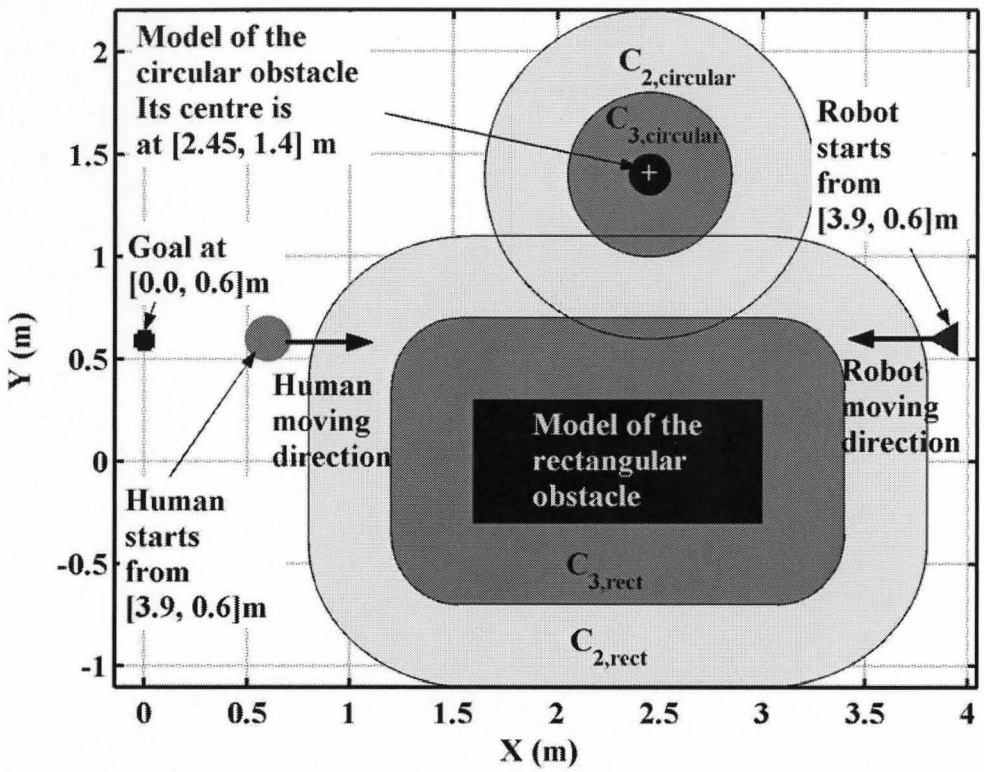


Fig. 6. 3d. Obstacle configuration 4: avoiding a walking human and two stationary obstacles.

These four configurations are fundamental for navigation systems in manufacturing and service applications. More complex configurations can be built by combining these configurations. By comparing with the two important conventional VFF-based algorithms for these four configurations, the navigation performance of the new VFF-base algorithm will be investigated.

6.2.4 The VFF gains used in the simulations and experiments

The VFF gains influence the performance of a navigation system. To make the comparison fair, the gains for the new algorithm and the two conventional algorithms are

optimized. We used a numerical method to obtain the optimized gains. The detailed optimization procedure is presented in Section C.2 of Appendix C. Note that the gains of Ge & Cui's algorithm is optimized for Configurations 1, 3 and 4. Since it cannot handle the collinear condition, the human-robot collision will happen in Configuration 2. The gains of the new VFF and Masoud's algorithm are optimized for Configurations 1, 2 and 3. The gains for the three VFFs are listed in Table 6.1. Note that virtual force functions of the two conventional VFFs were presented in subsection 2.2.4 of Chapter 2. In Table 6.1, the items in the notation column show the names used in remaining figures and tables to denote the three algorithms.

In following experiments and simulations, the optimized VFF gains for the new VFF are not used. The first reason is the sensitivity of the total performance criterion to the VFF gains is small (1.9% for K_1, 0.5 % for K_3, 1.8% for K_5 and 0.4% for b). The numerical sensitivity analysis is presented in Section C.3 of Appendix C. The second reason is that the gains are hard to be optimized in actual applications due to the unlimited diversity of the obstacle configurations. Non-optimal gains will trend to be used in manufacturing and service environments. So in this chapter, we want to perform the experiments with a set of non-optimal gains to emulate the navigations in those environments. The gains of the new VFF used in the simulations and experiments are: $K_1 = 3$ N/m, $K_3 = 20$ N/m, $K_5 = 17$ N/m^2 (manually tuned based on the simulations of Configurations 1, 2 and 3) and $b = 0.25$ (manually tuned to eliminate the path oscillation caused by the positioning errors of the vision system). Then, K_1, K_3, K_5 and b are reduced by 43%, 2%, 93% and %63 from the optimized values in Table 6.1, respectively.

Table 6.1. The optimized gains of the conventional VFFs and the new VFF

Source	Notation	*Attractive force gain*	*b*	*Repulsive force gain*	*Detour force gain*
New VFF + non-optimal gains	New VFF	3	0.3	20	17
New VFF + the optimized gains	New VFF*	4	0.7	20	240
Masoud (2007) + the optimized gains	Masoud*	5	-	24	36
Ge and Cui (2002) + the optimized gains	Ge & Cui*	1	0.8	14	-

6.3 Sizes of active and critical regions of the obstacles

In the four obstacle configurations, three obstacles are used: a walking human, a stationary circular obstacle and a stationary rectangular obstacle. In this section, we will derive the sizes of their active and critical regions in accordance with the procedure from Section 5.3.

To obtain the size of the critical region of a walking human, we have:

$$\overline{\delta}_{r1} = \overline{V}_{mr} T_s + \tfrac{1}{2}\overline{a}_{mr}\Delta t_1^2 = 0.7\times 0.06 + \tfrac{1}{2}\times 10\times 0.07^2 = 0.067 \text{ m, and} \qquad (6.3.1a)$$

$$\overline{\delta}_{h1} = \overline{V}_h \times (T_s + \Delta t_1) = 1.0\times(0.06+0.07) = 0.13 \text{ m}, \qquad (6.3.1b)$$

where $\Delta t_1 = \overline{V}_{mr} / \overline{a}_{mr} = 0.7/10 = 0.07$ s. With (5.3.1):

$$\hat{r}_3 > \overline{\delta}_{h1} + \overline{\delta}_{r1} = 0.13 + 0.067 = 0.197 \text{ m}. \qquad (6.3.2)$$

We selected $\hat{r}_3 = 0.2$ m. The average step length of a human is around 0.8 m (Martin and Marshi 1992). The value can be used as the diameter of the human cylinders, *i.e.* $\rho_h = 0.4$ m. From (5.3.3):

$$r_3 = \hat{r}_3 + \rho_h + \rho_r = 0.2 + 0.4 + 0.2 = 0.8\,\text{m} \tag{6.3.3}$$

For the size of the active region, with (5.3.6b) and $t_a = \overline{V}_{mr} / \overline{a}_{mr} = 0.7/10 = 0.07\,\text{s}$:

$$\Delta t_2 > \frac{r_3 - \frac{1}{2}\overline{a}_{mr} t_a^{\,2}}{\overline{V}_{mr}} + t_a = \frac{0.8 - \frac{1}{2}\times 10 \times 0.07^2}{0.7} + 0.07 = 1.179\,\text{s}. \tag{6.3.4}$$

Let us set $\Delta t_2 = 1.18\,\text{s}$. With (5.3.7) and $\Delta t_2 = 1.18\,\text{s}$:

$$\begin{aligned} r_2 > \overline{\delta}_{hx2} + \overline{\delta}_{rx2} &= \overline{V}_{mr} \times (\Delta t_2 + T_s) + \overline{V}_h \times (\Delta t_2 + T_s) \\ &= 0.7 \times (1.18 + 0.06) + 1.0 \times (1.18 + 0.06) = 2.09\,\text{m}. \end{aligned} \tag{6.3.5}$$

In our implementation: $r_2 = 2.1\,\text{m}$.

The circular stationary obstacle is modeled with a 0.1 m radius, *i.e.* $\rho_{circular} = 0.1\,\text{m}$. The size of its critical region is derived by using (5.3.8a) as follows:

$$r_{3,cir} > \overline{\delta}_{r1} + \rho_{circular} + \rho_r \tag{6.3.6}$$

Since $\overline{\delta}_{r1} = 0.07\,\text{m}$ from (6.1a) and $\rho_r = 0.2\,\text{m}$, we have

$$r_{3,cir} > \overline{\delta}_{r1} + \rho_{circular} + \rho_r = 0.067 + 0.1 + 0.2 = 0.367\,\text{m}.$$

We selected $r_{3,cir} = 0.4\,\text{m}$. For the active region, using the steps presented in subsection 5.3.4, from (5.3.8b) we have:

$$r_{2,cir} > \overline{\delta}_{rx2,circular} \tag{6.3.7}$$

where $\overline{\delta}_{rx2,circular} = \overline{V}_{mr} \times (\Delta t_{2,circular} + T_s)$; and from (5.3.9b) $\Delta t_{2,circular}$ satisfies

$$\Delta t_{2,circular} > \frac{r_{3,cir} - \frac{1}{2}\overline{a}_{mr} t_a^{\,2}}{\overline{V}_{mr}} + t_a = \frac{0.4 - \frac{1}{2}\times 10 \times 0.07^2}{0.7} + 0.07 = 0.608\,\text{s} \tag{6.3.8}$$

Thus, $\Delta t_{2,circular} = 0.61\,\text{s}$. With (6.3.7) and $\Delta t_{2,circular} = 0.61\,\text{s}$,

$$r_{2,cir} > \overline{\delta}_{rx2,cir,} = \overline{V}_{mr} \times \left(\Delta t_{2,circular} + T_s\right) = 0.7 \times \left(0.61 + 0.06\right) = 0.47 \text{ m}. \quad (6.3.9)$$

With (6.3.9), we obtained the minimum values of $r_{2,cir}$. If the minimum values are used the robots will stop, change direction, then resume motion to avoid this stationary obstacle. To reduce the time cost for reaching the goal the robot should remain in motion during the avoidance. This can be accomplished by increasing the value of $r_{2,cir}$. In our implementation we selected $r_{2,cir} = 0.8$ m to make the holonomic robot remain in motion.

For the rectangular obstacle, from (5.3.11a) and (6.3.1), we have

$$\hat{r}_{3,rect} > \overline{\delta}_{r1} = 0.067 \text{ m} \quad (6.3.10)$$

In our implementation, $\hat{r}_{3,rect} = 0.2$ m was selected. So:

$$r_{3,rect} = \hat{r}_{3,rect} + \rho_r = 0.2 + 0.2 = 0.4 \text{ m} \quad (6.3.11)$$

Since in the experiments, the robot will approach the obstacle from its shorter side rather than its longer side as shown in Fig. 5.4, we will use its width to determine the active region. So $B = \frac{1}{2} B_{wid} = 0.3$ m and from (5.3.14), $\Delta t_{2,rect}$ must satisfy

$$\Delta t_{2,rect} > \frac{r_{3,rect} + B - \frac{1}{2}\overline{a}_{mr} t_a^2}{\overline{V}_{mr}} + t_a = \frac{0.4 + 0.3 - \frac{1}{2} \times 10 \times 0.07^2}{0.7} = 1.035 \text{ s}. \quad (6.3.12)$$

Thus, $\Delta t_{2,rect} = 1.04$ s. With (5.3.12) and $\Delta t_{2,rect} = 1.04$ s:

$$r_{2,rect} > \overline{V}_{mr} \times \left(\Delta t_{2,rect} + T_s\right) = 0.7 \times \left(1.04 + 0.07\right) = 0.78 \text{ m}. \quad (6.3.13)$$

In our implementation, $r_{2,rect} = 0.8$ m. Note that in the simulations with the two conventional VFF-based algorithms, the sizes of active regions for the three obstacles are also defined as 2.1 m, 0.8 m and 0.8 m, respectively.

6.4 Simulation and experiment results and analysis

6.4.1 Configuration 1: Avoiding a stationary circular obstacle

The simulation and experiment results for Configuration 1 are presented in this subsection. We will use $d_{circular}$ to denote the centre-to-centre distance between the robot and the circular obstacle. The performance criteria are listed in Table 6.2. In this table, $d_{cir,\min}$ is the minimum $d_{circular}$ during the navigation. From Fig. 6.4, we can see that with all three VFFs the robot avoids the obstacle. However, it is clear that Ge and Cui's algorithm incurs severe oscillating paths of the robot; this is because their VFF is discontinuous at the boundary of $\mathbb{C}_{2,circular}$. Although the VFF in Masoud's algorithm is also discontinuous, the path has no oscillation due to the relatively large detour force (the maximum detour force is 29 N in comparison with less than 3 N in the new VFF). The new VFF produces the smooth path due to its continuous VFF. In Fig. 6.5, e is the distance between the robot and its goal (*i.e.* $e = \|\mathbf{E}\|_2$). Note that $d_{circular} \geq 0.58$ m in the simulation is always greater than $r_{3,cir} = 0.4$ m therefore the robot never enters the critical region of the obstacle. We show the line $d_{circular} = 0.3$ m in Fig. 6.5 since if $d_{circular} \leq 0.3$ m a collision will happen (*i.e.* $\rho_r + \rho_{circular} = 0.2 + 0.1 = 0.3$ m). Due to the lack of path oscillations, the robot with the Masoud's VFF and the new VFF takes less time to reach its goal (T_{arrive} = 6.3 s and T_{arrive} = 6.7 s, repspectively) than T_{arrive} = 13.7 s for the VFF of Ge and Cui (2002). Since generating a smaller force consumes less energy, the robot with the new VFF wastes less energy than with Masoud's VFF, as

shown by the L_{energy} values in Table 6.2. However, the centre-to-centre distance between the robot and obstacle is shorter than the other two VFFs. This is because the two conventional VFF are discontinuous. When the robot comes into $\mathbb{C}_{2,circular}$, their repulsive forces jump to a large value to push the robot away and increase $d_{circular}$. Overall, the robot with the new VFF has a better performance in terms of H than Ge and Cui's VFF (19% better) but has a lower H value (8% lower for the non-optimal gains and 2% lower for the optimal gains) than Masoud's VFF. Note that figures of the simulation path, e and $d_{circular}$ with the optimized gains will not be presented in this chapter.

Table 6.2. The performance criteria for Configuration 1

Methods	H	L_{energy} (J)	$d_{cir.min}$ (m)	T_{arrive} (s)
New VFF (Sim.)	0.70	3.6	0.54	6.7
New VFF* (Sim.)	0.74	3.6	0.55	6.3
Masoud* (Sim.)	0.76	4.0	0.67	6.3
Ge & Cui* (Sim.)	0.50	13.1	0.75	13.5
New VFF (Exp.)	0.55	8.9	0.59	7.4

* with optimized gains

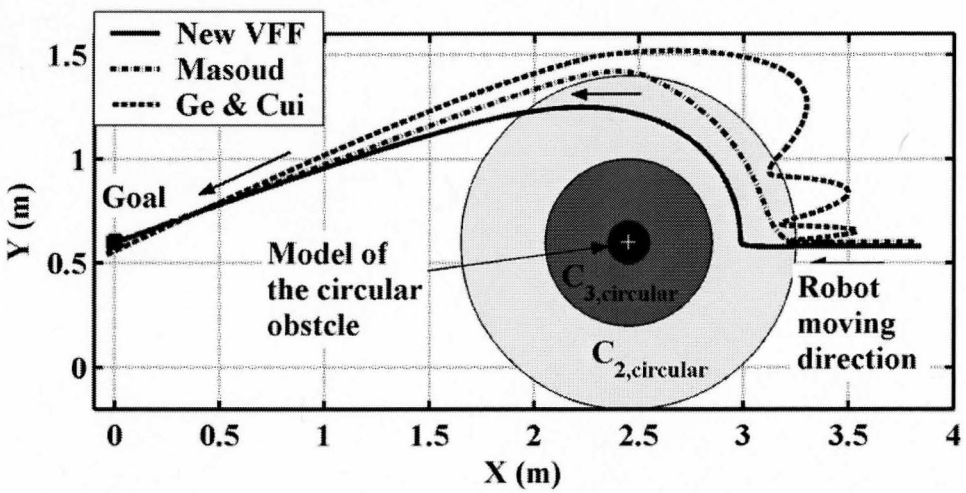


Fig. 6.4. Comparison of the simulation results for the three VFF-based algorithms for Configuration 1.

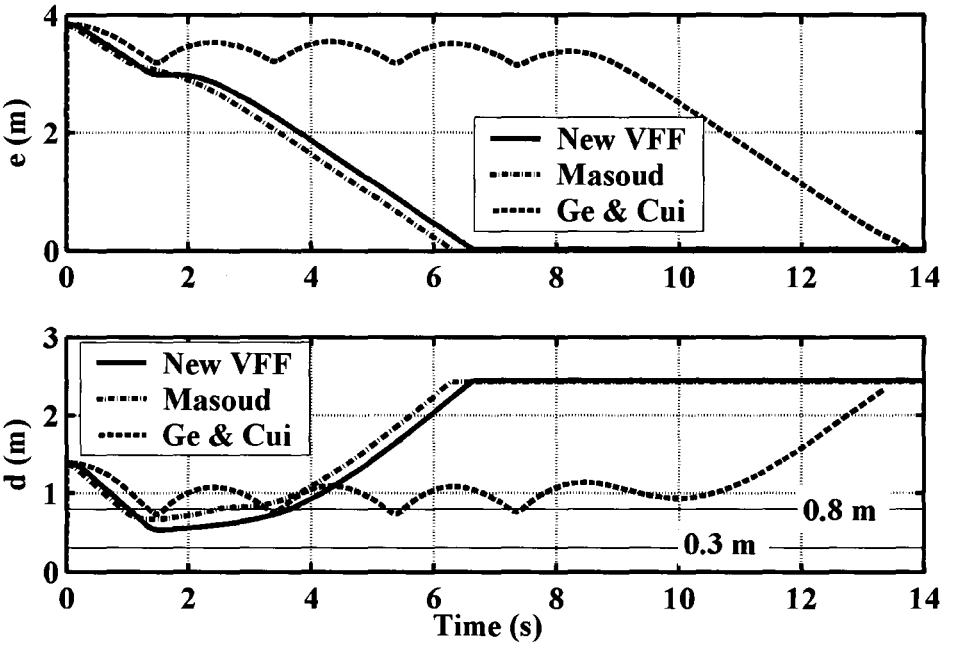


Fig. 6.5. Comparison of the simulation results for distances e and $d_{circular}$ for Configuration 1.

The comparison between the simulation and the experiment results with the new VFF are shown in Figs 6.6 and 6.7. We can see the paths for the simulation and experiment are similar. The paths and T_{arrive} discrepancy between the simulation and experiment results is primarily due to the wheels sliding on the floor. The sliding makes the robot move slower than the ideal robot and respond more slowly when a large acceleration of the robot is required, such as when the robot just starts from rest. So T_{arrive} = 7.4 s in the experiment compared with 6.7 s in the simulation. However, the distance between the robot and obstacle in the experiment (0.58 m) is larger than in the simulation (0.49 m). This distance difference is from the errors of the vision system. The errors will have a significant influence on the calculation of the velocity of the robot; this will

further influence values of the virtual forces and cause the simulation and experiment results differ. Additional experiment results and a repeatability analysis are presented in Appendix F. Observing Table 6.2, we can see that L_{energy} in the experiment is nearly double the simulation value. In the simulations, it is assumed that the dynamic models used in the wheel controller are perfect, therefore only the model-based feedforward portion of the control contributes to the motor currents. However, since the dynamic models must have errors, for example due to the varying gear friction, the PD feedback portion in those wheel controllers also contributes to the motor currents to compensate those errors. This caused more energy to be wasted in the experiment than in the simulation. Overall, H decreased 8% relative to the value in the simulation (see Table 6.2).

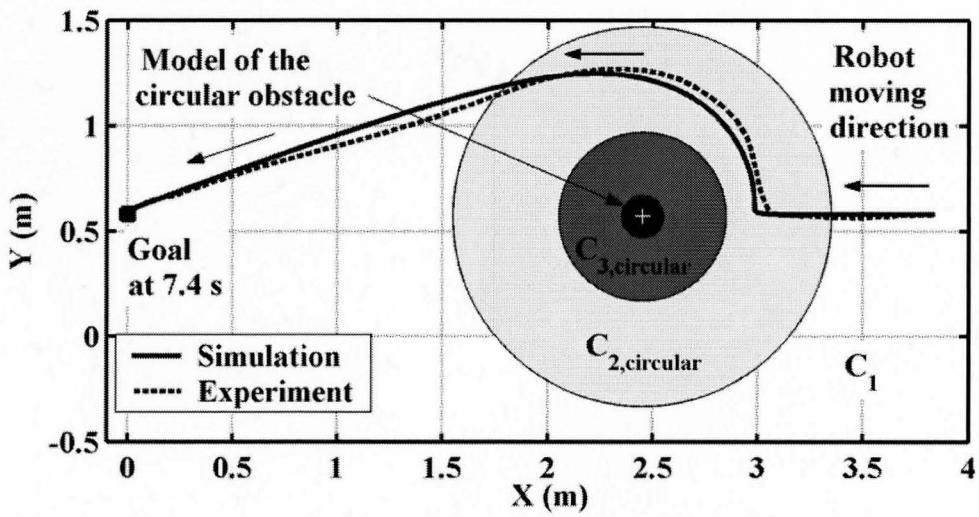


Fig. 6.6. Comparison of simulation and experiment results with the new VFF for Configuration 1.

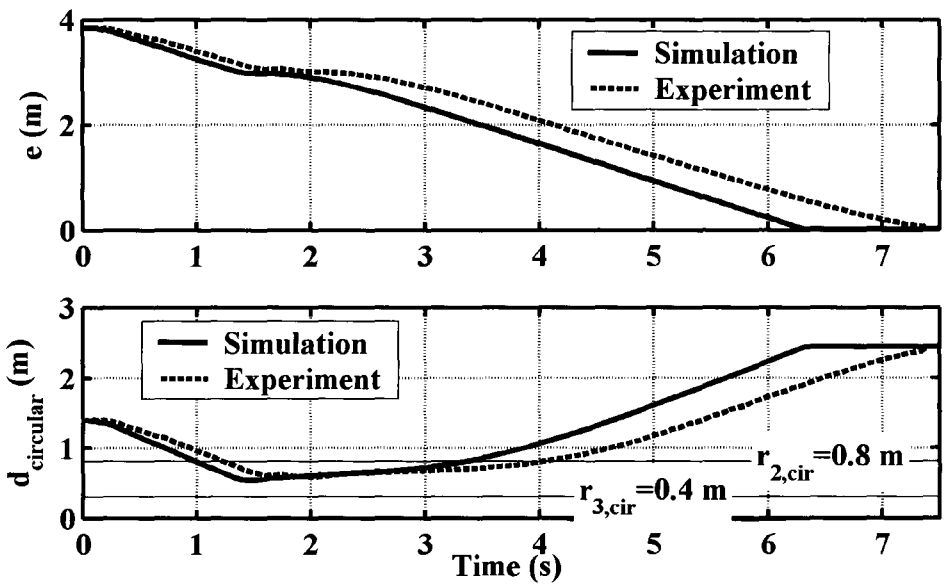


Fig. 6.7. Comparison of the simulation and experiment results with the new VFF: distances e and $d_{circular}$ for Configuration 1.

6.4.2 Configuration 2: Avoiding a walking human in the collinear condition

The simulation and experiment results for Configuration 2 are presented in this subsection. We will define d_h as the centre-to-centre distance between the human and robot. The performance criteria are listed in Table 6.3. $d_{h,\min}$ is the minimum d_h during the navigation. In Fig. 6.8, the simulation paths with the new VFF and Masoud's VFF are presented. We can see that detour forces of the two VFFs drag the robot move sideway to avoid the human. Since Ge and Cui's VFF has a zero-detour force in collinear condition, the robot collides with the human in this configuration. So the simulation path with this algorithm is not shown. The paths of the simulation and experiment with the new VFF are compared in Fig. 6.8. These two robot paths have an obvious difference in the Y-direction. We can see that the robot needs to change its moving direction sharply in

comparison with Fig. 6.6. The reason is the human's approaching reduces d_h. This acts to increase the magnitudes of the repulsive and detour forces. Then the robot needs a large acceleration to move backwards and sideways fast to avoid the human. The large acceleration causes severe sliding of the robot wheels and the path difference in the Y-direction. Fig. 6.9 shows that the measured human velocity, V_h, varies by ± 0.8 m/s, and has an average of around 1.0 m/s while walking (*i.e.* before 5.2 s). Note that the simulations use the same human path data as the experiment. $d_h > 1.1$ m in the simulation and experiment with the new VFF, and larger than r_3 =0.8 m. The robot is always out of the critical region of the human model. We also can see $T_{arrive} \approx 8.2$ s in both the simulation and experiment. For Masoud's VFF, $d_h > 1.1$ m and $T_{arrive} \approx 7.5$ s. This shorter T_{arrive} is also due to the large detour force when the robot enters $\mathbb{C}_2$ of the human. In Fig. 6.9, the d_h data from simulations of the two conventional algorithms are also presented for comparison purposes. We can see a collision will happen with Ge and Cui's VFF at t = 4 s. From Table 6.3, we can see that the robot with Masoud's VFF has a larger H value (6% larger) than with the new VFF. We will perform another simulation in subsection 6.4.5 to further compare the new VFF and Masoud's VFF for avoiding a walking human. With the new VFF, the H value for the simulation with the optimized gains is worse than the simulation with the non-optimal gains. This is because the larger value of the optimized b amplified the noise of the measured human velocities and caused larger path oscillations.

Table 6.3. The performance criteria for Configuration 2

Methods	H	L_{energy} (J)	$d_{h,min}$ (m)	T_{arrive} (s)
New VFF (Sim.)	0.51	5.8	1.1	8.2
New VFF* (Sim.)	0.47	5.9	1.3	8.7
Masoud* (Sim.)	0.54	5.1	1.1	7.5
New VFF (Exp.)	0.46	18.0	1.0	8.2

* With optimized gains

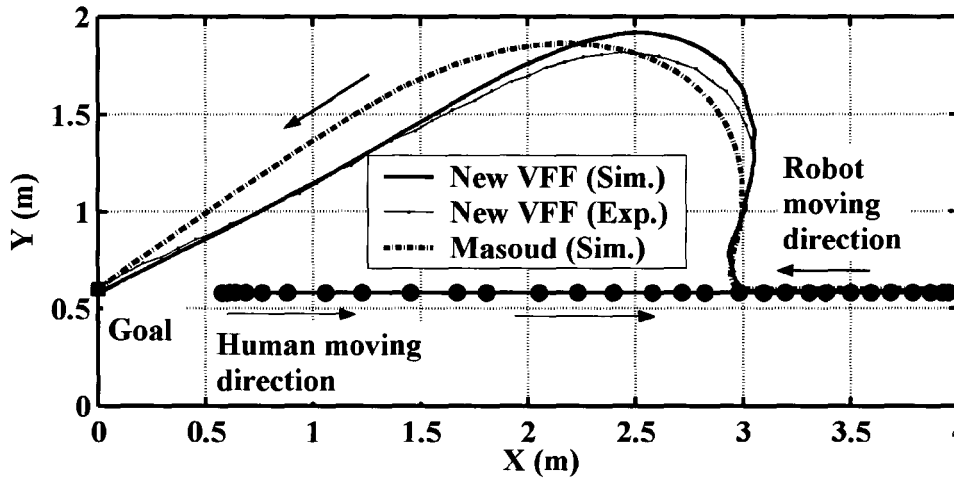


Fig. 6.8. Comparison of simulation and experiment results with the new VFF for Configuration 2. The algorithm of Ge and Cui fails to work under this condition.

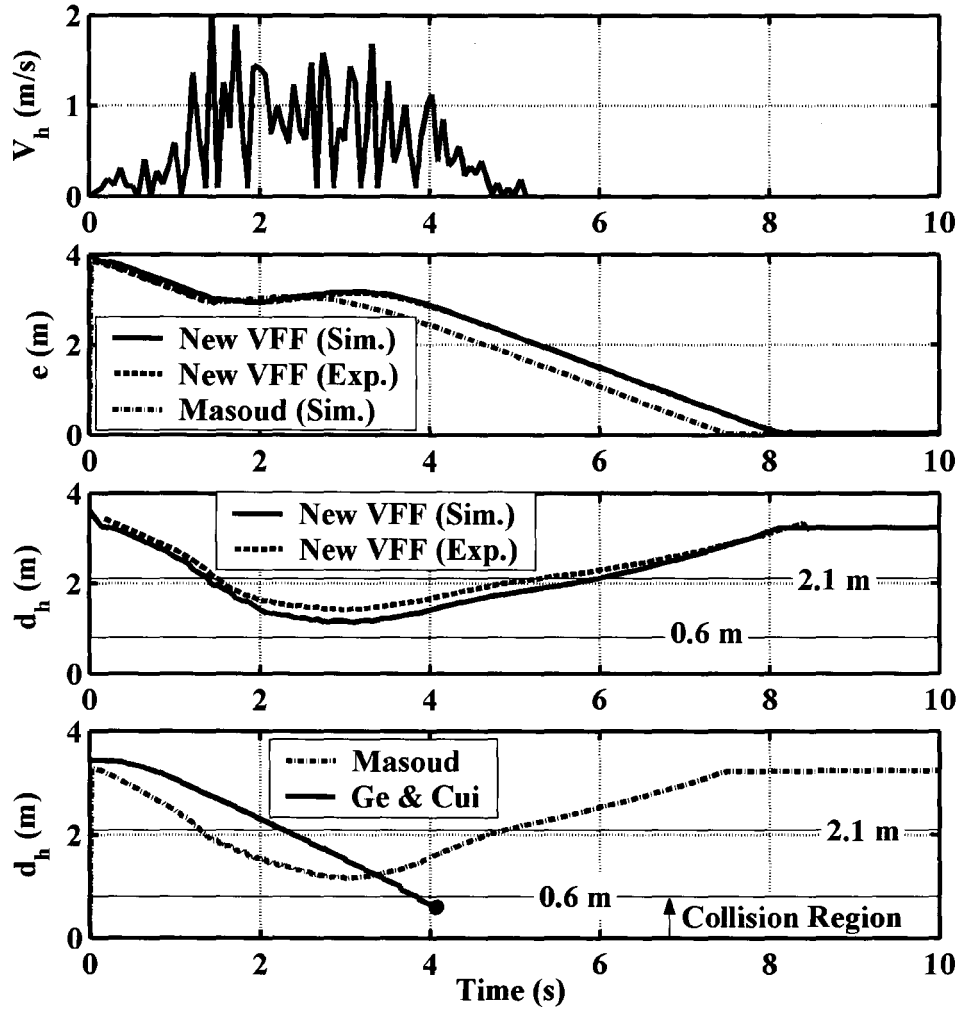


Fig. 6.9. Comparison of the simulation and experiment results with the new VFF: distances e and d_h for Configuration 2, and also with simulation results of the two conventional VFFs.

6.4.3 Configuration 3: Avoiding two stationary obstacles

Another test of the VFF algorithm is avoiding two stationary obstacles in Configuration 3. Figs. 6.10 and 6.11 display the robot avoidance paths and the distances to the goal and the two obstacles for the three VFF-based algorithms. With the new VFF,

the robot moves straight towards its goal at the beginning. Upon entering $\mathbb{C}_{2,rect}$, the robot avoids the rectangular obstacle and its path deflects towards the upper-left due to the repulsive and detour forces from this obstacle until $\mathbb{C}_{2,circular}$ for the circular obstacle is encountered. The repulsive force and detour force from the circular obstacle drives the robot towards the lower-left. This causes the path to bend downwards. Next, the robot passes between the two obstacles (*i.e.* through the intersection of the two active regions). The path is slightly oscillatory. This phenomenon was also observed by Koren and Borenstein (1991). This path oscillation is always present when two or more active regions of obstacles intersect. The oscillation is mainly generated by the repulsive forces from the obstacles. The approximate net repulsive force surface for the two obstacles is plotted in Fig. 6.12. The surface is not exact since it was generated for a fixed value of $\mathbf{V}_{mr}$ whereas the actual $\mathbf{V}_{mr}$ will vary during navigation. The figure shows that a valley is formed between the critical regions. The robot will typically enter at one side of the valley and then $\mathbf{F}_{\Lambda}$ will push it towards the centre of the valley. The momentum of the robot will move it past the centre and $\mathbf{F}_{\Lambda}$ will once again act to re-centre it. This behavior repeats, causing the path oscillations. The oscillations will be larger if the VFF is discontinuous. Similarly, if the force changes rapidly at the boundary of an active region the slope of the valley will be steeper and the oscillations will be more severe. We have designed the VFF to be continuous, and to gradually increase the force when entering an active region, in order to mitigate the path oscillation problem. Furthermore, the term $K_4\dot{\Lambda}^*$ in (5.4.3) acts in a manner similar to viscous friction and further reduces these

oscillations. Note that if the critical regions of two obstacles intersect, it is obvious the robot should not attempt to pass between them. Therefore, those two obstacles should be modeled as one obstacle by merging their critical regions, and merging their active regions.

From Fig. 6.11, the robot avoids the two obstacles with all three VFF-based algorithms: *i.e.* the distance, d_{rect}, from the robot centre to the exterior profile of the rectangular obstacle is larger than $\rho_r = 0.2$ m and $d_{circular} > \rho_{circular} + \rho_r = 0.3$ m. T_{arrive} = 6.3 s for the new VFF, which is smaller than the T_{arrive}=7.5 s produced by the VFF of Masoud (2007) and T_{arrive}=15.9 s produced by the VFF of Ge & Cui (2002). The other performance criteria are listed in Table 6.4. In this table, $d_{rect,\min}$ is the minimum d_{rect} during the navigation. From Table 6.4, the robot with Ge and Cui's VFF has a much lower H value in this configuration than the other two. With the new VFF, the robot costs less time (19% less than Masoud's VFF) and wastes less energy (28% less than Masoud's VFF) but has a shorter distance (23% shorter for $d_{rect,\min}$ and 7% shorter for $d_{cir,\min}$ less than Masoud's VFF). Finally, the H value with the new VFF that is 10% better than Masoud's VFFs.

Table 6.4. The performance criteria for Configuration 3

Methods	H	L_{energy} (J)	$d_{rect,min}$ (m)	$d_{cir,min}$ (m)	T_{arrive} (s)
New VFF (Sim.)	0.74	3.5	0.51	0.55	6.1
New VFF* (Sim.)	0.78	4.1	0.54	0.56	6.2
Masoud* (Sim.)	0.67	4.9	0.70	0.60	7.5
Ge & Cui* (Sim.)	0.55	9.2	0.76	0.75	13.3
New VFF (Exp.)	0.71	17.7	0.51	0.54	7.0

* with optimized gains

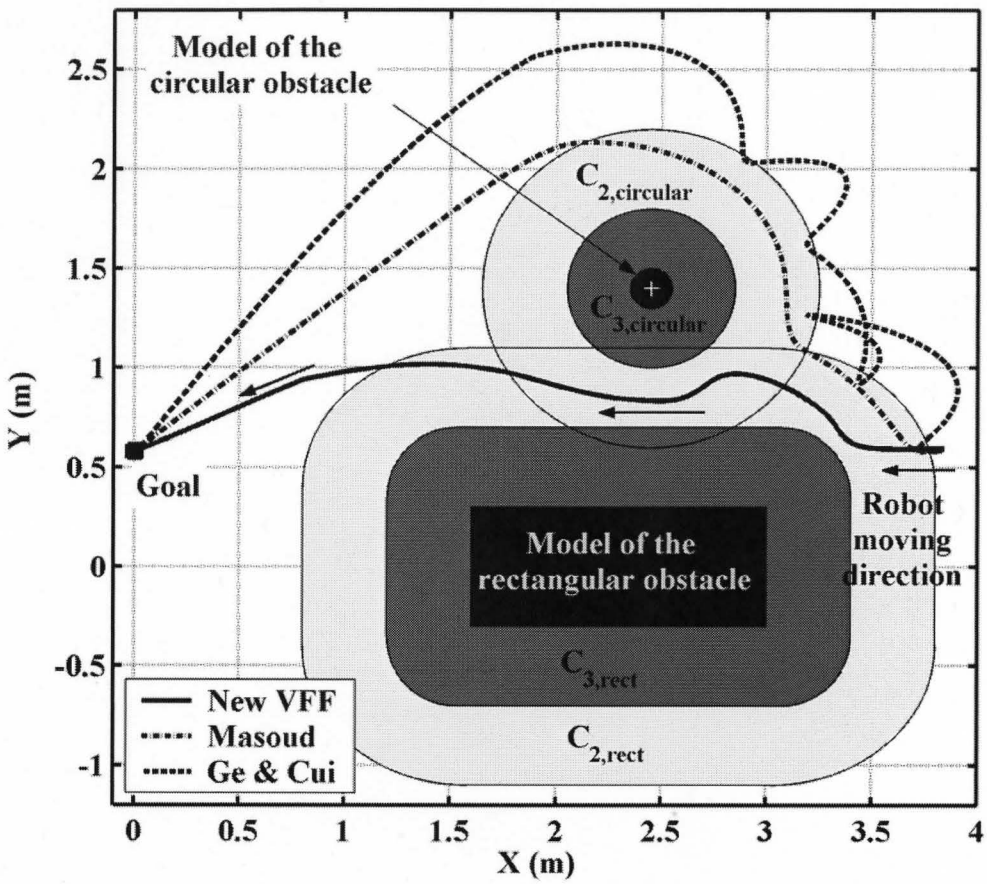


Fig. 6.10. Comparison of the simulation results for the three VFF-based algorithms for Configuration 3.

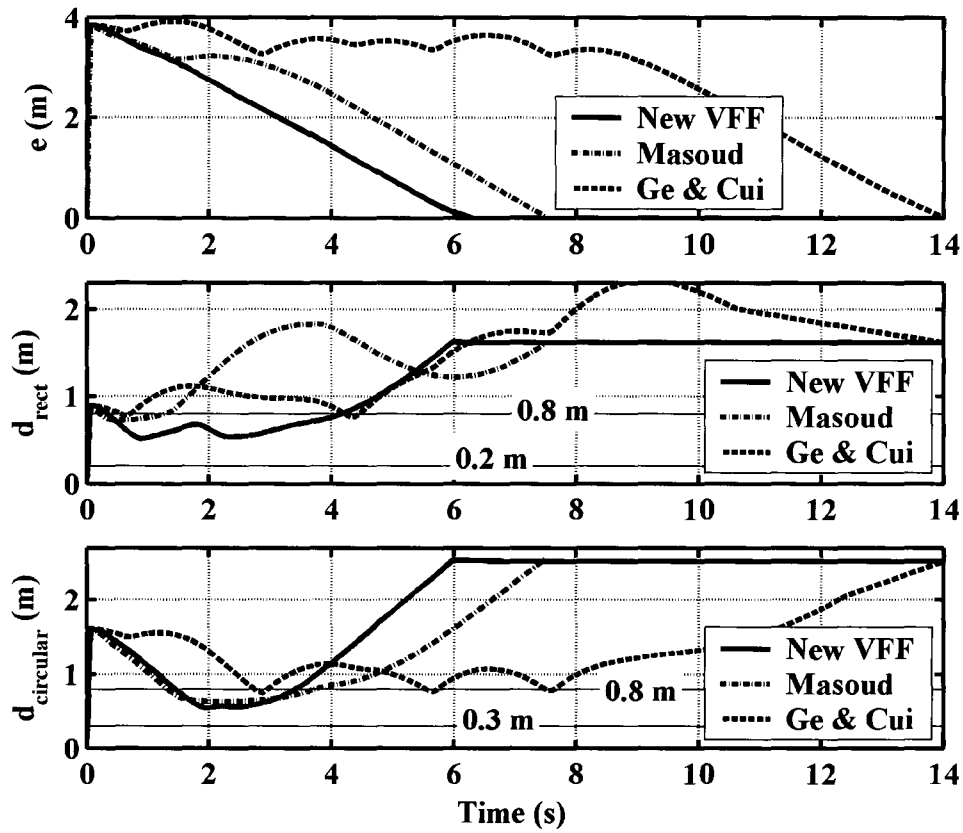


Fig. 6.11. Comparison of the simulation results for the three VFF-based algorithms: distances e and d to the rectangular and circular obstacles for Configuration 3.

In Fig. 6.13, the simulation and experiment paths for Configuration 3 with the new VFF are compared. The paths are in close agreement. In Fig. 6.14, the distances to the goal and the two obstacles vs. time are plotted. The T_{arrive} = 7.0 s in the experiment is larger than the T_{arrive} = 6.2 s in the simulation. This difference between the experiment and simulation is again caused by the wheel sliding. In the experiment, a 15% longer T_{arrive} and 290% more energy were consumed. This is also caused by the feedback PD portion of the wheel controller. No obvious difference in $d_{circular,min}$ and $d_{\text{rect,min}}$. So the H value in the experiment is 9% more than in simulation.

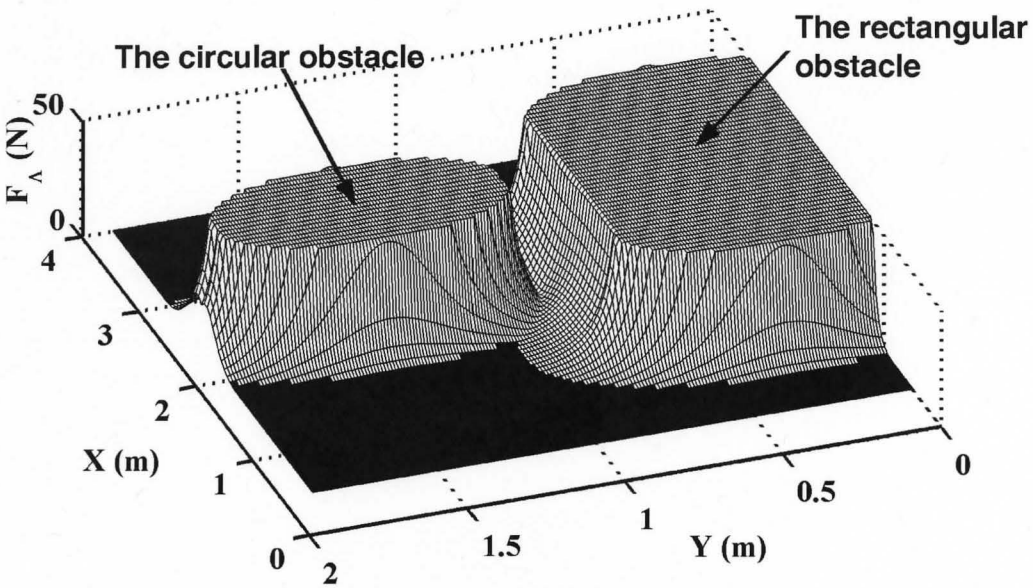


Fig. 6.12. The approximate net repulsive force surface for the circular and rectangular obstacles in Configuration 3.

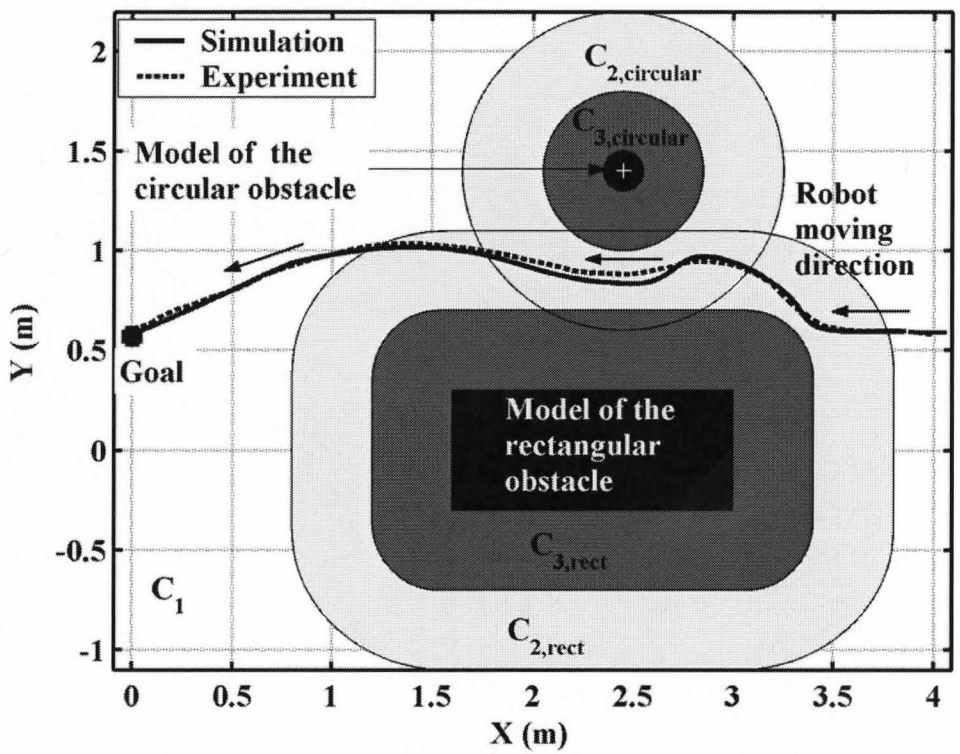


Fig. 6.13. Comparison of simulation and experiment results with the new VFF for Configuration 3.

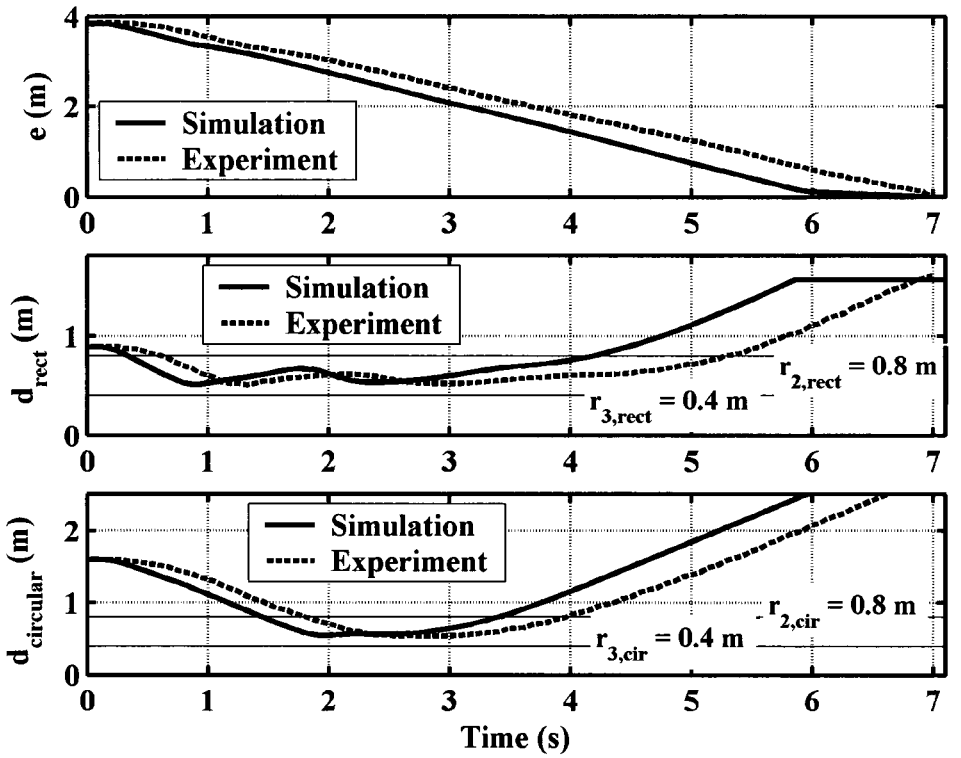


Fig. 6.14. Comparison of the simulation and experiment results with the new VFF: distances e, d_{rect} and $d_{circular}$ for Configuration 3

6.4.4 Configuration 4: Avoiding a moving human and two stationary obstacles

Another challenging test of the VFF algorithm involves avoiding two stationary obstacles and a moving human, *i.e.* Configuration 4. The results of simulations with the three VFFs are shown in Figs. 6.15 and 6.16. We can see from Fig. 6.16 that the navigation was successful for all three VFFs. The robot first avoids the rectangular obstacle, and then the human gets close so the robot moves sideways to avoid him. After that, the circular obstacle is encountered. Thus the robot is repelled by both the human and the circular obstacle and detours around them. For Ge and Cui's VFF, since the

robots are pushed backwards much farther by the repulsive forces from the human, the robots take a much longer time and more energy to reach the goal. The new VFF and Masoud's VFF produce paths without oscillation. The performance criteria data are listed in Table 6.5. The robot with Ge and Cui's algorithm also has a much lower H value. T_{arrive} = 7.9 s for using Masoud's VFF and the new VFF. They also produces a similar L_{energy} values. The new VFF also produces a 18% shorter $d_{rect,min}$, a 32% shorter $d_{cir.min}$ and a 8% shorter $d_{h,min}$. The overall performance criterion H with the new VFF is 14% lower than Masoud's VFF-based algorithms. We can see that with the new VFF the minimum distances between the robot and obstacles are also smaller than the

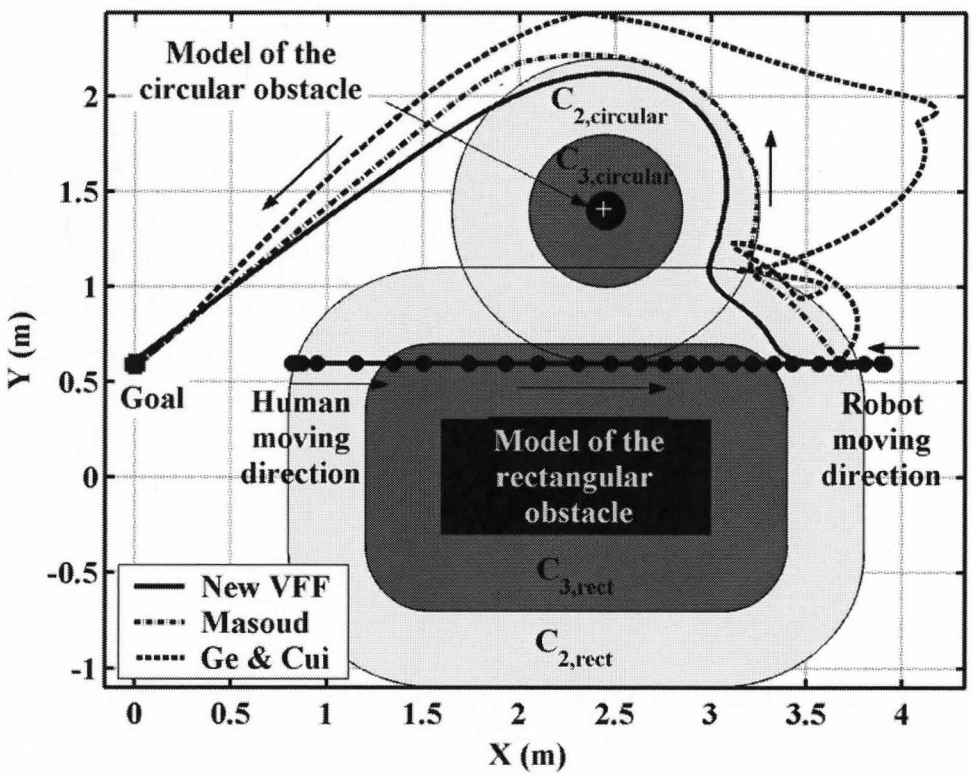


Fig. 6.15. Comparison of the simulation results for the three VFF-based algorithms for Configuration 4.

corresponding distances with the other two VFFs. Using the optimized gains in the simulation, the total performance criterion H will be decreased by 4%. This is again because the larger value of the optimized b amplified the noise of the measured human velocities and caused larger path oscillations. Note that this configuration was not used to obtain the optimized gains.

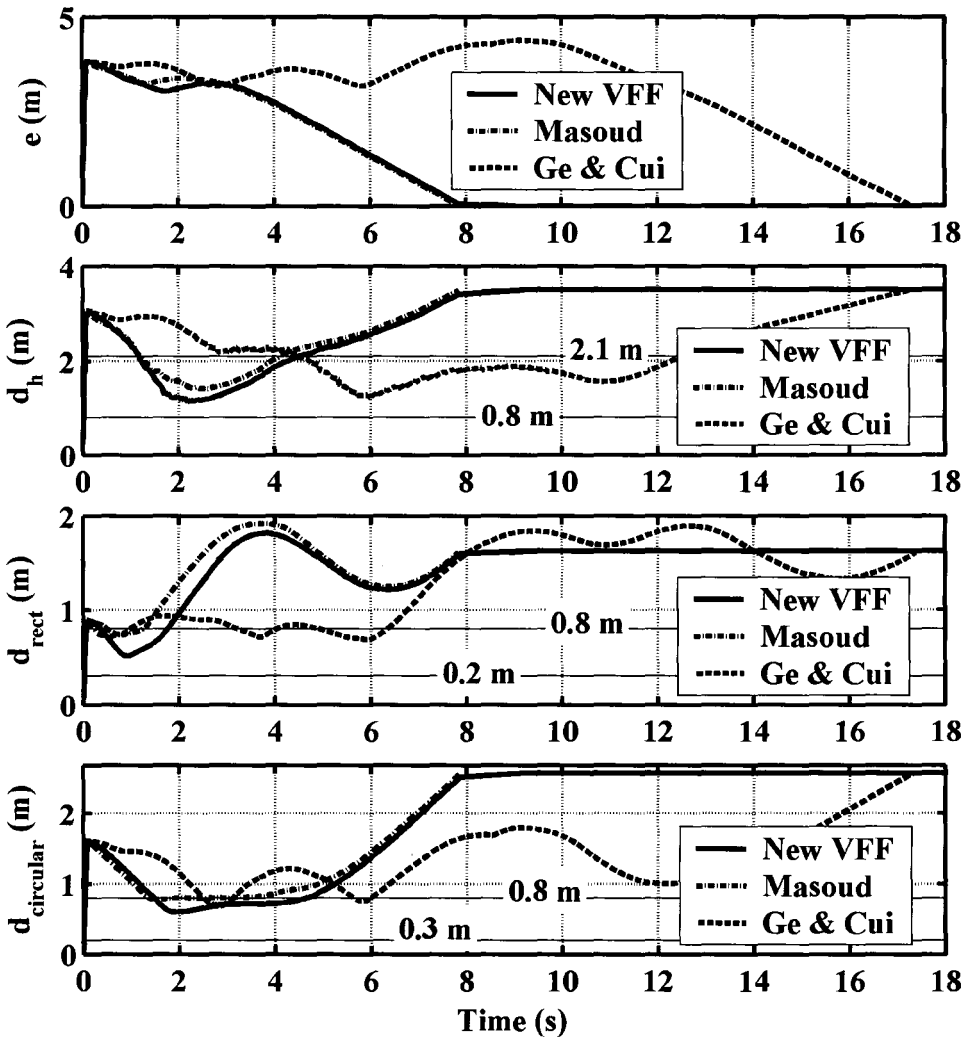


Fig. 6.16. Comparison of the simulation results for the three VFF-based algorithms: distances e and d_h, d_{rect} and $d_{circular}$ for Configuration 4.

Table 6.5. The performance criteria for Configuration 4

Methods	H	L_{energy} (J)	$d_{h,min}$ (m)	$d_{rect,min}$ (m)	$d_{cir,min}$ (m)	T_{arrive} (s)
New VFF (Sim.)	0.55	5.5	1.1	0.61	0.52	7.9
New VFF* (Sim.)	0.53	5.2	1.2	0.62	0.54	7.8
Masoud* (Sim.)	0.64	5.3	1.2	0.72	0.77	7.9
Ge & Cui* (Sim.)	0.40	9.7	1.2	0.68	0.75	17.4
New VFF (Exp.)	0.47	18.7	1.0	0.60	0.50	8.4

* with optimized gains

The simulation and experiment results are compared in Figs. 6.17 and 6.18. From Fig. 6.17 it can be observed that the two paths are similar although the robot comes closer to the circular obstacle in the experiment. In Fig. 6.18, $d_{circular}$ is significantly larger than

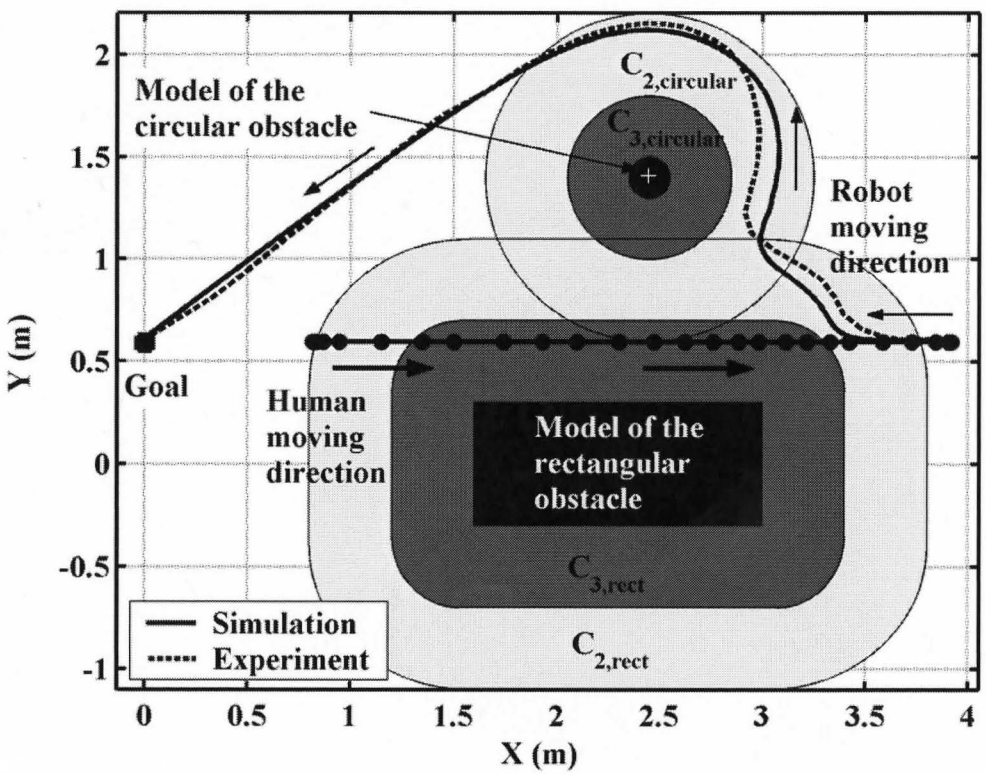


Fig. 6.15. Comparison of the simulation results for the three VFF-based algorithms for Configuration 4.

the $d_{circular}$ data for Configuration 1 in Fig. 6.8. This is because in Fig. 6.8, the avoidance is near to the worst case, the collinear condition, so the robot comes closer to (but still does not enter) the obstacle's critical region, $\mathbb{C}_{3,cir}$. From Table 6.5, we can see the criterion H in the experiment is also 2% smaller than the simulation.

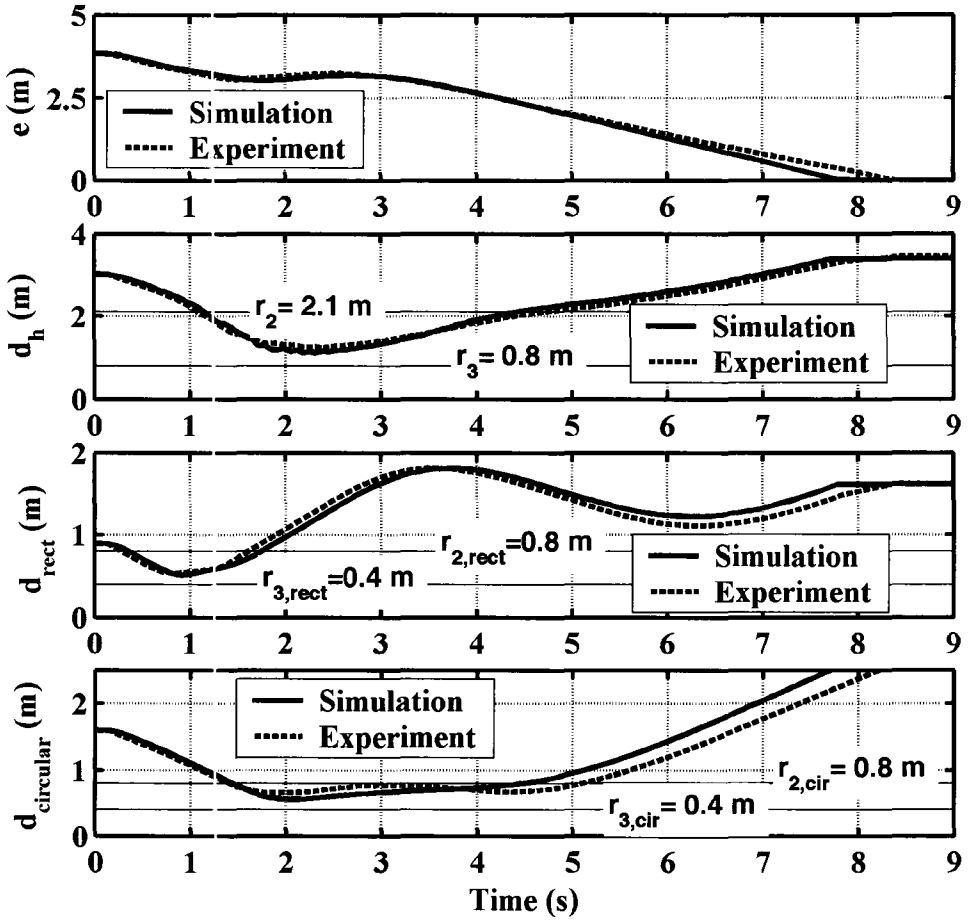


Fig. 6.18. Comparison of the simulation and experiment results with the new VFF distances e and d_h, d_{rect} and $d_{circular}$ for Configuration 4.

6.4.5 Avoiding a moving human parallel to the robot moving direction

From above simulations, the navigation performance with the new VFF-based algorithm is 21% better on average than Ge and Cui's algorithm and 4% on average lower than Masoud's algorithm. In above simulations, the collinear or the near collinear conditions are used since they are the worst cases for the new VFF-based algorithm (also for Ge and Cui's algorithm). In this simulation, the configuration for avoiding a moving human parallel to the robot moving direction is performed to compare only the new VFF and Masoud's VFF. In this configuration, the human starts from position coordinates [0.6, 1.2] m, moves from left to right and then stops at [4.0, 1.2] m. At the same time, the robot starts from [3.9, 0.6] m and moves from right to left and towards its goal at [0.0, 0.6] m. In Fig. 6.19, the navigation paths of the robot with two algorithms are presented. In Fig. 6.20, distances e and d_h are also shown. We can see that the robot with the new VFF completes the navigation without a collision and $T_{arrive} = 7.3$ s. Masoud's VFF produces a collision at $t = 2.7$ s due to the incorrect fixed detour force direction. The detour force drags the robot to move in front of the human and that action incurs a collision. To solve this problem, either the repulsive force gain needs be increased significantly or the detour force gain has to be decreased. However, the H value will be significantly reduced. For example, if the repulsive force gain is doubled to 48 (*i.e.* double of the value in Table 6.1), the robot will complete the navigation with $d_h > 0.61$ m. Note that as discussed in subsection 6.4.2, $d_h > 0.6$ m is required to prevent a collision with a human. However, the H values for the four configurations will be reduced about 8% on average. On the other hand, the 8% change of H values imply that H is much more sensitive to the VFF gains

with Masoud's VFF than with the new VFF. This means, when designing a navigation system, more effort on gain tuning will definitely be required with his algorithm.

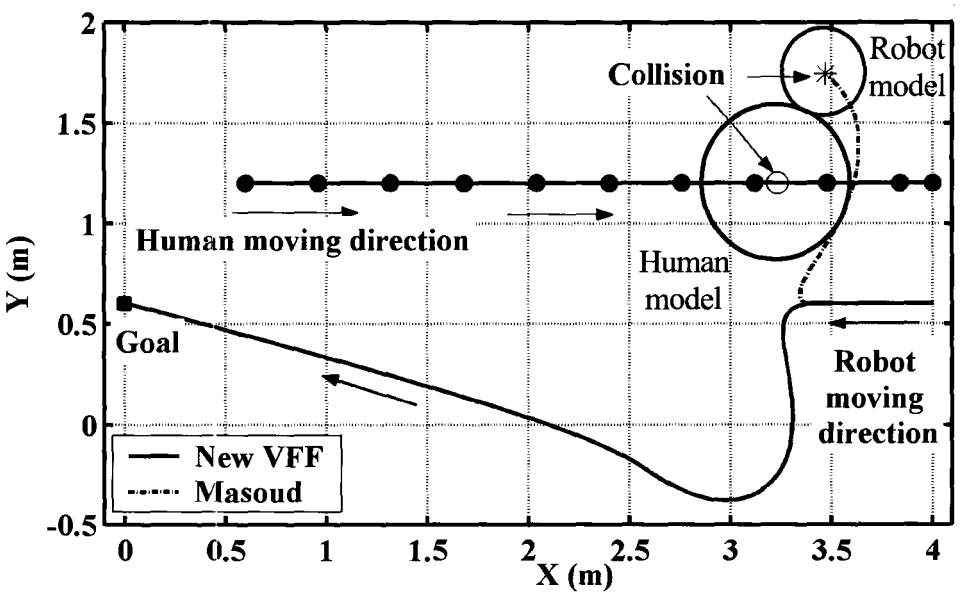


Fig. 6.19. Comparison of the simulation results for the new VFF-based algorithm and Masoud's VFF-based algorithm for avoiding a moving human parallel to the robot moving direction.

6.5 Conclusions

In this chapter, we utilized four fundamental obstacle configurations to check the navigation performance of the new VFF-based algorithm with a holonomic robot in simulations and experiments. Since in most manufacturing and service applications, VFF gain optimization is too complex to be performed, in the simulations and experiments, a set of non-optimal VFF gains (*i.e.* manually tuned) were used. The performance for the new VFF with non-optimal gains, the new VFF with optimal gains (optimized for Configurations 1, 2 and 3) and two important conventional VFF-based algorithms with optimal gains are compared in simulations.

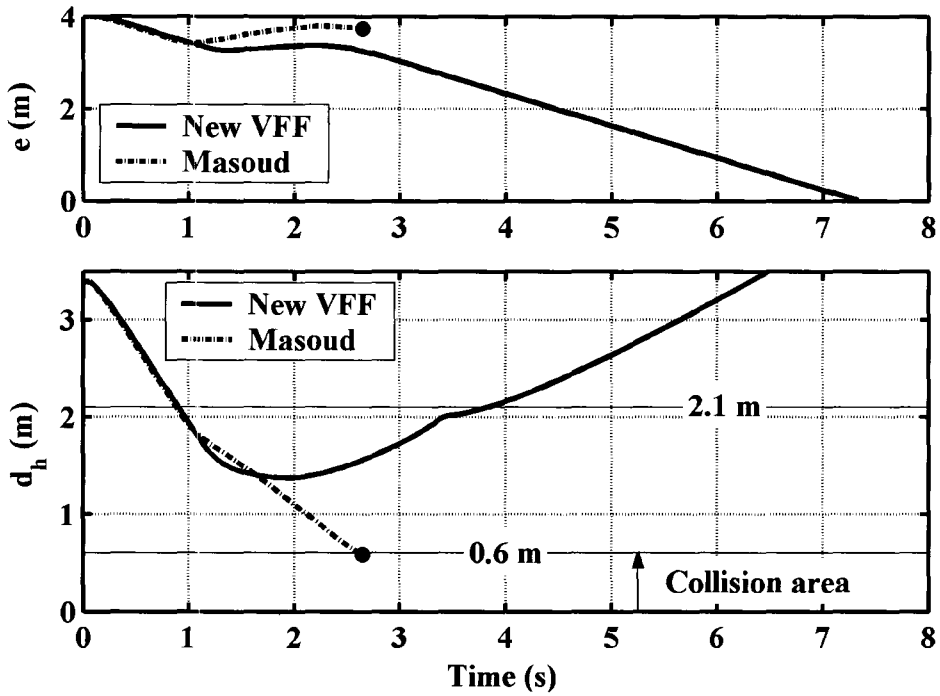


Fig. 6.20. Comparison of the simulation results for the new VFF-based algorithm and Masoud's VFF-based algorithm: distance e and d_h for avoiding a moving human parallel to the robot moving direction.

Summarizing the simulation results for the four configurations, the non-optimal gains performed the navigation with slightly reduced performance criteria in comparison with these using the optimal gains. This shows that the new algorithm does not required a large gain-tuning effort to work well. With simulations for four typical obstacle configurations, we can see a robot operated with the new VFF-based algorithm will have a better performance (21% larger H value on average) than the algorithm of Ge and Cui (2002). This is because the detour force drags the robot towards the goal while avoiding the obstacles, and the continuous force field reduces the path oscillation. The algorithm of Masoud (2007) produces the 4% better performance than the new VFF does. However,

due to the fixed detour force direction, it may cause a collision when avoiding a walking human.

In this chapter, we also presented an experiment setup built for the purpose of verifying the simulation results. It includes a holonomic robot; a vision system to measure the positions of the human and robot; a PC to perform image process; and another PC to generate the VFF, and control the robot to conduct the navigation. With this setup, experiments for the four configurations were performed and compared with the simulation results. During the experiments, the robot successfully avoids the obstacle(s) and reaches its goal. The experiment data and the simulation results are in good agreement. The error of the positions of the human and robot from the vision system and the wheel slide of the robot reduced the navigation performance and caused the discrepancy between the experiment data and the simulation results. The navigation performance with the new VFF and a holonomic robot has been demonstrated. In the next chapter, the navigation performance with the new VFF and a nonholonomic robot will be presented.

Chapter 7

Navigation simulations and experiments with a nonholonomic robot

7.1 Introduction

Nonholonomic robots are less expensive and more popular than holonomic robots, but they are harder to control. In this chapter, navigation simulations and experiments with a nonholonomic robot are presented. We will use a control system suitable for using any Cartesian VFF with a nonholonomic robot. Next, we will derive the active and critical regions of the three obstacles (*i.e.* the same obstacles as Chapter 6) for nonholonomic robots. This is followed by a brief description of the nonholonomic robot built for performing the navigation experiments. The simulation and experiment results with this robot are then compared with simulations of the holonomic robot used in Chapter 6. Four obstacle configurations are used in this chapter for the comparisons. In addition, a simulation with multiple moving obstacles will be presented. The chapter ends with a conclusions section.

7.2 Nonholonomic robot control system

The control system for using any Cartesian VFF with the nonholonomic robot has three levels, as shown in Fig. 7.1. The combination of the first and second levels is similar to the *nonholonomic motion planner* presented in Laumond *et al.* 1994. The first level is the path planner. In the planner, the virtual force, $\mathbf{F}_v$, is generated from the

current measured position(s) of the dynamic obstacle(s), the position(s) of the static obstacle(s) and the measured robot position, $\mathbf{P}_r$, using the equations presented in Section 5.4. The vision system used to measure the positions was described in Section 6.4. Note that Fig. 7.1 only shows the case with one human as the dynamic obstacle whose measured position is $\mathbf{P}_h$. $\mathbf{F}_v$ cannot be applied directly to the nonholonomic robot dynamics since the sideways force component would be ignored. To include this force component, we apply $\mathbf{F}_v$ to the dynamic equations of a reference holonomic robot. The holonomic dynamics is defined in (5.2.1) and rewritten as:

$$\mathbf{a}_{mr} = \mathbf{F}_v / m_{mr} \tag{7.2.1}$$

Next, with the acceleration vector $\mathbf{a}_{mr}$, known, the reference path $\mathbf{P}_{r,d}$ of the reference robot is derived by numerical integration.

The robot controller computes the linear and angular velocities, v and ω to track $\mathbf{P}_{r,d}$ under the nonholonomic constraint. In Laumond *et al.* (1994), a complex optimal

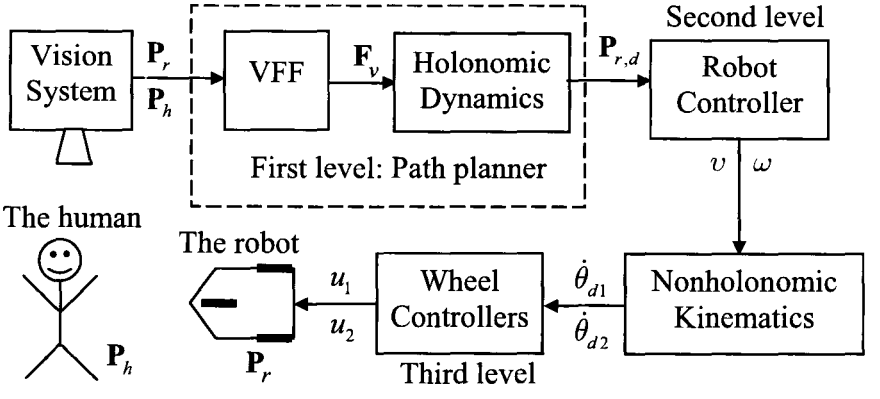


Fig. 7.1. The nonholonomic robot control system.

controller was used to minimize e. In this thesis, the simple nonlinear controller from Klančar and Škrjanc (2007) is used:

$$\begin{bmatrix} v \\ \omega \end{bmatrix} = \begin{bmatrix} v_d \cos\phi_e \\ \omega_d \end{bmatrix} + \begin{bmatrix} K_x & 0 & 0 \\ 0 & K_y & K_\phi \end{bmatrix} \begin{bmatrix} x_e \\ y_e \\ \phi_e \end{bmatrix} \tag{7.2.2}$$

where $\begin{bmatrix} x_e & y_e \end{bmatrix}^{\mathrm{T}} = \mathbf{P}_{r,d} - \mathbf{P}_r$; $\dot{\mathbf{P}}_{r,d} = \begin{bmatrix} \dot{P}_{rx,d} & \dot{P}_{ry,d} \end{bmatrix}^{\mathrm{T}}$; $\dot{\mathbf{P}}_r = \begin{bmatrix} \dot{P}_{rx} & \dot{P}_{ry} \end{bmatrix}^{\mathrm{T}}$; $\phi_e = \phi_d - \phi$ where $\phi_d = \mathrm{atan2}\left(\dot{P}_{ry,d}, \dot{P}_{rx.d}\right)$ is the heading angle of the reference robot; $\phi = \mathrm{atan2}\left(\dot{P}_{ry}, \dot{P}_{rx}\right)$ is the heading angle of the nonholonomic robot; $v_d = \sqrt{\dot{P}_{rx,d}^2 + \dot{P}_{ry,d}^2}$ and $\omega_d = \phi_e / T_s$. K_x, K_y and K_ϕ are time-varying positive control gains. Following the work of the authors, they are derived with:

$$\begin{aligned} \omega_{nh} &= \sqrt{\omega_d^2 + k_{nh} v_d^2}\,, \\ K_x &= K_\phi = 2\xi_{nh}\omega_{nh} \quad \text{and} \\ K_y &= k_{nh} v_d \end{aligned} \tag{7.2.3}$$

where $\xi_{nh} \in (0,1)$ is the desired damping coefficient of the controller, and k_{nh} is a positive gain. After manual tuning, we used $\xi_{nh} = 0.95$ and $k_{nh} = 10$ in our implementation. Note that due to the nonholonomic constraint, the robot may not be able to reach the desired position at the next sampling instant.

Next, v and ω are transformed to the desired angular, velocities ($\dot{\theta}_{d1}$ and $\dot{\theta}_{d2}$) of the driven wheels of the robot using the nonholonomic robot's kinematic equations (described in Section E.1 of Appendix E). The desired angular accelerations and angular positions are obtained by numerical differentiation and integration, respectively. The wheel controllers generate the control signals (u_1 and u_2) used to power the DC motors

driving the wheels. A FF+PD controller is employed for each wheel. The design of the controller can be found in Section E.3 of Appendix E.

7.3 Experimental setup and procedure

7.3.1 Experimental setup

Other than the robot, the experimental setup in this chapter is as same as the setup used in Chapter 6 (Fig. 6.2). A nonholonomic robot was built to conduct the experiments, as shown in Fig. 7.1. Its mass is 3.7 kg. Two conventional wheels are operated by two DC brushed motors (Maxon model RE40-148877) through a 10:1 ratio worm gear set. The two DOFs of the robot, moving forward/backward and turning are provided by rotating these two wheels. An omnidirectional wheel (*i.e.* Wheel 3) is installed to provide support. Two colour patches are attached on the robot. The first one is located on the centroid of the robot model and is shown as a '+' in Fig. 7.1. Its position defines the robot position $\mathbf{P}_r$. The other patch is 0.1 m away from the first patch. In experiments, we will use its position $\mathbf{P}_{cp} = \left[P_{cpx},\ P_{cpy} \right]^{\mathrm{T}}$ and $\mathbf{P}_r$ to estimate the heading angle of the robot as follows:

$$\phi = \operatorname{atan2}\left(P_{cpy} - P_{ry},\ P_{cpx} - P_{rx} \right) \tag{7.3.1}$$

7.3.2 Experimental procedure

The experiments and simulations will be performed for four obstacle configurations. Since the sizes of the active regions for the three obstacles are different from that for the holonomic robot, the four configurations are slightly different from the

configurations used in Chapter 6. The simulation and experiment results of the four configurations will be compared with simulations of a holonomic robot. Note that in the simulations the nonholonomic robot is assumed to be an ideal robot with no wheel slip. In addition, we will present a simulation of navigation in the presence of a walking human, a second mobile robot and a stationary circular obstacle. The same VFF gains used in the experiments in Chapter 6 are employed in this chapter: $K_1 = 3$ N/m , $K_3 = 20$ N/m , $K_5 = 17$ N/m and $b = 0.25$.

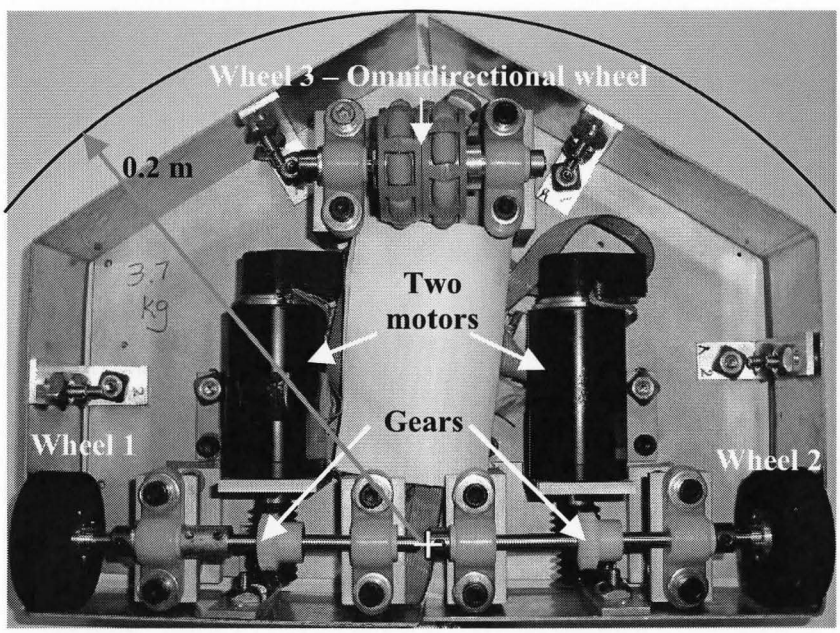


Fig. 7.2. The nonholonomic robot used in the experiments.

7.4 Sizes of the active and critical regions of the three obstacles for the nonholonomic robot

In this chapter, the three obstacles, a walking human, a stationary circular obstacle and a rectangular obstacle, that were used in Chapter 6 are also used. Since the velocity and acceleration limits for the nonholonomic robot are same as for the holonomic robot,

we will use the same sizes of critical regions for the first two obstacles. For the walking human, $r_3 = 0.8$ m ; and for the stationary circular obstacle, $r_{3,cir} = 0.4$ m . For the stationary rectangular obstacle, with (6.3.10), $\hat{r}_{3,rect} > \bar{\delta}_{r1} = 0.067$ m , we selected $\hat{r}_{3,rect} = 0.1$ m for the nonholonomic robot. An $\hat{r}_{3,rect}$ value closer to $\bar{\delta}_{r1}$ should make the avoidance more efficient and T_{arrive} smaller but requires little or no sliding of the robot wheels to avoid a collision. Since the traction of the wheels of the nonholonomic robot is much better than with the holonomic robot, a smaller $\hat{r}_{3,rect}$ is used in this chapter. We will denote it as $\hat{r}_{3,rect,nh}$ for the nonholonomic robot. Similar to (6.3.11), the size of the critical region of the rectangular obstacle is

$$r_{3,rect,nh} = \hat{r}_{3,rect,nh} + \rho_r = 0.1 + 0.2 = 0.3 \text{ m} . \tag{7.4.1}$$

In Chapter 6, when we computed the sizes of the active regions for those obstacles, we assumed the robot can be directly accelerated sideways (Y-direction). This is true if the robot is holonomic. For a nonholonomic robot, it must be turned $\pi/2$ first, then it can be accelerated in the Y-direction. Therefore, the turning time t_{turn} also needs to be considered. By assuming the robot turns with its maximum angular acceleration $\bar{\dot{\omega}}$, we have:

$$\frac{\pi}{2} = \frac{1}{2}\bar{\dot{\omega}} t_{turn}^2 \tag{7.4.2a}$$

$$\therefore \quad t_{turn} = \sqrt{2\left(\tfrac{\pi}{2}\right)/\bar{\dot{\omega}}} = \sqrt{\pi/\bar{\dot{\omega}}} . \tag{7.4.2b}$$

$\bar{\dot{\omega}} \approx 70$ rad/s^2 for the nonholonomic robot we used, so we have: $t_{turn} = \sqrt{\pi/70} = 0.22$ s . Note, when calculating the sizes of critical regions, t_{turn} is not considered since the robot

will be linearly decelerated to a stop.

When deriving the sizes of active regions for the three obstacles, t_{turn} must be considered. So for the human, (5.3.7) is changed to:

$$\begin{aligned} r_{2,nh} > \bar{\delta}_{hx2} + \bar{\delta}_{rx2} &= \bar{V}_{mr} \times (\Delta t_2 + t_{turn} + T_s) + \bar{V}_h \times (\Delta t_2 + t_{turn} + T_s) \\ &= 0.7 \times (1.18 + 0.22 + 0.06) + 1.0 \times (1.18 + 0.22 + 0.06) = 2.47 \text{ m}. \end{aligned} \tag{7.4.3}$$

In our implementation, we selected $r_{2,nh} = 2.5$ m. For the circular obstacle with a 0.1 m radius, (5.3.8b) is changed to:

$$\begin{aligned} r_{2,cir,nh} > \bar{\delta}_{rx2,circular,nh} &= \bar{V}_{mr} \times (\Delta t_{2,circular} + t_{turn} + T_s) \\ &= 0.7 \times (0.61 + 0.22 + 0.06) = 0.62 \text{ m}; \end{aligned} \tag{7.4.4}$$

In our implementation, we selected $r_{2,cir,nh} = 0.65$ m. For the rectangular obstacle, if the robot is assumed to come into the active region of the obstacle from the middle of it shortest side (*i.e.* $B = \frac{1}{2} B_{wid} = 0.3$ m). From (5.3.14):

$$\begin{aligned} \Delta t_{2,rect} &\geq \frac{r_{3,rect} + B - \frac{1}{2}\bar{a}_{mr} t_a^2}{\bar{V}_{mr}} + t_a \\ &\geq \frac{0.3 + 0.3 - \frac{1}{2} \times 10 \times 0.07^2}{0.7} + 0.07 = 0.92 \text{ s}. \end{aligned} \tag{7.4.5}$$

We selected $t_{2,rect} = 0.93$ s. Then by considering t_{turn}, (5.3.12) is rewritten as:

$$r_{2,rect,nh} > \bar{V}_{mr} \times (t_{2,rect,nh} + t_{turn} + T_s) = 0.7 \times (0.93 + 0.22 + 0.06) = 0.85 \text{ m} \tag{7.4.6}$$

Due to the space limitations of our laboratory, in our implementation, we used $r_{2,rect,nh} = 0.55$ m. This is compatible with Configurations 2 and 4 where $B = 0$.

7.5 Simulation and experiment results and analysis

7.5.1 Configuration 1: Avoiding a stationary circular obstacle in the collinear condition

The basic test of a VFF algorithm is to avoid a stationary obstacle. In Fig. 7.3 the simulation and experiment results with the nonholonomic robot are compared with simulations of the holonomic robot used in Chapter 6. The robot starts moving from position coordinates [3.99, 0.60] m to reach its goal at [0, 0.60] m; the obstacle is modelled as a disk with 0.1 m radius. Its centre is located at [2.16, 0.60] m. Hence, the avoidance condition is the collinear condition. The active and critical regions of the obstacle are labeled as $\mathbb{C}_{2,cir,nh}$ and $\mathbb{C}_{3,cir}$, respectively. The size of $\mathbb{C}_{2,cir,nh}$ for the nonholonomic robot is also used in the simulation with the holonomic robot $r_{2,cir} = r_{2,cir,nh} = 0.65$ m. We can see that both robots avoid the stationary obstacle. Both robots change moving direction at [2.6, 0.6] m. The experiment and simulation paths of the nonholonomic robot are in close agreement. From Fig. 7.4, we can see that the path with holonomic robot takes less time to reach its goal (T_{arrive} = 6.8 s in the simulation) than that with the nonholonomic robot (T_{arrive} = 8.3 s in the experiment and simulation). This is mainly because the nonholonomic robot must decelerate, turn and then move sideways to avoid the obstacle. Those actions require 1.6 s (from 2 s to 3.6 s). From Fig. 7.4, $d_{circular}$ is always greater than $r_{3,cir,nh}$ therefore the robot never enters the critical region of the obstacle. The performance measures are listed in Table 7.1. From this table, the holonomic robot wastes less energy and uses less time to reach the goal. In Fig. 7.4, the simulated values of L_{energy} vs. time for the two robots are presented. Comparing the

two curves, we can see that the larger L_{energy} for the nonholonomic robot is mainly because the turning of the nonholonomic robot consumes over 7 J during the interval from 2.6 s to 3.2 s. In simulations, the total performance criterion, H, with the nonholonomic robot is 26% less than that with the holonomic robot. In the experiment, the H value was reduced by 11% from the simulation with the nonholonomic robot. This is because L_{energy} was larger and $d_{min,cir}$ was smaller in the experiments.

Table 7.1. The performance criteria for Configuration 1

Methods	H	L_{energy} (J)	$d_{cir,min}$ (m)	T_{arrive} (s)
Nonholonomic (Sim.)	0.46	5.9	0.43	8.2
Nonholonomic (Exp.)	0.41	11.7	0.42	8.3
Holonomic (Sim.) + $r_{2,cir}$ = 0.65 m	0.62	4.3	0.44	6.8

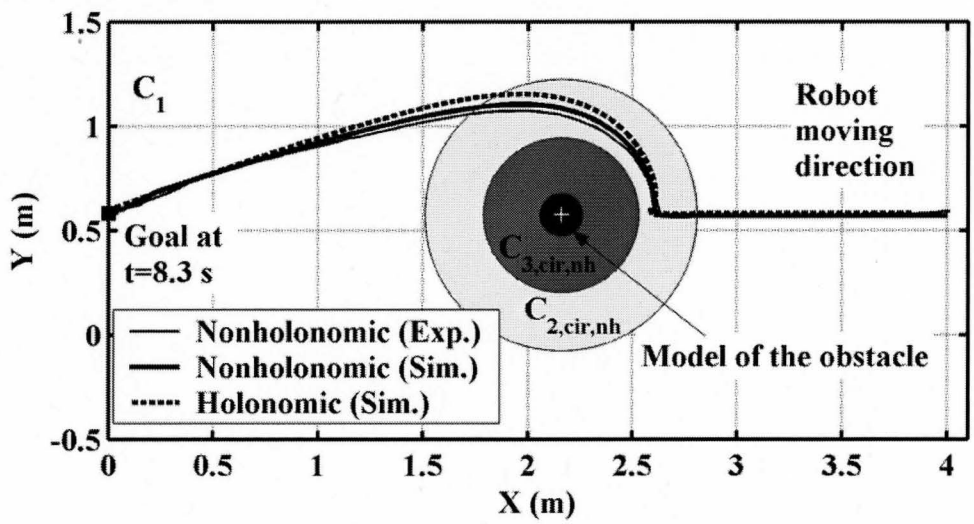


Fig. 7.3. Comparison of the simulation and experiment results with the nonholonomic robot and the simulation with the holonomic robot for Configuration 1.

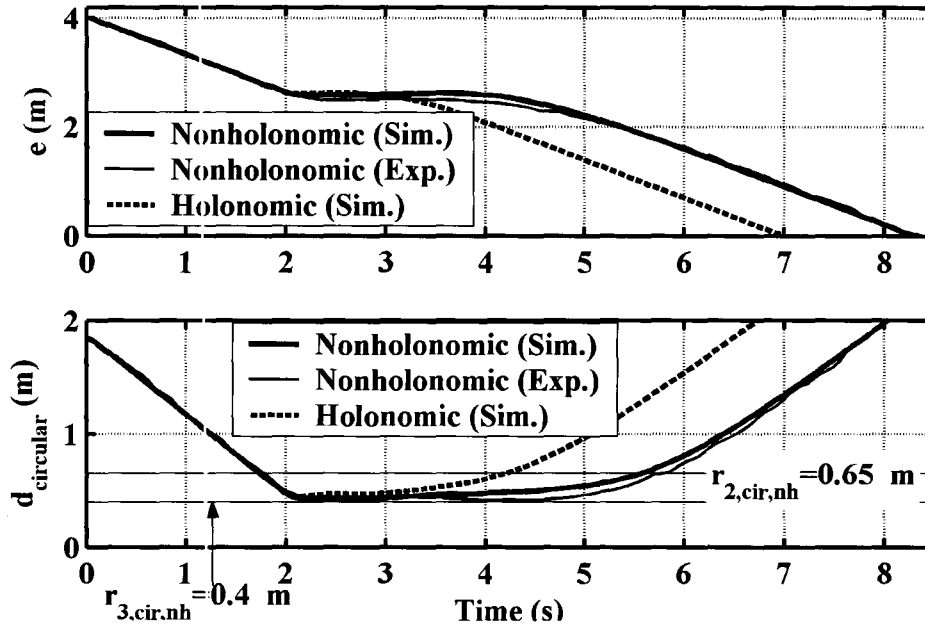


Fig. 7.4. Comparison of the simulation and experiment results for the nonholonomic robot and the simulation for the holonomic robot for distances e and $d_{circular}$ for Configuration 1.

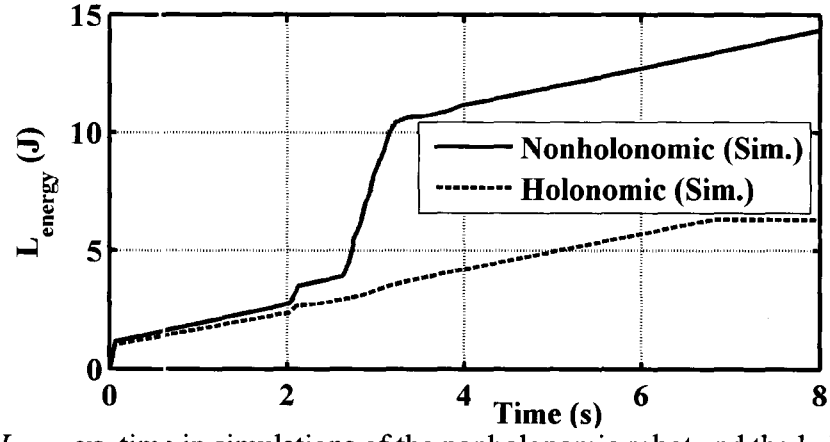


Fig.7.5. L_{energy} vs. time in simulations of the nonholonomic robot and the holonomic robot for Configuration 1.

7.5.2 Configuration 2: Avoiding a moving human in the collinear condition

Configuration 2 is more challenging for a VFF-based algorithm. In this configuration, shown in Fig. 7.6, the human starts from position [0.2, 0.6] m, and moves from left to right and then stops at [4.0, 0.0] m. At the same time, the robot starts from [3.99, 0.60] m and moves from right to left towards its goal at [0.0, 0.60] m. From Fig. 7.6 and 7.7, we can see that the nonholonomic robot avoided the walking human and the robot did not enter the critical region of the human. In comparison with the simulation with the holonomic robot, we can see that the nonholonomic robot moves backwards (*i.e.* the positive X-direction) more than the holonomic robot. When the robot needs to move sideways (at the position of [3.1, 0.6] m and t = 1.4 s), it has to take time to turn first. During the turning, the human gets closer and the repulsive force increases to push the robot to move backwards. Fig. 7.7 shows that V_h varies by ±0.4 m/s, and has an average of around 1.0 m/s while walking (*i.e.* before 5.2 s). The distance, d_h, is over 0.83 m in the simulation (over 0.91 m in the experiment), and is larger than r_3 (0.8 m for humans). The robot is always outside of the critical region of the human. T_{arrive} = 10.8 s for the nonholonomic robot in comparison with T_{arrive} = 8.3 s for the holonomic robot. Note that the human positions from the experimental data are used in the simulation. The discrepancy between the simulation and experimental results is mainly from the positioning error of the vision system for the robot's positions. The vision system error also influences the calculation of the virtual forces and causes a larger difference between the simulation and experimental results. The performance criteria are listed in Table 7.2. We can see that the nonholonomic robot wastes more L_{energy} and costs a longer T_{arrive}; and

also has a smaller $d_{h,min}$ than the holonomic robot. So, the H value with the nonholonomic robot is 48% less than the value with the holonomic robot. The H value in the experiment was slightly worse (4%) than in the simulation with the nonholonomic robot; and L_{energy} was also larger. This was due to a larger $d_{h,min}$.

Table 7.2. The performance criteria for Configuration 2

Methods	H	L_{energy} (J)	$d_{h,min}$ (m)	T_{arrive} (s)
Nonholonomic (Sim.)	0.26	14.5	0.85	10.8
Nonholonomic (Exp.)	0.25	33.8	0.91	10.9
Holonomic (Sim.) + $r_{2,cir}$ = 2.5 m	0.50	6.8	1.14	8.3

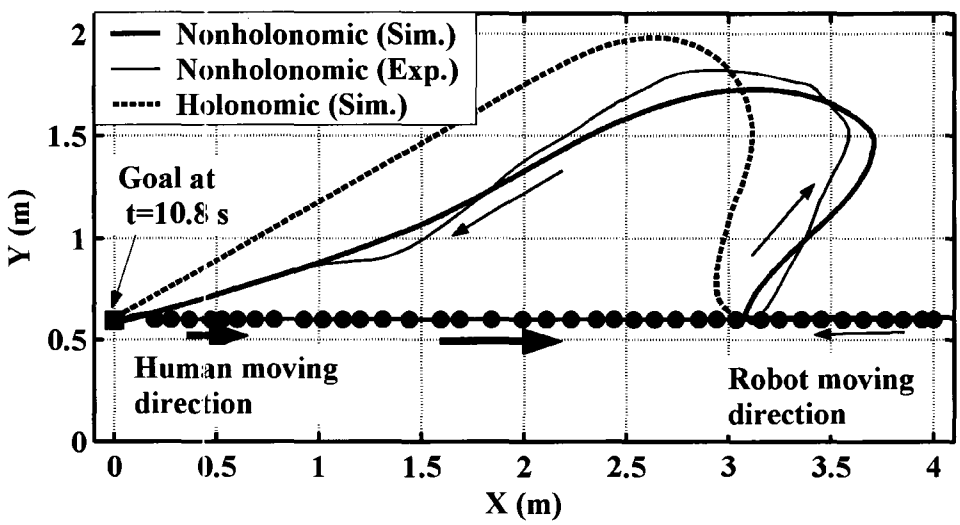


Fig. 7.6. Comparison of the simulation and experiment results with the nonholonomic robot and the simulation with the holonomic robot for Configuration 2.

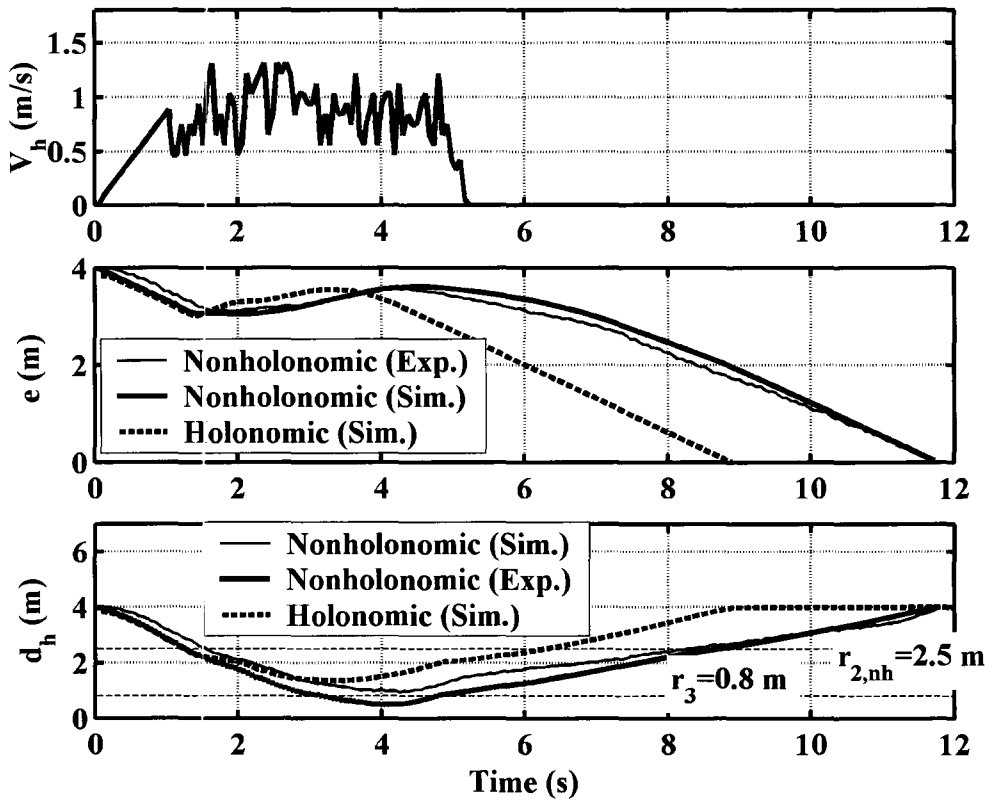


Fig. 7.7. Comparison of the simulation and experiment results for the nonholonomic robot and the simulation for the holonomic robot for distances e and d_h for Configuration 2.

7.5.3 Configuration 3: Avoiding Two Stationary Obstacles through a narrow passage

Another test of our VFF algorithm is the avoidance of two stationary obstacles in Configuration 3, as shown in Fig. 7.8. The rectangular obstacle's four corners are located at [-0.1,1.6], [0.5,1.6], [0.5,3.0] and [-0.1,3.0] m. $\mathbb{C}_{2,rect,nh}$ and $\mathbb{C}_{3,rect}$ are shown in the figure. The circular obstacle is located at [2.16, 1.5] m. Note that the active regions of the two obstacles intersect. At the beginning, the robot moves straight towards its goal. Upon entering the active region of the rectangular obstacle, the robot avoids this obstacle and its path deflects towards the upper-left due to the repulsive and detour forces from the

rectangular obstacle until the circular obstacle is encountered. The repulsive force and detour force from this obstacle drives the robot towards the lower-left. This causes the path to bend downwards. Next, the robot passes between the two obstacles (*i.e.* through the intersection of the two active regions). The path is slightly oscillatory. The reason for the oscillation was described in subsection 6.4.3. From Figs. 7.8 and 7.9, the nonholonomic robot avoids the two obstacles in the simulation and experiment, passes through the passage between the two obstacles and reaches its goal at 11.2 s. The difference between the simulation and experiment results are also mainly from the positioning error of the vision system. Comparing to the simulation with the nonholonomic robot, the holonomic robot also reaches the goal with a much shorter T_{arrive} and a smaller L_{energy}. The performance criteria are listed in Table 7.3. The H value with the nonholonomic robot was 24% less than the holonomic robot. Due to a larger L_{energy}, the H in the experiment was 16% less than in the simulation with the nonholonomic robot.

Table 7.3. The performance criteria for Configuration 3

Methods	H	L_{energy} (J)	$d_{min,rect}$ (m)	$d_{min,cir}$ (m)	T_{arrive} (s)
Nonholonomic (Sim.)	0.51	4.3	0.35	0.56	11.2
Nonholonomic (Exp.)	0.43	12.5	0.35	0.60	11.3
Holonomic (Sim.)	0.67	3.8	0.32	0.56	6.3

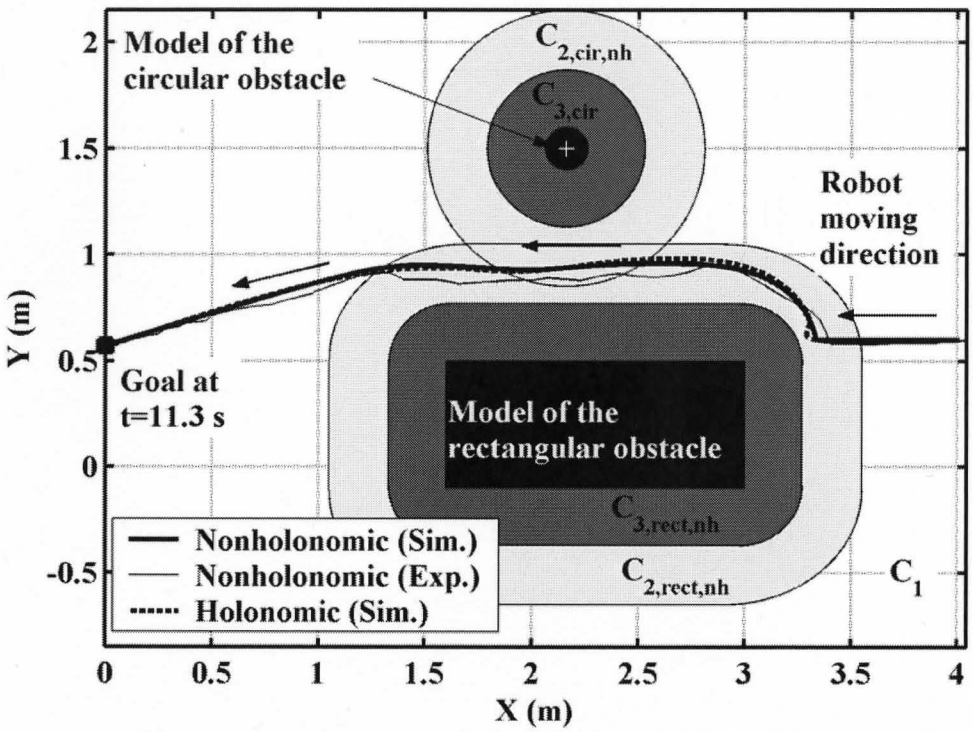


Fig. 7.8. Comparison of the simulation and experiment results with the nonholonomic robot and the simulation with the holonomic robot for Configuration 3.

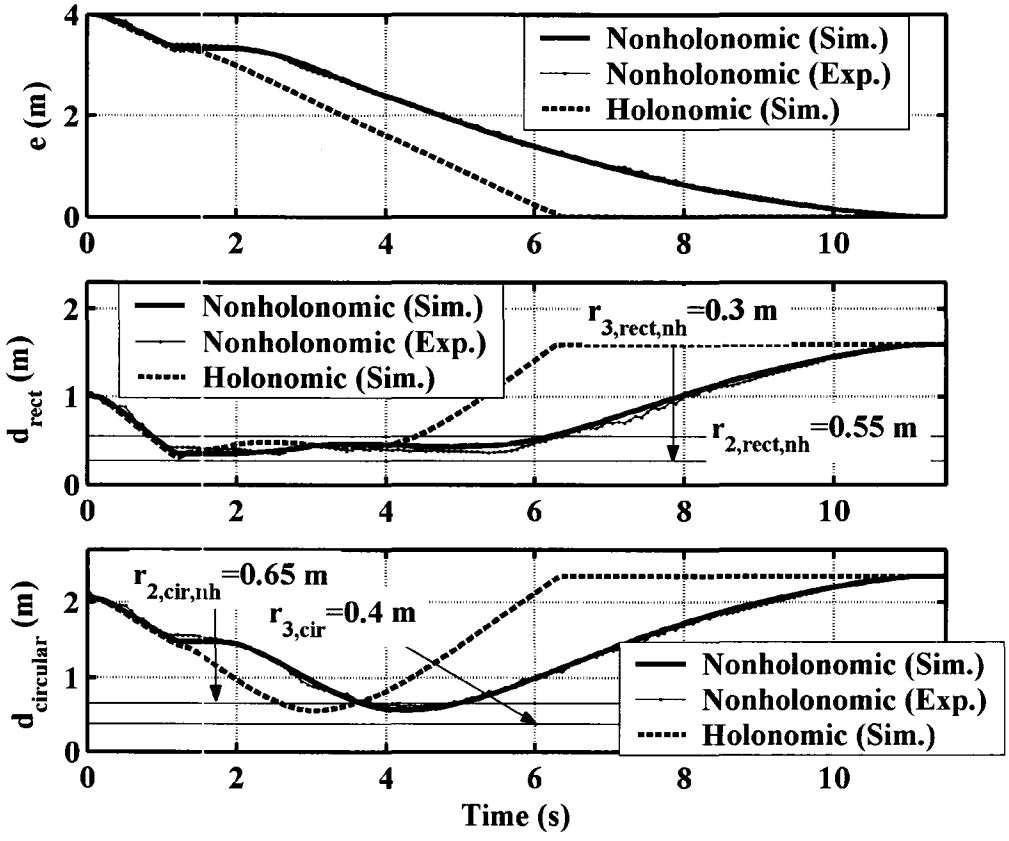


Fig. 7.9. Comparison of the simulation and experiment results for the nonholonomic robot and the simulation for the holonomic robot for distances e, d_{rect}, and $d_{circular}$ for Configuration 2.

7.5.4 Configuration 4: Avoiding a moving human and two stationary obstacles

Another challenging test of our VFF algorithm involves avoiding two stationary obstacles and a moving human. The configuration of the two stationary obstacles are as same as the two obstacle used in Section 7.4.3. The human starts from position [0.2, 0.60] m, moves from left to right, and then stops at [3.9, 0.60] m. Results of the simulation and experiment with the nonholonomic robot and the simulation with the

holonomic robot are shown in Figs. 7.10 and 7.11. We can see that the robot successfully avoided the three obstacles. The nonholonomic robot reaches the goal at 11.7 s in the experiment and 11.6 s in the simulation. The time when the robot is actively avoiding each obstacle can be found by comparing its d value with its r_2 value. The nonholonomic robot first avoids the rectangular obstacle (from 0.6 to 2.3 s), then the human gets close so the robot moves sideways to avoid him (from 1.1 to 7.6 s). After that, the circular obstacle is encountered and avoided (from 6.9 to 8.2 s). In Fig. 7.11, $d_{circular}$ is significantly larger than the $d_{circular}$ data in Fig. 7.4. It is because in Fig. 7.4, the avoidance is the worst case, the collinear condition, so the robot comes closer to (but still does not enter) the circular obstacle's critical region. Comparing to the simulation with the holonomic robot, the nonholonomic robot has a larger T_{arrive} and a larger L_{energy}. The performance criteria are listed in Table 7.4. The H value with the nonholonomic robot was 46% less than the value with the nonholonomic robot due to a smaller $d_{h,min}$, a larger L_{energy} and a longer T_{arrive} in the simulation. Due to a larger L_{energy}, the H in the experiment was 7% less than in the simulation with the nonholonomic robot.

Table 7.4. The performance criteria for Configuration 4

Methods	H	L_{energy} (J)	$d_{h,min}$ (m)	$d_{rect,min}$ (m)	$d_{cir,min}$ (m)	T_{arrive} (s)
Nonholonomic (Sim.)	0.31	21.0	0.98	0.35	0.56	11.6
Nonholonomic (Exp.)	0.29	57.3	1.01	0.35	0.58	11.7
Holonomic (Sim.)	0.57	6.0	1.31	0.35	0.59	8.2

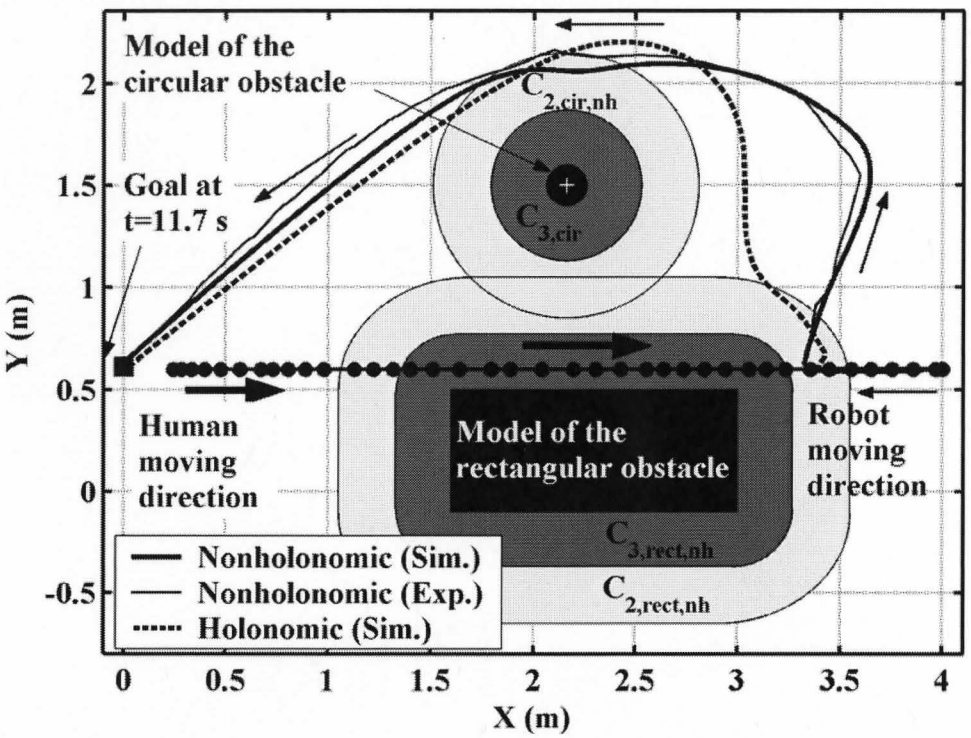


Fig. 7.10. Comparison of the simulation and experiment results with the nonholonomic robot and the simulation with the holonomic robot for Configuration 4.

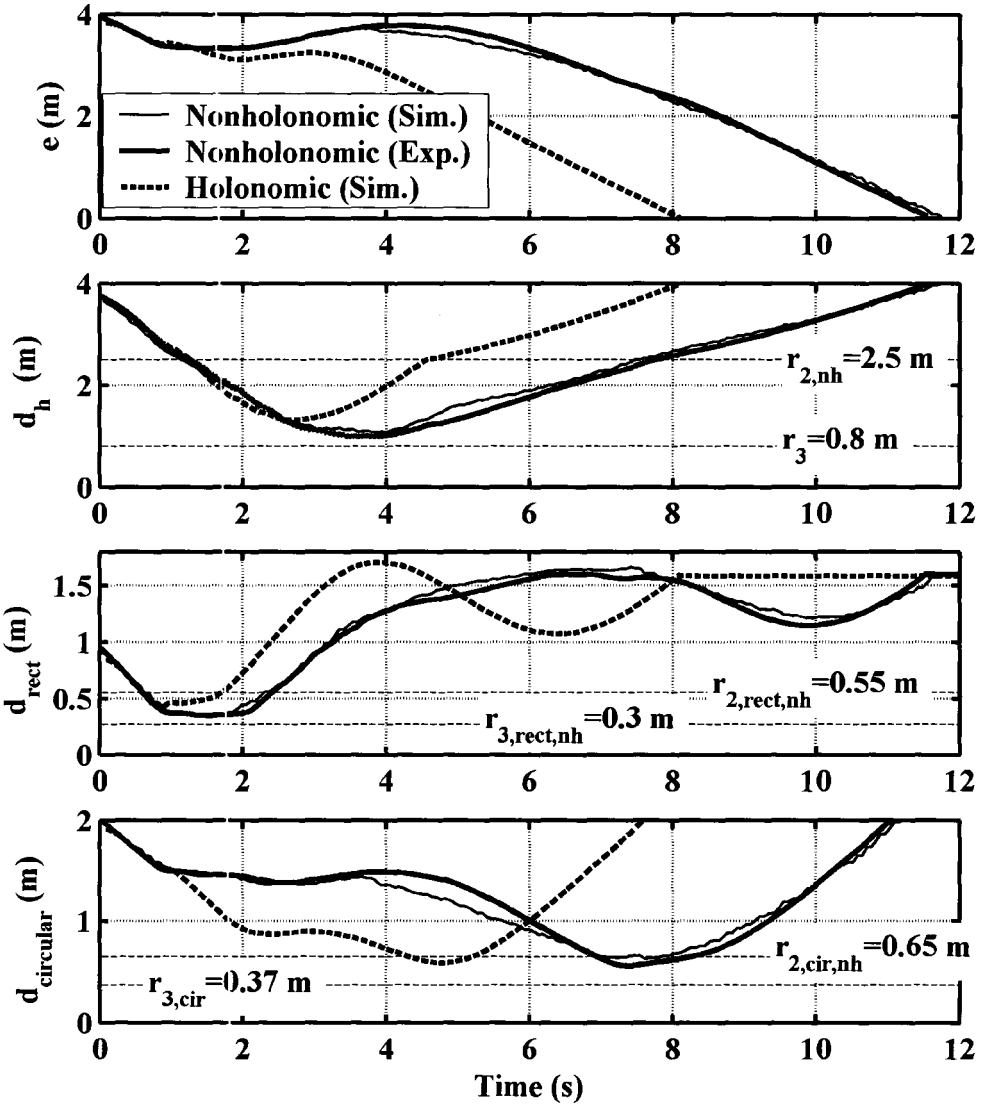


Fig. 7.11. Comparison of the simulation and experiment results for the nonholonomic robot and the simulation for the holonomic robot for distances e, d_{rect}, $d_{circular}$ and d_h for Configuration 2.

7.5.5 Simulation results for the robot avoiding a walking human, a mobile robot and a stationary circular obstacle

In this simulation, the robot avoids a stationary circular obstacle, a walking human and a mobile robot (termed the Obstacle Robot), as shown in Fig. 7.12. The

stationary circular obstacle (also used in Section 7.4.1) is located at [4.1, 2.0] m. The walking human starts from [0, 0.6] m and moves towards the right with a constant velocity vector [-1.0, 0] m/s. The Obstacle Robot is modeled as a disk with a 0.2 m radius. It starts from [3.0, 4.0] m and moves downwards with a constant velocity [0, -0.5] m/s (i.e. Its maximum velocity is $\overline{V}_{mob} = 0.5$ m/s). The size of its critical region can be computed by following the steps in Sections 5.3.2 and 5.3.3. Similar to (5.3.3), since $\overline{\delta}_{mob} = \overline{V}_{mob}\Delta t_1 = 0.5\times 0.13 = 0.065$ m, we have:

$$r_{3,mob} > \overline{\delta}_{mobile} + \overline{\delta}_{r1} + \rho_r + \rho_{mobile} = 0.065 + 0.067 + 0.2 + 0.2 = 0.54 \text{ m}. \quad (7.5.1)$$

We selected $r_{3,mob} = 0.55\ m$. Similar to (5.3.6b)

$$\Delta t_{2,mob} \geq \frac{r_{3,mob} - \frac{1}{2}\overline{a}_r t_a^{\,2}}{\overline{V}_r} + t_a = \frac{0.55 - \frac{1}{2}\times 10\times 0.07^2}{0.7} + 0.07 = 0.821 \text{ s} \quad (7.5.2)$$

We selected $\Delta t_{2,mob} = 0.83$ s. Similar to (5.3.7), we have:

$$\begin{aligned} r_{2,mob} &> \overline{\delta}_{rx2,mobile} + \overline{\delta}_{rx,2} \\ &= 0.5\times(0.83 + 0.23 + 0.06) + 0.7\times(0.83 + 0.23 + 0.06) = 1.34 \text{ m} \end{aligned} \quad (7.5.3)$$

We selected $r_{2,mob} = 1.35$ m. The active and critical regions of the three obstacles are presented in Fig. 7.12

The navigation path of the robot and the positions of the obstacles are presented over a time sequence in Fig. 7.13. The distances to the goal and the three obstacles are displayed in Fig. 7.14. The robot avoids the three obstacles. Note that d_{mobile} is the centre-to-centre distance between the Obstacle Robot and the nonholonomic robot. The nonholonomic robot moves directly towards its goal at beginning of the simulation ($t <$

0.9 s). At t = 0.9 s, the walking human is encountered. The nonholonomic robot moves sideways to avoid the human. Then at t = 3 s, the robot comes into the active region of the stationary obstacle and at t = 3.1 s the Obstacle Robot is also confronted. The nonholonomic robot avoids the three obstacles from 3.1 s to 5.3 s. During this time span, the nonholonomic robot is trapped among the three obstacles. So the distances to those three obstacles (see Fig. 7.13, t = 4 s and t = 5 s) are small but still larger than the corresponding r_3. At t = 5.3 s and 5.7 s, the nonholonomic robot moves out of the active regions of the stationary obstacle and the human, respectively. Next, the nonholonomic robot avoids the Obstacle Robot. Since the Obstacle Robot moves away from the nonholonomic robot, the repulsive force for this obstacle is small. The nonholonomic robot passes closely behind this obstacle, leaves its active region at t = 9.0 s, and continues towards its goal. The robot arrives its goal at t = 15 s. Due to laboratory space limitations, experiments for this configuration could not be performed.

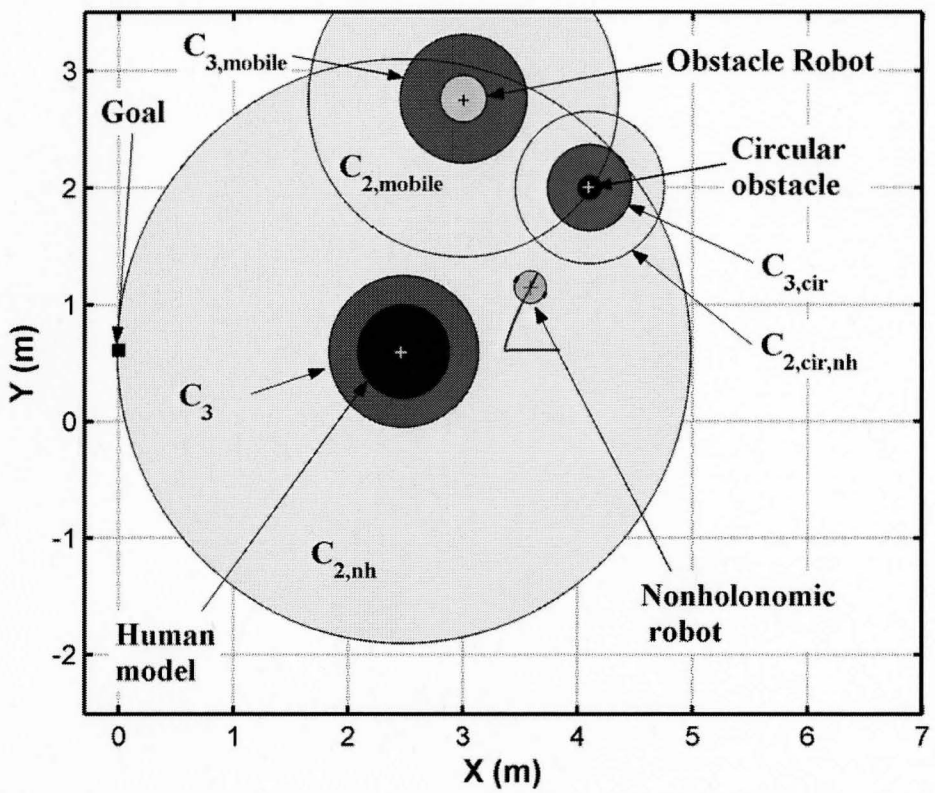


Fig. 7. 12. The configuration of the simulation with a walking human, two mobile robots and a stationary circular obstacle.

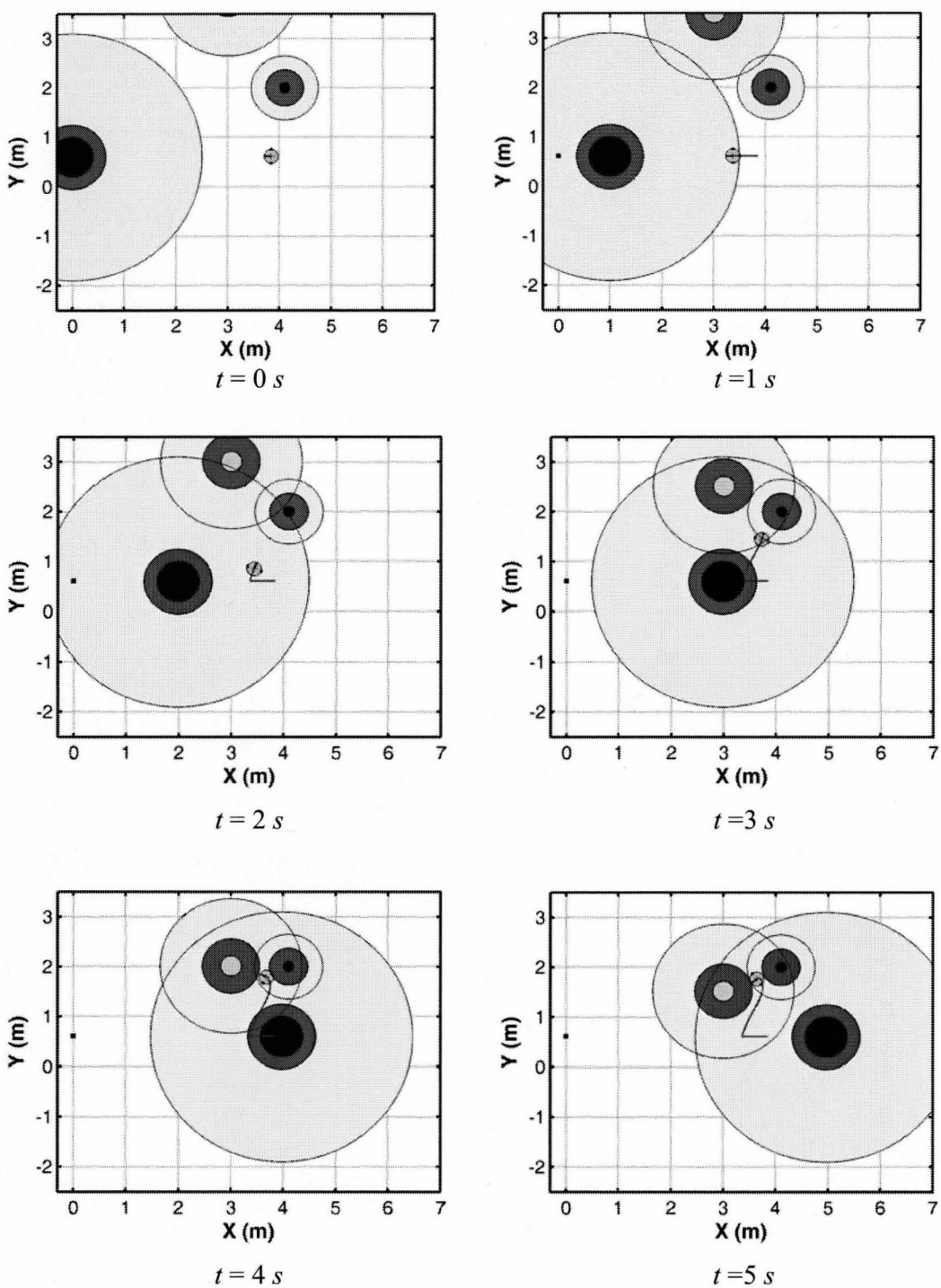
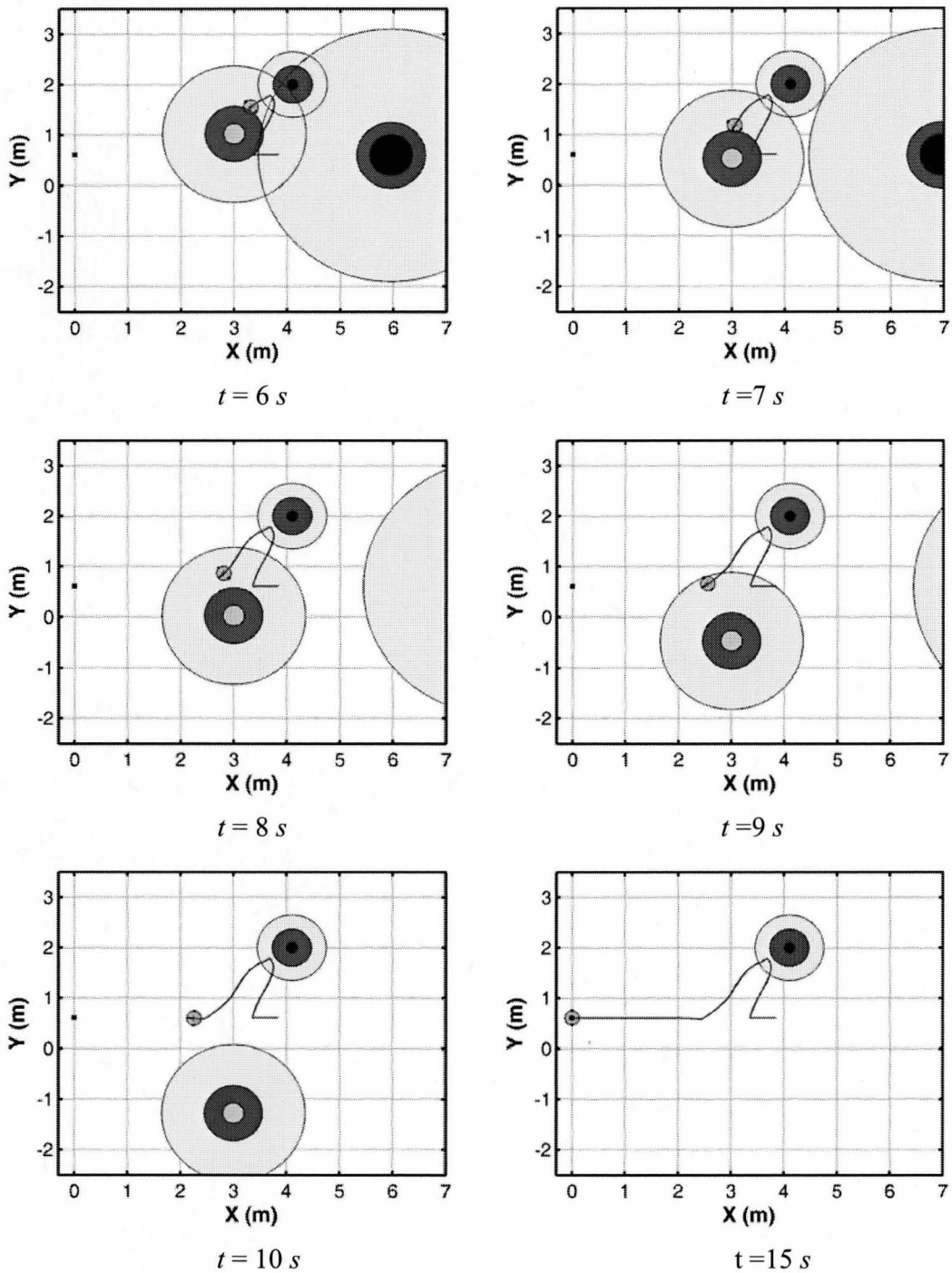


Fig. 7.13. The time sequence of the robot path and obstacle positions from the simulation with a walking human, two mobile robots and a stationary circular obstacle.



Continuation of Fig. 7.13. The time sequence of the robot path and obstacle positions from the simulation with a walking human, two mobile robots and a stationary circular obstacle.

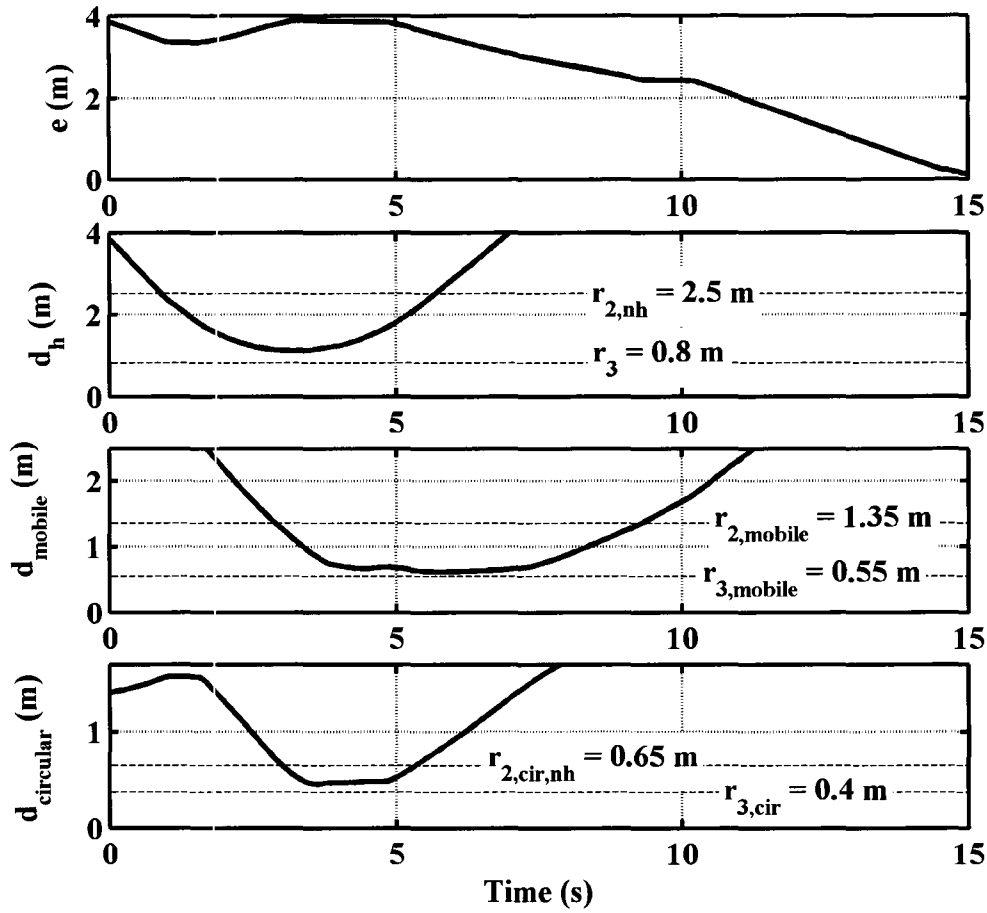


Fig. 7.14. Distances e, d_h and d_{mobile} from the simulation with a walking human, two mobile robots and a stationary circular obstacle.

7.6. Conclusions

At the beginning of this chapter, a control system suitable for using any Cartesian VFF with nonholonomic robots was used. With this system, the robot path for avoiding the obstacles is generated by assuming the VFF acts on a reference holonomic robot. A nonlinear controller is used to command the robot to track this path in consideration of the nonholonomic constraints.

Four obstacle configurations were then used to verify the control system and study the navigation performance with a nonholonomic robot and the new VFF. A nonholonomic robot was designed and built for this purpose. Simulation and experiment results for the four configurations were presented. The robot avoided the obstacle(s) and reached its goal in all of the configurations. The performance criteria for those configurations were compared to the simulations with the holonomic robot. We can conclude that the navigation with the nonholonomic robot has a lower performance (40% lower H on average) than that with the holonomic robot. The nonholonomic robot always requires a longer T_{arrive} and a larger L_{energy} to reach the goal. Another simulation for avoiding a walking human, a stationary circular obstacle and a mobile robot was also presented in this chapter. The nonholonomic robot avoids those obstacles and reaches the goal at t = 15 s. This simulation revealed the navigation ability of the new VFF for multiple moving obstacles.

Chapter 8

Conclusions and recommendations

8.1 Summary

In this thesis, we proposed an improved human-manipulator impact model in Chapter 3 that incorporates the manipulator dynamics, foam covering dynamics and the coupling between the human head and torso. This model was experimentally verified in Chapter 4 with two different manipulators, two foam coverings and different impact velocities. The maximum error between the predicted and experimental $a_{h,\max}$ results was less than 9%. The maximum error between the predicted and experimental $\varepsilon_{\max}$ results was less than 12%. Based on this impact model, the important design parameters that significantly influence the head acceleration, $a_{h,\max}$ and impact force f_c were investigated in Chapter 3. From this investigation, using foam coverings to soften the manipulator's surfaces (*i.e.* reducing K_c) was found to be the most effective way to reduce $a_{h,\max}$ and enhance the human safety. Reducing the effective mass of the manipulator (i.e. M_r) can effectively reduce $a_{h,\max}$ only when this mass is less than 10 kg; and the effective stiffness and damping of the manipulator (i.e. K_r and C_r) only have a minor effect on $a_{h,\max}$. A model-based foam covering design procedure was used to properly select the parameters of foam coverings in accordance with an impact-force-based and/or head-acceleration-based human-manipulator impact safety criterion and the foam thickness constraint was proposed in Chapter 3 and was validated experimentally in Chapter 4.

A novel VFF-based mobile robot navigation algorithm was proposed in Chapter 5. At first, an obstacle configuration with a motion unpredictable obstacle was simulated. We found that the algorithms of D*, RRT and Velocity Obstacle are not suitable for avoiding motion-unpredictable obstacles. Next, the new algorithm was compared with two important conventional VFF-based algorithms (by Ge and Cui (2002) and by Masoud (2007)) that are suitable for avoiding motion-unpredictable obstacles. With simulations for four typical obstacle configurations, we observed that a holonomic robot operated with the new VFF-based algorithm has a better performance (21% larger H value on average) than with the algorithm of Ge and Cui. This is because our detour force drags the robot towards the goal while avoiding the obstacles, and our continuous force field reduces the path oscillation. The algorithm of Masoud produces a 4% better H value on average than the new algorithm does. However, due to its fixed detour force direction in this VFF, his algorithm may cause a collision when avoiding a walking human. With the new algorithm, sensitivity of the optimal solution to changes in the VFF gains was proven to be small (<2%) in Appendix D. This low sensitivity helps to reduce the gain-tuning work when designing a navigation system. A new Lyapunov function for the piecewise VFF was created in Chapter 5. Stability of the new VFF was proven using this function and Lyapunov's second method. The methods to calculate sizes of active and critical regions for different obstacles were presented in Chapter 5. In Chapters 6 and 7, the sizes were calculated for four obstacles including two moving obstacles, a human and a mobile robot, and two stationary obstacles, a circular obstacle and a rectangular obstacle. The new VFF-based mobile robot navigation algorithm and the methods to

calculate the sizes of the active and critical regions were verified by simulations and experiments for four typical obstacle configurations with the holonomic robot in Chapters 6. Due to the existence of wheel sliding and positioning errors from the vision system, the experiment results are slightly different from the simulations. In Chapter 7, with a nonholonomic robot control system, navigation simulations and experiments were successfully performed with four obstacle configurations. A navigation simulation for multiple moving obstacles was also performed. From those simulations, we can conclude that the navigation with the nonholonomic robot has a worse performance (40% lower H values on average) than the holonomic robot. In particular, the nonholonomic robot always requires a longer time and wastes more energy to reach the goal.

8.2 Main research contributions

The main contributions of this research to the field of human-friendly robots are:

1. A novel human-manipulator impact model was proposed to incorporate the manipulator dynamics and the previously neglected coupling between the human head and torso. To establish this model, the method for approximating the configuration-dependent dynamics of robotics manipulators by the dynamics of a single DOF manipulator was introduced in this thesis. No methods for this purpose have been published in the existing literature. Furthermore, the dynamics of the manipulator, the foam covering and the human head-neck-torso were included and make the new impact model more realistic than existing impact models.
2. A new VFF-based mobile robot navigation algorithm was proposed. It

features improved functions for the repulsive and detour virtual forces, a new stabilizing virtual force, and a properly designed detour force direction. The new VFF is continuous which should diminish path oscillations. This algorithm is suitable for navigating in environments with motion-unpredictable obstacles, including humans. The mobile robot with the new VFF-based algorithm successfully completed navigations without causing a collision while the other five existing algorithms incurred collisions in one of the configurations.

3. A design procedure was introduced for properly selecting dynamic parameters of foam coverings in accordance with a safety criterion and a foam thickness constraint. This procedure provides a solution for designing foam coverings for robotic manipulators to enhance human safety. This design procedure is also applicable to both impact-force-based and head-acceleration-based safety criterions.
4. Methods to calculate sizes of active and critical regions for different obstacles were introduced. These methods consider the worst avoidance conditions and utilize the shapes, velocity limits and acceleration limits of the obstacles and robots. No methods for this purpose have been published in existing literature.
5. A new Lyapunov function for the new piecewise VFF was created. With this function, the stability of the new VFF for avoiding motion-unpredictable obstacles was solved. In the existing literature, the Lyapunov functions were used only for APT-based algorithms and only for analyzing the stability when

navigating among stationary and motion-predictable obstacles.

8.3. Recommendations for future work

The new VFF-based mobile robot navigation algorithm suffers from the well-known local minima problem, which makes the robot stop at a point and fail to reach its goal. A local minima scenario with the new VFF-based navigation algorithm is shown in Fig. 8.1. This scenario involves two identical stationary obstacles. The attractive, repulsive and detour forces from the goal and obstacles are balanced (i.e. $\mathbf{F}_a + \Sigma\mathbf{F}_\Lambda + \Sigma\mathbf{F}_\psi = \mathbf{0}$) at the local minima point. If the robot is stopped at this point, the stabilization force also equals zero. Then the robot will stay at this point indefinitely. A solution is needed for this local minima scenario.

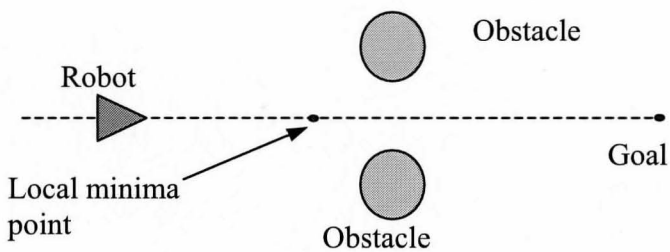


Fig. 8.1. A local minima scenario.

We calculated the sizes of the active and critical regions for a stationary rectangular obstacle. In service and manufacturing applications, some moving obstacles should also be modelled as rectangles, such as a moving forklift in a factory and a moving bed in a hospital. A method is required to calculate the active and critical regions for moving rectangular obstacles.

In Chapter 3, we modelled the human-manipulator impact. During human-robot

cooperation, mobile robots also could have unintentional physical contact with humans in the applications requiring them to work in close proximity of humans, such as rescue robots. An impact model for mobile robots should be investigated.

Also in Chapter 3, we assumed the stiffness of foam coverings is a constant. For some foam coverings, the relationship between the static applied force and the compressed depth is nonlinear. The stiffness cannot be approximated as a constant. To more accurately model the impact with those foam coverings, nonlinear foam stiffness models should be used, such as the fifth order polynomial model in the paper of Singh, Davies and Bajaj (2003).

In Appendix B, we analyzed the stability of the new VFF without considering the mobile robot's velocity and acceleration limits. With those limits the stability of the new VFF may not be preserved (for related work see: Chen and Sereenath 1992). Therefore, the stability analysis for the new VFF including the robot velocity and acceleration limits is required.

In Chapter 7, the goal of the nonholonomic robot is successfully reached in simulations and experiments. However, according to Brockett's theorem (Brockett 1983), the continuous feedback controller presented in (7.2.2) cannot guarantee that the nonholonomic robot will reach the goal. To solve this problem, a different control algorithm is required when the nonholonomic robot is near its goal and out of any active regions of obstacles, *e.g.* the algorithm from Sørdalen and De Wit (1993). This control algorithm can operate the nonholonomic robot to reach the goal with an exponential convergence to the goal.

References:

AAAM. 1980. The Abbreviated Injury Scale (AIS) 1980. *Barrington, IL, USA*, Association for the Advancement of Automotive Medicine

AAAM. 2008. Abbreviated Injury Scale (AIS) 2005 – Update 2008. *Barrington, IL, USA*, Association for the Advancement of Automotive Medicine

Athans, M., Sandell, N., and Varaiya, P. 1975. Stability of interconnected systems, In *Proceeding of IEEE Conference on Decision and Control including the 14th Symposium on Adaptive Processes*, Huston, TX, USA, **14**(1): 456 – 462.

Applied Robotics Inc. Collision sensors. [on-line], available: http://www.arobotics.com /products/collision/collision.aspx

Belhouche, F. 2009. Reactive path planning in a dynamic environment. *IEEE Transactions on Robotics*, **25**(4): 902-911.

Bicchi, A., and Tonietti, G. 2004. Fast and 'soft-arm' tactics. *IEEE Robotics & Automation Magazine*, **11**(2): 22 – 33.

Blake, A., and Zissermen, A. *Virtual reconstruction*, MIT press, Cambridge, Massachusetts, 1987,

Bone, G., and Elbestawi, M. 1991. Active end effector control of a low precision robot in deburring, *Robotics and Computer-Integrated Manufacturing*, **8**(2): 87 – 96.

Borenstein, J., and Koren, Y. 1989. Real-time obstacle avoidance for fast mobile robots. *IEEE Transaction on Systems, Man, and Cybernetics*, **19**(5): 1179–1186.

Brockett, R. 1983. Asymptotic stability and feedback stabilization. In *Proceeding of*

Differential Geometric Control Theory, Cambridge, MA, 1983, pp. 181–191.

Bruce, J., and Veloso, M. 2002. Real-time randomized path planning for robot navigation, In *proceeding of IEEE/RSJ International Conference on Intelligent Robots and Systems*, Lausanne, Switzerland, pp. 2383 – 2388.

Bruce, J., and Veloso, M. 2006. Safe multirobot navigation with dynamics constraints. *Proceeding of the IEEE*, **94**(7): 1398 – 1410.

Chen, C., and Sreenath, N. 1992. Stability of mixed structures with control saturation. In *Proceedings of the 1992 American Control Conference*, Chicago, IL, USA, **4**: 3099-100.

Cheng, X., Liu, Z., and Zhang, X. 2006. Trajectory optimization for ship collision avoidance system using genetic algorithm. In *Proceeding of IEEE OCEANS 2006 Asia-Pacific*, Singapore pp. 1 – 6.

Cleveland Motion Control, Brush servo motor 3500 series. [on-line], available: http://www.clevelandmotioncontrols.com/downloads/servo_motors/platform3500.pdf

De Luca, A., and Mattone, R. 2005. Sensorless Robot Collision Detection and Hybrid Force/Motion Control. In *Proceed of IEEE International Conference on Robotics and Automation 2005*, Barcelona, Spain pp. 999 – 1004.

De Luca, A., Albu-Schäffer, A, Haddadin, S., and Hirzinger, G. 2006. Collision detection and safe reaction with the DLR-III lightweight manipulator arm, *Proceeding of 2006 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 1623 – 1630.

De Santis, A., Siciliano, B., De Luca, A., and Bicchi, A. 2008. An atlas of physical

human-robot interaction, *Mechanism and Machine Theory*, **43**(3): 253 – 270.

Deng, M., Inoue, A., Sekiguchi, K., and Jiang, L. 2010. Two-wheeled mobile robot motion control in dynamic environments. *Robotics and computer-integrated manufacturing*, **26**(3): 268 – 272.

Divebiss, A.,and Wen, J. 1997. A path space approach to nonholonomic motion planning in the presence of obstacles. *IEEE Transaction on Robotics and Automation*, **13**(3): 443 – 451.

Djordjevic, M. 1983. Stability analysis of interconnected systems with possibly unstable subsystems, *System and Control Letters*, **3**: 165 – 169.

Dorf, R., and Bishop, R. *Modern control systems*. Prentice Hall, New Jersey, 1991

Earl, M., and D'Andrea, R. 2005. Iterative MILP methods for vehicle-control problems. *IEEE Transaction on Robotics*, **21**(6): 1158–1167.

Ellis, H. *Observers in control systems: a practical guide*. Academic press, San Diego, US, 2002.

EuroNCAP (2004). European Protocol New Assessment Programme—Frontal Impact Testing Protocol.

Fahimi, F., Nataraj, C., and Ashrafiuon, H. 2009. Real-time obstacle avoidance for multiple mobile robots. *Robotica*, **27**(2): 189-198.

Fraichard, T., and Asama, H. 2004. Inevitable Collision States: a step towards safer roots?, *Advanced Robotics*, **18**(10): 1001 – 1024.

Ferguson, D., and Stentz, A. 2006. Using interpolation to improve path planning the field D* algorithm, *Journal of Field Robot*, **23**(2): 79 – 101.

Fiorini, P., and Shiller, Z. 1998. Motion planning in dynamic environments using velocity obstacles, *The International Journal of Robotics Research*, **17**(7): 760 – 772.

Fox, D., Burgard, W., and Thrun, S. 1997. The dynamic window approach to collision avoidance. *IEEE Robotics and Automation Magazine*, **4**(1): 23 – 33.

Ge, S., and Cui, Y., 2002. Dynamic motion planning for mobile robots using potential field method. *Autonomous Robots*, **13**: 207–222.

Haddadin, S., Albu-Schäffer, A., and Hirzinger, G. 2007. Safety evaluation of physical human-robot interaction via crash-testing. Robotics: Science and Systems Conference, Atlanta, USA pp. 217 – 224.

Haddadin, S., Laue, T., Frese, U., Wolf, S., Albu-Schäffer, A., and Hirzinger, G. 2008. Kick it like a Safe Robot: Requirements for 2050. *Robotics and Autonomous Systems*: *Special Issue on Humanoid Soccer Robots*, **57**(8): 761 – 775.

Haddadin, S., Albu-Schäffer, A., and Hirzinger, G. 2008. The role of the robot mass and velocity in physical human-robot interaction – Part I: Non-constrained Blunt Impacts. In *Proceeding of IEEE International Conference on Robotics and Automation*, Pasadena, California, USA pp. 1331 – 1338.

Haddadin, S., Albu-Schäffer, A., and Hirzinger, G. 2009. Requirements for safe robot: measurements, analysis and new insights. *The International Journal of Robotics Research*, **28**(11-12): 1507 – 1527.

Haddadin, S., Albu-Schäffer, A., Frommberger, M., Rossmann, J., and Hirzinger, G. 2009. The "DLR Crash Report": Towards a standard crash-testing protocal for robot safety – part I: Results, In *Proceeding of IEEE International Conference on*

Robotics and Automation, Kobe, Japan, pp. 272 – 279.

Haddadin, S., Albu-Schäffer, A., Eiberger, O., and Hirzinger, G. 2010. New Insights Concerning Intrinsic Joint Elasticity for Safety, In *proceeding of IEEE/RSJ International Conference on Intelligent Robots and Systems*, Taipei, Taiwan, pp. ????.

Hart, P., Nilsson, N., and Rafaeal, B. 1968. A formal basis for the heuristic determination of minimum cost paths, *IEEE Transactions on Systems Science and Cyernetics*, **4**: 100 – 107.

Heinzmann, J., and Zelinsky, A. 2003. Quantitative safety guarantees for physical human-robot interaction. *International Journal of Robotics Research*, **22**(7 – 8): 479 – 504.

Holmberg, R., and Khatib, O. 2000. Development and control of a holonomic mobile robot for mobile manipulation tasks. *International Journal of Robotics Research*, **19**(11): 1066 – 1074.

Hsu, D., Kindel, R., Latombe, J., and Rock, S. 2002. Randomized kinodynamic motion planning with moving obstacles. *The International Journal of Robotics Research*, **21**(3): 233–255.

Hu, H., Brady, M., and Probert, P. 1991. Navigation and control of a mobile robot among moving obstacles. In *Proceeding of 30th IEEE Conference On Decision and Control 1991*, Brighton, England **1**: 698-703.

Ikuta, K., and Nokata, M. 2003. Safety evaluation method of design and control for human-care robots. *International Journal Robotics Research*, **22**(5): 281 – 297.

ISO, Robots for industrial environments – Safety requirements – Part I: Robot, ISO Standard 10218-1, 2006.

Jacobs, S., Ferrein, A., Schiffer, S., Beck, D., and Lakemeyer, G. 2010. Robust collision avoidance in unknown domestic environments, *RoboCup 2009: Robot Soccer World Cup XIII.* Springer, Berlin, 2010, pp. 116 – 127.

Jaillet, L., Cortés, J., and Siméon, T. 2010. Sampling-based path planning on configuration-space costmaps. *IEEE Transactions on Robotics*, **26:** 1-12.

Jeong, S. and Takahashi, T. 2009. Impact force reduction of manipulators using a dynamic acceleration polytope and flexible collision detection sensor. *Advanced Robotics*, **23**(3): 367 – 383.

Jiang, B., and Gainer, C. 1987. A cause-and-effect analysis of robot accidents. *Journal of Occupational Accidents*, **9**(1): 27 – 45.

Johansson, M. *Piecewise Linear Control Systems*, Springer, Berlin, 2002.

Jung, S., Jang, E., and Hsia, T. 2005. Collision Avoidance of a Mobile Robot Using Intelligent Hybrid Force Control Technique, In *Proceedings of IEEE International Conference on Robotics and Automation* 2005, Barcelona, Spain, **15**(4): 4418-4423.

Kanayama, Y., Kimura, Y.; Miyazaki, F.; Noguchi, T. 1990. A stable tracking control method for a non-holonomic mobile robot, In *Proceedings of IEEE International Workshop on Intelligent Robots and Systems 1991*, Osaka, Japan, **3**: 1236 – 1241.

Khatib, O. 1985. Real-time obstacle avoidance for manipulators and mobile robots. In *Proceeding of IEEE International Conference on Robotics and Automation 1985*,

2: 500–505.

Khatib, O., Yokoi, K., Brock, O., Chang, K., and Casal, A. 1999. Robots in human environments: basic autonomous capabilities. *The International Journal of Robotics Research*, **18**(7): 684 – 696.

Klančar, G., and Škrjanc, I. 2007. Tracking-error model-based predictive control for mobile robots in real time. *Robotics and Autonomous Systems*, **55**(6): 460–469.

Ko, N., and Lee, B. 1996. Avoidability measure in moving obstacle avoidance problem and its use for robot motion planning. In *Proceeding of. IEEE/RSJ International Conference on Intelligent Robots and Systems 1996*, Osaka, Japan **3**: 1296–1303.

Koenig, S., and Likhachev, M. 2002. Improve fast replanning for robot navigation in unknown terrain, In *Proceeding. of IEEE International Conference on Robotics and Automation 2002*, Washington, DC, USA, pp. 968 – 975.

Koren, Y., and Borenstein, J. 1991. Potential field methods and their inherent limitations for mobile robot navigation. In *Proceeding. of IEEE International Conference on Robotics and Automation 1991*, Sacramento, California, USA **2**: 1398–1404.

Kosuge, K., Yoshida, H., and Fukuda, T. 1993. Dynamic control for robot-human collaboration. *Proceedings of IEEE/RSJ International workshop on Robot and Human Communication 1993*, Tokyo, Japan pp. 398 – 401.

Laffranchi, M., Tsagarakis, N., and Caldwell, D. 2009. Safe human robot interaction via energy regulation control. *Proceedings of IEEE/RSJ International Conference on Intelligent Robotics and Systems 2009*, St. Louis, Missouri, USA pp. 35 – 40.

Large, F., Laugier, C., and Shiller, Z. 2005. Navigation among moving obstacles using

the NLVO: principles and applications to intelligent vehicles, *Autonomous Robot*, **19**(2): 159 – 171.

Laumond, J. Jacobs, P., Taïx, M., and Murray, M. 1994. A motion planner for nonholonomic robots, *IEEE Transactions on Robotics and Automation*, **10**(5): 577–593.

Lavalle, S. 1998. Rapidly-exploring random trees: a new tool for path planning, In *Technical Report No. 98 – 11.*

Lavalle, S., James, J., and Kuffner, J. 2001. Randomized kinodynamic planning, *International Journal of Robotics Reseaech*, **20**(5): 378 – 400.

Law, S. 1993. Thickness and resistivity variations over the upper surface of the human skull. *Brain Topography*, **6**: 99 – 109.

Lee, B., and Lee, C. 1987. Collision-free motion planning of two robots. *IEEE Transaction on Systems, Man, and Cybernetics*, **17**(1): 21–31.

Lim, H., and Tanie, K. 2000. Human safety mechanisms of human-friendly robots: passive viscoelastic trunk and passive movable base. *International Journal of Robotics Research*, **19**(4): 307 – 334.

Liu, Y., and Arimoto, S. 1992. Path planning using a tangent graph for mobile robots among polygonal and curved obstacles. *The International Journal of Robotics Research*, **11**(4): 376–382.

Loizou, S., and Kyriakopoulos, K. 2008. Navigation of multiple kinematically constrained robots. *IEEE Transactions on Robotics*, **24**(1): 221-231.

Lund, H. 2004. Modern artificial intelligence for human-robot interaction. In *Proceeding*

of IEEE, **92**(11): 1821-1838.

Lyapunov, A. 1892. *The general problem of the stability of motion*. Translated by Fuller, A., *International Journal of Control*, **55**(3): 531-773, Mar. 1992.

Marija, S., and Ivan, P. 2007. Dynamic window based approach to mobile root motion control in the presence of moving obstacles. In *Proceeding of IEEE International Conference on Robotics and Automation 2007*, Roma, Italy, pp. 1986–1991.

Martin, O., and Marshi, A. 1992. Step length and frequency effects on ground reaction forces during walking. *Journal of Biomechanics*, **25**(10): 1237–1239.

Martinez-Gomez, L., and Fraichard, T. 2009 Collision avoidance in dynamic environments: an ICS-base solution and its comparative evaluation, In *Proceeding of IEEE International Conference on Robotics and Automation 2009*, Kobe, Japan, pp. 100 – 105.

Masoud, A. 2007. Decentralized self-organizing potential field-based control for individually motivated mobile agents in a clustered environment: a vector-harmonic potential field approach. *IEEE Transaction on* Systems, Man, and Cybernetics -Part A: Systems and Humans, **37**(3): 372–390.

Masoud, S., and Masoud, A. 2000. Constrained motion control using vector potential fields. *IEEE Transaction on* Systems, Man, and Cybernetics -Part A: Systems and Humans, **30**(3): 251–272.

Mills, N., and Gilchrust, A. 2006. Bicycle helmet design. In *Proceedings of the Institution of Mechanical Engineering, Part L: Journal of Materials: Design and Application*, **220**(4): 167 – 180.

Minguez, J., and Montano, L. 2009. Extending Collision avoidance methods to consider the vehicle shape, kinematics, and dynamics of a mobile robot. *IEEE Transaction on Robotics*, **25**(2): 367–381.

Nin, S. 2004. Theoretical and experimental study of pneumatic servo motion control systems. *PhD thesis*, McMaster University.

Oberer, S., and Schraft, R. 2007. Robot-dummy crash tests for robot safety assessment. In *Proceeding of IEEE International Conference on Robotics and Automation 2007*, Roma, Italy pp. 2934-2939.

Ostrovskaya, S., Angeles, J., and Spiteri, R. 2000. Dynamics of a mobile robot with three ball-wheels, *Inernational Journal of Robotics Research*, **19**(4): 383 – 394.

Park, J., Kim, B., Song, J., and Kim, H. 2007. Safe link mechanism based on passive compliance for safe human-robot collision. In *Proceeding IEEE International Conference on Robotics and Automation 2007*, Roma, Italy pp. 1152 – 1157.

Park, J., and Song, J. 2009. Collision Analysis and evaluation of collision safety for service robots working in human environments, In *Proceeding IEEE International Conference on Robotics and Automation 2009,* Munich, Germany, pp. 1 – 6.

Park, S., and Lee, B. 2006. Analysis of robot collision characteristics using concept of the collision map. *Robotica*, **24**(3): 295–303.

Pivtoraiko, M., Knepper, R., and Kelly, A. 2009. Differentially constrained mobile robot motion planning in State Lattices. *Journal of Field Robotics*, **26**(3): 308-333.

Ponticel, P. 2007. DCX research advances shop safety, paint processes. *Automotive engineering international*, **115**(2): 57 – 58.

Pratt, G., and Williamson, M. 1995. Series elastic actuator. In *Proceeding IEEE/RSJ International Conference of Intelligent Manipulators System 1995*, Pittsburgh, Pennsylvania, USA **1**: 399 – 406.

Qu, H., Yang, S., Willms, A., and Yi, Z. 2009. Real-time robot path planning based on a modified pulse coupled neural network model. *IEEE Transactions on Neural Networks.* **20**(11):1724-1739.

Ostrovskaya, S., Angeles, J., and Spiteri, R. 2000. Dynamics of a mobile robot with three ball-wheels. *International Journal of Robotics Research*, **19**(4): 383 – 394.

Ren, J., McIsaac, K., and Patel, R. 2008. Modified Newton's method applied to potential field-based navigation for nonholonomic robots in dynamic environments. *Robotica*, **26**(3): 285 – 294.

Rivin, E. *Passive vibration isolation.* New York: The American society of mechanical engineers, New York, 2003.

Robotic industries association, 1999. American national standard for industrial robots and robot systems - safety requirements. ANSI/RIA R15.06-1999.

Rosales, S., Scaglia, G., Mut, V., and di Sciascio, F. 2009. Trajectory tracking of mobile robots in dynamic environments-a linear algebra approach. *Robotica*, **27**(7): 981 – 997.

Rossetter, E., and Gerdes, J. 2006. Lyapunov based performance guarantees for the potential field lane-keeping assistance system. *Journal of Dynamic Systems, Measurement and Control*, **128**(3): 510–522.

Saravanan, R., Ramabalan, S., and Balamurugan, C. 2009. Evolutionary multi-criteria trajectory modeling of industrial robots in presence of obstacles. *Engineering*

applications of artificial intelligence, **22**(2): 329 – 342.

Shi, C., Wang, Y., and Yang, L. 2010. A local obstacle avoidance method for mobile robots in partially known environment. *Robotics and autonomous systems*, **58**(5): 425-434.

Singh, R., Davies, P., and Bajaj, A. 2003. Identification of nonlinear and viscoelastic properties of flexible polyurethane foam. *Nonlinear dynamics* **34**(3–4): 319 – 346.

Skiena, S. *The algorithm design manual*. Springer-Verlag, London, 2008.

Slotine, J., and Li, W. *Applied nonlinear control*. Prentice Hall, New Jersey, 1991.

Spangelo, I., and Egeland, O. 1994. Trajectory planning and collision avoidance for underwater vehicles using optimal control. *IEEE Journal of Oceanic Engineering*, **19**(4): 502 – 511.

Spong, M., and Vidyasagar, M. *Robot dynamics and control*, John Wiley & Sons, New York, 1989.

Stentz, A. 1994. Optimal and efficient path planning for partially-known environment. In *Proceeding of IEEE International Conference on Robotics and Automation 1994*, New Orleans, LA, USA, pp. 3310 – 3317.

Suita, K., Ikeda, H., Yamada, Y., Tsuchida, N., Sugimoto, N., and Imai, K. 1995. A failure-to-safety 'Kyozon' system with simple contact detection and stop capabilities for safe human-autonomous robot coexistence. In *Proceeding of IEEE International Conference on Robotics and Automation 1995*, Nagoya, Aichi, Japan **3**: 3089 - 3096.

Sultan, C., Seereeram, S., and Mehra, R. 2004. Minimization and equalization of energy

for formation flying reconfiguration. In *Proceeding of IEEE International Conference on Robotics and Automation 2004*, New Orleans, LA, USA, **4**: 4047 – 4452.

Sørdalen, O., and De Wit, C. 1992. Exponential control law for a mobile robot: Extension to path following. *IEEE Transactions on Robotics and Automation* **9**(6): 1791–1797.

Tanner, H., Loizou, S., and Kyriakopoulos, K. 2001. Nonholonomic stabilization with collision avoidance for mobile robots. I*n Proceeding of International Conference on Intelligent Robots and Systems 2001*, Maui, Hawaii, USA **2**: 1220 – 1224.

Tanner, H., Loizou, S., and Kyriakopoulos, K. 2003. Nonholonomic navigation and control of cooperating mobile manipulators. *IEEE Transaction on Robotics and Automation,* 1**9**(1): 53–64, 2003.

Tomovic, R., and Vukobratovic M. *General Sensitivity theory*. 35th edition, American Elsevier, New York, 1972.

Townsend, M., and McCammond, D. 1975. Response of human and anthropometric model skulls to impact loading. *Medical and Biological Engineering*, **13**(3): 405-413.

Tsai, R. 1987. A Versatile Camera Calibration Technique for High-Accuracy 3D Machine Vision Metrology Using Off-the-Shelf TV Cameras and Lenses. IEEE Journal of Robotics and Automation, **3**(4): pp. 323–344.

Tsubouchi, T., Kuramochi, S., and Arimoto, S. 1995. Iterated forecast and planning algorithm to steer and drive a mobile robot in the presence of multiple moving obstacles. In *Proceeding of IEEE International Conference on Intelligent Robots*

and Systems, Pittsburgh, Pennsylvania, USA **2**: 33–38.

Tsoularis, A., and Kambhampati, C. 1999. Avoiding moving obstacles by deviation from a mobile robot's nominal path. *The International J. of Robotics Research*, **18**(5): 454 – 465.

van der Schuur, M., van der Heide, E., Jan, F., and Gaymans, R. 2004. Elastic behavior of flexible polyether (urethane – urea) foam materials. *Polymer*, **45**(8): 2721 – 2727.

van Emmerik, R., and Wagenaar, R. 1996. Effects of walking velocity on relative phase dynamics in trunk in human walking. *Journal of Biomechanics*, **29**(9): 1175–1184.

Vannoy, J., and Xiao, J. 2008. Real-time adaptive motion planning (RAMP) of mobile manipulators in dynamic environments with unforeseen changes. *IEEE Transaction on Robotics*, **24**(5): 1199–1212.

Vanualailai, J., Sharma, B., and Nakagiri, S. 2008. An asymptotically stable collision-avoidance system. *International Journal of Non-Linear Mechanics*, **43**(9): 925 – 932.

Vermeulen, M., and Wisse, M. 2008. Maximum allowable manipulator mass based on cycle time, impact safety and pinching safety. *Industrial Robot*, **35**(5): 410 – 420.

Versace, J. 1971. A review of the severity index. In *Proceeding. of 15th Stapp Car Crash conference 1971*, SAE paper 710881, 771 – 796.

Viano, D. 2003. Seat influences on female neck responses in rear crashes. *Trafic Injury Prevention*, **4**(3): 228 – 239.

Watanabe, K. 1998. Control of an omnidirectional mobile robot. *International Conference on Knowledge-based Intelligent Electronic Systems*, Adelaide, South Australia pp. 51–60.

Weisberg, S. *Applied linear regression*, 3rd edition, John Wiley & Sons, New York, 2005.

Willinger, R., Bourdet, N., Fischer, R., and Le Gall, F. 2005. Model analysis of the human neck in vivo as a criterion for crash test dummy evaluation. *Journal of Sound and Vibration*, **287**(3): 405 – 431, 2005.

Yamada Y., Hirasawa, Y., Huang, S., Umetani, Y., and Suita, K. 1997. Human-robot contact in the safeguarding space. *IEEE/ASME Transaction on Mechatronics*, **2**(4): 230-236.

Yamada, T., Watanabe, K., Kiguchi, K., and Izumi, K. 2001. Dynamic model and control of a holonomic omnidirectional mobile robot, *Autonomous Robots*, **11**(2): 173 – 189.

Yamamoto, M., Shimada, M., and Mohri, A. 2001. Online navigation of mobile robot under the existence of dynamically moving multiple obstacles, In *Proceeding of 4th IEEE International Symposium on Assembly and Task Planning Soft Research Park 2001*, Fukuoka, Japan pp.13 – 18.

Yang, S., and Meng, M. 2001. Neural network approaches to dynamic collision-free trajectory generation. *IEEE Transactions on Systems, Man, and Cybernetics, Part B.* **31**(3): 302-318.

Zinn, M., Khatib, O., Roth, B., and Salisbury, J. 2004. A new actuation concept for human-friendly robot design: playing it safe. *IEEE Robotics & Automation Magazine*, **11**(2): 12 – 21.

Zollo, L., Siciliano, B., Laschi, C., Teti, G., and Dario, P. 2003. An experimental study on compliance control for a redundant personal robot arm. *Robotics and Autonomous Systems*, **44**(1): 101 – 129.

Appendix A

A.1 Measuring the stiffness and damping of foam coverings

The stiffness and damping of a foam covering must be measured for use in the impact model. An apparatus was built for this purpose, as shown in Fig. A.1. In this apparatus, a half-cylindrical aluminium part acts as the mechanical impactor and is fixed to a linear slide. This slide incorporates a linear ball bearing that allows it to slide on a vertical rail. A linear position transducer (T150 with 0.0001 m resolution from Novotechnik Inc.) is used to measure the displacement of the slide, y_c. The total moving mass, including the movable part of the transducer, the slide and the half cylindrical part, is $m_m = 1.54$ kg.

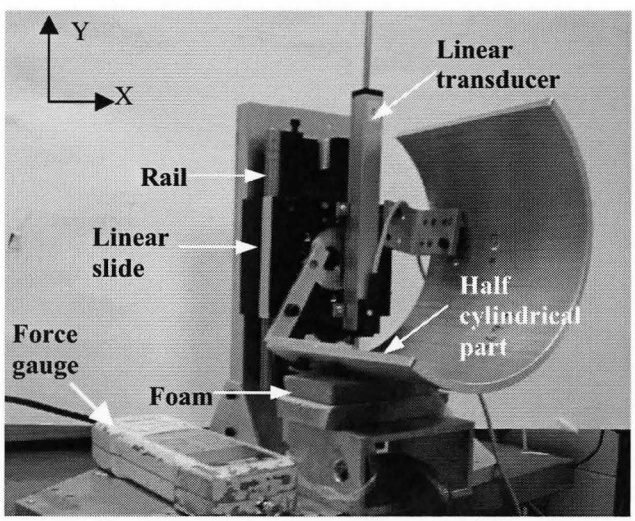


Fig. A.1. The apparatus for measuring the stiffness and damping of foam coverings.

The foam stiffness will be estimated first. The half-cylindrical part is pushed down with different static forces, f_{static}. The values of these forces is read using a digital force gauge (Chatillon DFE-10 from Ametek with 0.1 N resolution). The cylindrical

surface of the aluminium part compresses the foam. The slide's displacement measured by the linear transducer is converted into the compressed depth ε of the foam by $\varepsilon = y_0 - y_c$ where y_0 is the position where the cylindrical part first contacts the foam. The physical relationship between f_{static} and ε with this apparatus is:

$$f_{static} = K_c \varepsilon - m_m g \tag{A.1.1}$$

Letting $m_m g = K_c \varepsilon_0$ where ε_0 is the initial compressed depth of the foam caused by gravity, (A.1.1) becomes:

$$f_{static} = K_c (\varepsilon - \varepsilon_0) = K_c \varepsilon_c \tag{A.1.2}$$

The stiffness of the foam is estimated by fitting a line to f_{static} vs. ε_c using the linear regression method (see Chapter 2 of Weisberg 2005). Two foams were measured. The f_{static} vs. ε_c regression lines and the measured data from the two foams are shown in Figs. A.2a and A.3a, respectively. Comparing the two foams, we can see that the f_{static} vs. ε_c of Foam 1 is more linear than that for Foam 2. The measured stiffness of Foam 1 is 24.25 kN/m and the stiffness of Foam 2 is 13.48 kN/m.

To estimate the damping of the foam, the slide is lifted up and then released under gravity to impact the foam. After contact it will be bounced upwards by the foam and pulled downwards by gravity again. During this behaviour the data of the displacement of the slide y_c vs. time t is collected at 1 kHz. Those data are presented in Figs. A.2b and A.3b for the two foams, respectively. When the slide contacts the foam, its motion is dictated by the friction on the rail, the foam dynamics and the gravity. If it does not contact the foam, it will only be affected by the gravity and friction. Therefore, we need

to find the friction between the rail and slide first. For simplicity, the dynamic friction is assumed to equal the static friction. The friction is obtained by only fitting the y_c vs. t data from the beginning of downward motion; *i.e.* the data within $t = [0 \quad 0.1]$ s in Fig. A.2b and $t = [0 \quad 0.08]$ s in Fig. A.3b. Since the motion of the slide is only affected by the gravity and friction within those intervals, y_c is:

$$y_c = y_{fall} - \tfrac{1}{2}\left(g + f_{friction}\text{sign}(\dot{y}_c)/m_m\right)t^2 \tag{A.1.3}$$

where $f_{friction}$ is the friction force; y_{fall} is the initial height of the slide that is measured by the linear transducer. The *fmincon* function in Matlab is used to minimize the sum of squared errors between the measured data and the results from (A.1.3). The estimated friction is $f_{friction} = 2.0\ N$. The physical relationship between y_c and t when it contacts the foam is:

$$\left(m_m g + f_{friction}\text{sign}(\dot{y}_c)\right) = m_m \ddot{y}_c + 2\xi_c\sqrt{m_m K_c}\,\dot{y}_c + K_c y_c \tag{A.1.4}$$

In (A.1.4), we now have values for the mass, foam stiffness and friction. $\ddot{y}_c$ and $\dot{y}_c$ can be obtained by numerically differentiating the measured y_c. Only the foam damping is unknown. The *fmincon* function is used to curve fit (A.1.4) to the data of y_c vs. t to obtain the damping ratio ξ_c. Three tests were taken with different heights for the two foams and the mean value of the ξ_c estimates will be used as the estimated damping ratio of the corresponding foam. The estimated damping ratio of the first foam is 0.112; for the second foam, it is 0.086. Comparisons between the measured data and the simulation results with the estimated stiffness and damping are shown in Figs. A.2b and

A.3b. From the two figures, we can see that the simulation data and the measured data are in good agreement. Since the half-cylindrical part is also used in human-manipulator experiments, the stiffness and damping ratios of the two foams should be identical in the human-manipulator impact experiments. Since the head mass is 4.4 kg in those experiments, the values of the damping of Foam 1 and 2 can be calculated with $C_c \approx 2\xi_c\sqrt{M_h K_c}$. The damping coefficients of the two foams in the impact experiments are then:

$$C_c \approx 2\times 0.11\sqrt{4.4\times 24250} = 71.9 \text{ Ns/m for Foam 1; and}$$

$$C_c \approx 2\times 0.086\sqrt{4.4\times 13840} = 43.8 \text{ Ns/m for Foam 2.}$$

The estimated dynamic parameters of the two foams are listed in Table 3.1.

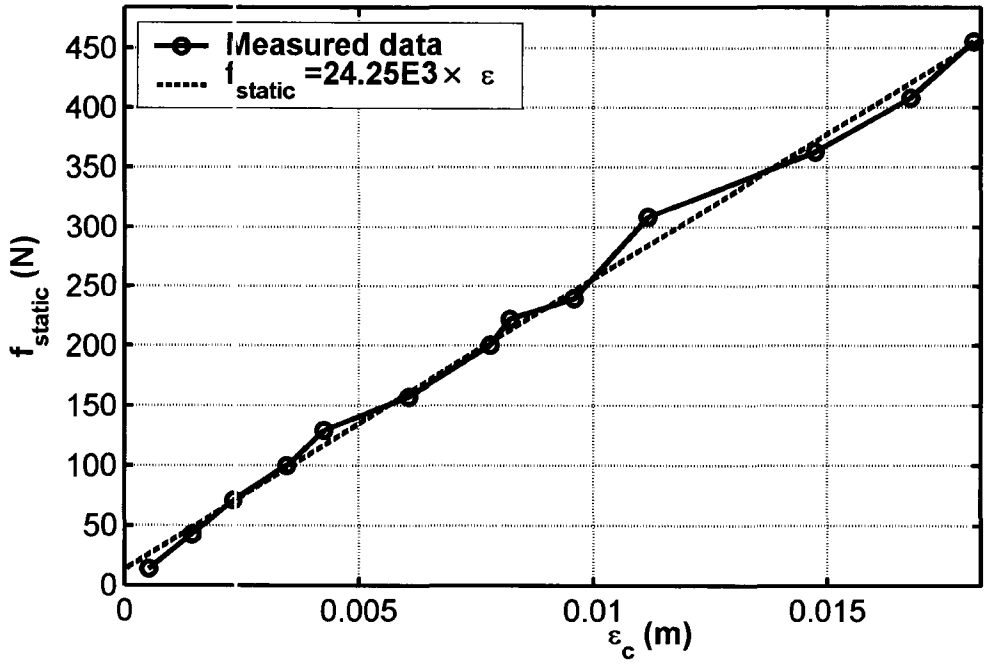


Fig. A.2a. f_{static} vs. ε_c for Foam 1.

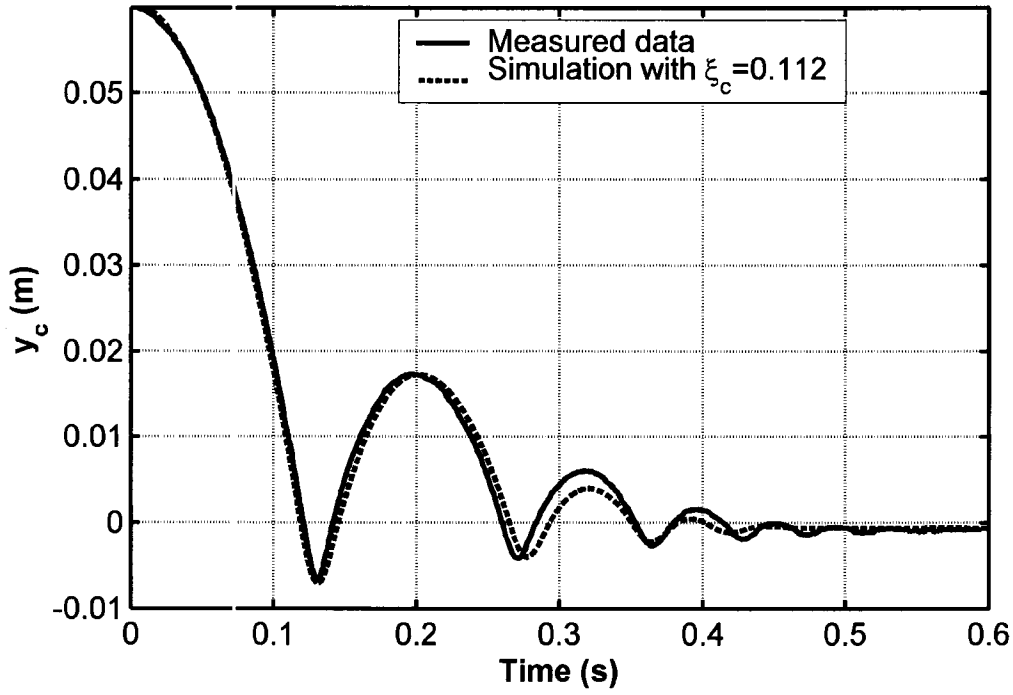


Fig. A.2b. y_c vs. t for Foam 1.

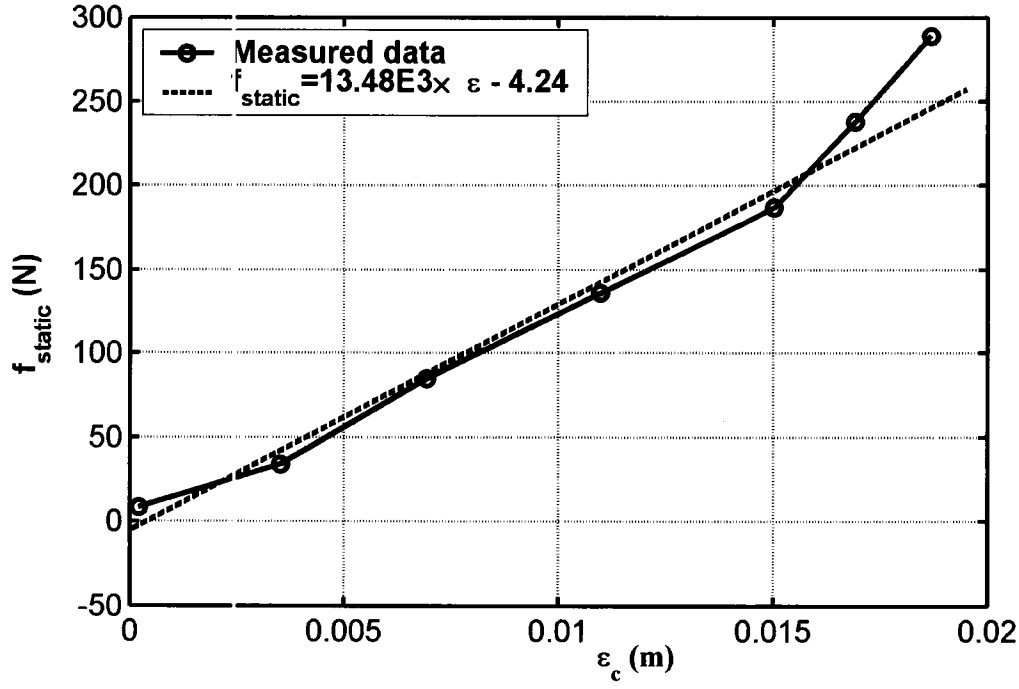


Fig. A.3a. f_{static} vs. ε_c for Foam 2.

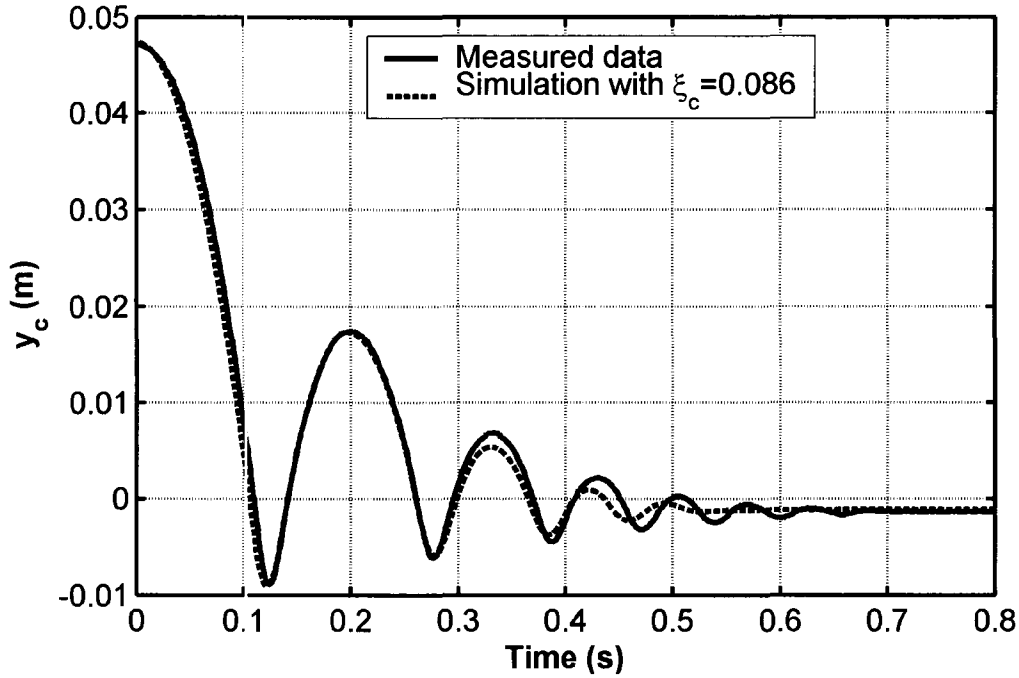


Fig.A.3b. y_c vs. t for Foam 2.

A.2 Design and control of the direct-drive manipulator

A.2.1 Calculating of the moment of inertia of the manipulator

The direct-drive manipulator is an arm directly driven by two electric motors connected in series without a gearbox. The end-effecter is ignored in this design. The shape and size of the link of the manipulator is shown in Fig. A.4. It consists of a long hollow square aluminium tube and a thick circular steel disk. This disk is installed to make the centre of mass of the arm close to the rotary centre of the joint. It has a mass m_{disk} = 1.497 kg. The mass of the tube is m_{tube} = 0.5343 *kg*. The masses of the disk and the tube were measured on a precise scale (SG32000 from Mettler Toledo with 1*g* accuracy). The wall thickness of the tube is b_{tube} = 0.003 m.The density of the material of the tube is equal to

$$\chi_{tube} = \frac{m_{tube}}{l_{tube}w_{tube}h_{tube} - (l_{tube} - b_{tube})(w_{tube} - b_{tube})(h_{tube} - b_{tube})} = 2,700 \text{ kg/m}^3 \quad \text{(A.2.1)}$$

where l_{tube} = 0.664 m, w_{tube} = 0.0508 m and h_{tube} =0.0508 m are the length, width and height of the tube (see Fig. A.4), respectively. $I_{tube,centre}$, the moment of inertia of the tube about its centre of mass is

$$\begin{aligned} I_{tube,centre} &= \tfrac{1}{12}\chi_{tube} l_{tube} w_{tube} h_{tube}\left(l_{tube}^2 + w_{tube}^2\right) - \\ &\tfrac{1}{12}\chi_{tube}(l_{tube} - b_{tube})(w_{tube} - b_{tube})(h_{tube} - b_{tube})\left[(l_{tube} - b_{tube})^2 + (w_{tube} - b_{tube})^2\right] \quad \text{(A.2.2)} \\ &= 0.0208 \text{ kgm}^2 \end{aligned}$$

Its moment of inertia about the joint centre is:

$$I_{tube} = I_{tube,centre} + m_{tube}d_{tube}^2 \quad \text{(A.2.3)}$$

$$= 0.0208 + 0.5343 \times (0.502 - 0.644/2)^2 = 0.038 \text{ kgm}^2$$

where l_{tube} is the length of the tube; d_{tube} is the distance from the centre of mass of the tube to the joint centre. The moment of inertia of the disk about the joint centre is:

$$I_{disk} = I_{disk,centre} + m_{disk}d_{disk}^2 = \tfrac{1}{2}m_{disk}r_{disk}^2 + m_{disk}d_{disk}^2 \quad \text{(A.2.4)}$$

$$= \tfrac{1}{2} \times 1.497 \times 0.0508^2 + 1.497 \times 0.113^2 = 0.021 \text{ kgm}^2$$

where r_{disk} is the radius of the disk; d_{disk} is the distance from the centre of mass of the

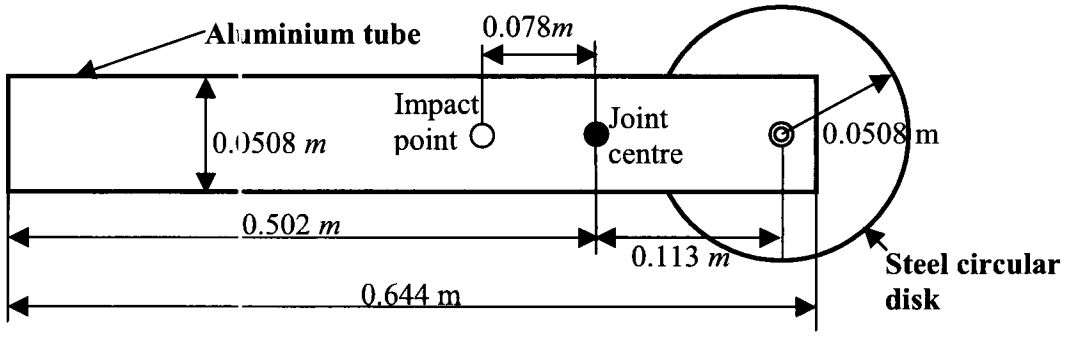


Fig. A.4. The arm of the direct-drive manipulator.

disk to the joint centre. Since the moment of inertia of the joint motor is very small (smaller than 0.0001 kgm^2), the moment of inertia of the manipulator, m_{dd}, is:

$$m_{dd} = I_{tube} + I_{disk} = 0.038 + 0.021 = 0.059 \text{ kgm}^2 \tag{A.2.5}$$

Since the distance between the impact point and the rotary centre of the joint is l_1=0.078 m, the effective mass of the direct-drive manipulator relative at the impact point is:

$$M_r = \frac{m_{11}}{l_1^2} = \frac{m_{dd}}{l_1^2} = \frac{0.059}{0.078^2} = 9.6 \text{ kg} \tag{A.2.6}$$

A.2.2 Identification of the dynamic model of the manipulator

To design the control system for the direct-drive manipulator, the dynamic model of the manipulator is required. The shafts of two DC brush motors (CMC model 3515) are mechanically connected in series with a coupler to rotate the arm of the manipulator. The torque driving the arm equals the sum of the output torques of the two motors. The two motors are controlled by a PC through two identical amplifiers. The amplifier gains are K_{amp} = 1 A/V. Their control commands (*i.e.* voltages) from the PC are also identical. The torque is simply:

$$\tau_{dd} = \sum_{i=1}^{2} K_{t,dd,i} i_{dd,i} = K_{t,dd} K_{amp} u_{dd} \tag{A.2.7}$$

where τ_{dd} is the output torque of the joint; $K_{t,dd,i}$ is the torque constant of i^{th} joint motor; $i_{dd,i}$ is the electric current on i^{th} joint motor. Note that the control commands for the two motors are identical and $i_{dd,i} = K_{amp} u_{dd}$ where u_{dd} is the control command (unit: V) sent

from the PC to the amplifiers. We denote $K_{t,dd} = \sum_{i=1}^{2} K_{t,dd,i}$ as the summed motor torque constant. The viscous-friction and the static friction of the two connected motors are also summed. Hence, the dynamics of the manipulator can be written as:

$$\tau_{dd} = m_{dd}\ddot{\theta}_{dd} + B_{dd}\dot{\theta}_{dd} + \tau_{st,dd}\,\mathrm{sgn}(\dot{\theta}_{dd}) \tag{A.2.8}$$

where θ_{dd} is the angular position of the joint of the manipulator; B_{dd} is the summed motor viscous friction coefficient; and $\tau_{st,dd}$ is the summed motor static friction torque. Substituting (A.2.7) into (A.2.9), we have:

$$K_{t,dd}K_{amp}u_{dd} = m_{dd}\ddot{\theta}_{dd} + B_{dd}\dot{\theta}_{dd} + \tau_{st,dd}\,\mathrm{sgn}(\dot{\theta}_{dd}) \tag{A.2.9}$$

It can be rewritten as:

$$\ddot{\theta}_{dd} = \frac{K_{t,dd}K_{amp}}{m_{dd}}u_{dd} - \frac{B_{dd}}{m_{dd}}\dot{\theta}_{dd} - \frac{\tau_{st,dd}\,\mathrm{sgn}(\dot{\theta}_{dd})}{m_{dd}} \tag{A.2.10}$$

Open-loop tests will be performed to obtain $\frac{K_{t,dd}K_{amp}}{m_{dd}}$, $\frac{B_{dd}}{m_{dd}}$ and $\frac{\tau_{st,dd}}{m_{dd}}$ for different values of the input u_{dd}. The procedure is as follows:

1. Perform open-loop tests by sending different commands (*i.e.* u_{dd} = 0.3, 0.4, 0.6, 0.8 and 1.0 V), send each command twice. Obtain position profiles θ_{dd} vs. time t for those tests.
2. Let us set $\kappa_{dd} = B_{dd}/m_{dd}$. Since u_{dd} and $\tau_{st,dd}$ are constants for these open-loop tests, $A_{dd} = K_{t,dd}K_{amp}u_{dd}/m_{dd} - \tau_{st,dd}\,\mathrm{sgn}(\dot{\theta}_{dd})/m_{dd}$ is a constant, and we can rewrite (A.2.10) as

$$\ddot{\theta}_{dd} = A_{dd} - \kappa_{dd}\dot{\theta}_{dd}\,. \tag{A.2.11}$$

3. Use the Laplace transform and inverse Laplace transform to solve (A.2.11), to obtain θ_{dd} in the time domain:

$$\theta_{dd} = \kappa_{dd}\left(A_{dd}t - 1 + e^{-A_{dd}t}\right)/A_{dd} \tag{A.2.12}$$

4. Use the function *lspcurvefit* in MATLAB to fit the position profiles, θ_{dd} vs. t in the least-squares sense to find the values of A_{dd} and κ_{dd}. For the fitting, the initial values of those parameters A_{dd} and κ_{dd} are calculated with the values of $K_{t,dd}$, B_{dd} and $\tau_{st,dd}$ from the datasheet of the motor (Cleveland Motion Control, Brush servo motor 3500 series). The curve fitting results are listed in Table A.1.

Table A.1. Parameters from curve fitting

Input Voltage (u)	κ_{dd}		A_{dd}		**Mean of A_{dd} for the two tests**
	Test 1	**Test 2**	**Test 1**	**Test 2**	
0.3 V	0.121	0.161	0.593	0.400	0.497
0.4 V	0.098	0.110	0.736	0.880	0.808
0.6 V	0.109	0.113	1.452	1.556	1.504
0.8 V	0.138	0.138	2.540	2.569	2.555
1.0 V	0.184	0.183	3.722	3.724	3.723

5. The values of κ_{dd} in Table A.1 varies with different input voltages. To simplify the task, we will use the mean value for designing the control system, which is $\kappa_{dd} = B_{dd}/m_{dd} = 0.1355\ \text{s}^{-1}$. Since m_{dd} = 0.059 kgm^2, we have B_{dd} = 0.0079 Nm/s, slightly larger than the datasheet value (0.0076 Nm/s).

6. For the parameters $K_{t,dd}$ and $\tau_{st,dd}$ since

$$A_{dd} = K_{t,dd}K_{amp}u_{dd}/m_{dd} - \tau_{st,dd}\,\mathrm{sgn}(\dot{\theta}_{dd})/m_{dd}\,, \tag{A.2.13}$$

Let us set $\Theta_{dd} = K_{t,dd}K_{amp}/m_{dd}$ and $\Gamma_{dd} = -\tau_{st,dd}\,\mathrm{sgn}(\dot{\theta}_{dd})/m_{dd}$ are constants. Let us set $\mathbf{Y}_{dd}$ = [0.497 0.808 1.504 2.555 3.723] for the two tests, *i.e.* the data in last column of Table A.1, and $\mathbf{U}_{dd}$ = [0.3 0.4 0.6 0.8 1.0] the data in first column of Table A.1. Then (A.2.13) can be rewritten in the format of first order linear function $\mathbf{Y}_{dd} = \Theta_{dd}\mathbf{U}_{dd} + \Gamma_{dd}$. We use the linear regression method to obtain Θ_{dd} and Γ_{dd}:

$$\Theta_{dd} = \frac{5\mathbf{U}_{dd}\mathbf{Y}_{dd}^{'} - \sum\mathbf{U}_{dd}\sum\mathbf{Y}_{dd}}{5\mathbf{U}_{dd}\mathbf{U}_{dd}^{'} - \left(\sum\mathbf{U}_{dd}\right)^2}; \tag{A.2.14a}$$

$$\Gamma_{dd} = \left(\sum\mathbf{Y}_{dd} - \Theta\sum\mathbf{U}_{dd}\right)/5 \tag{A.2.14b}$$

Solving (A 2.14a) and (A.2.14b), we have Θ_{dd} = 4.59 and Γ_{dd} = -1.03. Therefore, $K_{t,dd}$ = 0.281 Nm/A, smaller than the datasheet value of 0.283 Nm/A (*i.e.* 0.142 Nm/A for each motor) and $\tau_{st,dd}$ = 0.061 Nm, larger than the datasheet value (0.057 Nm). So we have the dynamic model of the manipulator:

$$\ddot{\theta}_{dd} = 4.59u_{dd} - 0.136\dot{\theta}_{dd} - 1.03\mathrm{sgn}(\dot{\theta}_{dd})\,. \tag{A.2.15}$$

Figure A.5 displays the comparison between the open-loop test result and the simulation data from (A.2.15) for u_{dd} = 0.6 V. We can see they are in good agreement.

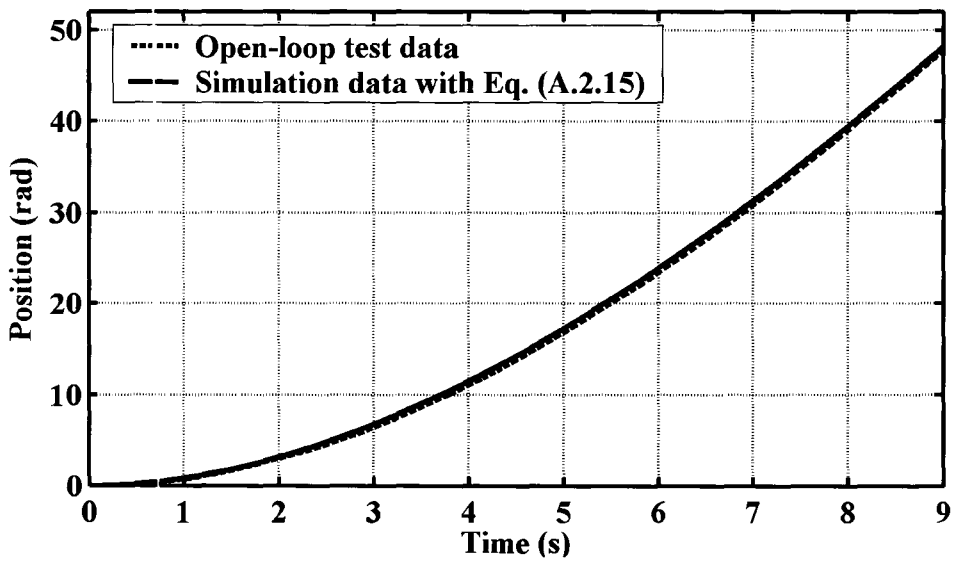


Fig. A.5. Comparison of the simulation data from (A.2.15) and the open-loop test data for $u_{dd} = 0.6$ V.

A.2.3 Design of the manipulator controller

The manipulator controller is designed as the combination of PD+FF plus friction compensation:

$$u_{dd} = \underbrace{\left(K_{p,dd}e_{dd} + K_{d,dd}\dot{e}_{dd}\right)}_{PD\ Feedback} + \underbrace{\frac{m_{dd}}{K_{t,dd}K_{amp}}\ddot{\theta}_{d,dd}}_{\substack{Acceleration \\ Feedforward}} + \underbrace{\frac{B_{dd}}{K_{t,dd}K_{amp}}\dot{\theta}_{d,dd}}_{\substack{Viscous\ Friction \\ Feedforward}} + \underbrace{\frac{\tau_{st,dd}}{K_{t,dd}K_{amp}}\mathrm{sgn}(\dot{\theta}_{dd})}_{\substack{Static\ Friction \\ Feedforward}} \tag{A.2.16}$$

where $e_{dd} = \theta_{d,dd} - \theta_{dd}$, $\theta_{d,dd}$ is the desired angular position of the direct-drive manipulator, $\dot{\theta}_{d,dd}$ is the desired angular velocity, $\ddot{\theta}_{d,dd}$ is the desired angular acceleration and $K_{p,dd}$ and $K_{d,dd}$ are the PD controller gains. Substituting (A.2.16) into (A.2.12) gives:

$$\ddot{\theta}_{dd} - \ddot{\theta}_{d,dd} = \frac{K_{t,dd}K_{amp}}{m_{dd}}\left(K_{P,dd}e_{dd} + K_{D,dd}\dot{e}_{dd}\right) + \frac{B_{dd}}{m_{dd}}\dot{e}_{dd} \tag{A.2.17}$$

Since the manipulator is desired to move at a constant velocity, $\ddot{\theta}_{d,dd} = 0$. (A.2.17) is rewritten as:

$$\ddot{\theta}_{dd} m_{dd} = K_{t,dd} K_{amp} K_{p,dd} e_{dd} + \left(K_{t,dd} K_{amp} K_{d,dd} + B_{dd} \right) \dot{e}_{dd} \tag{A.2.18}$$

Note that in (A.2.18), the external force (*e.g.* the impact force) is not included. So during the impact, the manipulator's dynamics is:

$$-\tau_c = \ddot{\theta}_{dd} m_{dd} - K_{t,dd} K_{amp} K_{p,dd} e_{dd} - \left(K_{t,dd} K_{amp} K_{d,dd} + B_{dd} \right) \dot{e}_{dd} \tag{A.2.19}$$

where τ_c is the torque caused by the impact force. Note that in (A.2.19a) $e_{dd} = \theta_{d,dd} - \theta_{dd}$. Compare to (3.5.3), since $-e_{dd} = \theta_J - \theta_d$, the equations (A.2.19) and (3.5.3) are identical. Since this manipulator is directly driven by the DC motors, the mechanical stiffness is much larger than the control stiffness. Comparing with (3.5.9a) and (3.5.9b), the joint stiffness and damping of the manipulator during an impact is:

$$K_J = K_{t,dd} K_{amp} K_{p,dd} \tag{A.2.20a}$$

$$C_J = K_{t,dd} K_{amp} K_{d,dd} + B_{dd} \tag{A.2.20b}$$

Then substituting (A.2.20a) and (A.2.20b) into (3.5.17) and reorganizing the result, we have:

$$K_{p,dd} = \frac{K_r l_1^2}{K_{t,dd} K_{amp}} \text{ and} \tag{A.2.21a}$$

$$K_{d,dd} = \frac{\left(C_r - B_{dd} \right) l_1^2}{K_{t,dd} K_{amp}}. \tag{A.2.21b}$$

If the manipulator's stiffness is set as $K_r = 20$ kN/m and its damping factor is set as

$C_r = 250$ Ns/m, the PD controller gains for the manipulator are calculated as:

$$K_{p,dd} = \frac{K_r l_1^2}{K_{t,dd} K_{amp}} = \frac{20000 \times 0.078^2}{0.2807} = 433.4 \text{ V and}$$

$$K_{d,dd} = \frac{(C_r - B_{dd}) l_1^2}{K_{t,dd} K_{amp}} = \frac{(250 - 0.0079) \times 0.078^2}{0.2807} = 5.4 \text{ Vs}.$$

Figure A.6 shows a closed-loop experiment result for a constant desired velocity of 5.2 rad/s (*i.e.* the impact point velocity is $5.2 \times 0.078 = 0.4$ m/s). With our hardware $|u_{dd}| \leq 5$ V. Note that the impact will happen when the manipulator rotates to about 4.3 rad. The velocity of the manipulator is constant at that time.

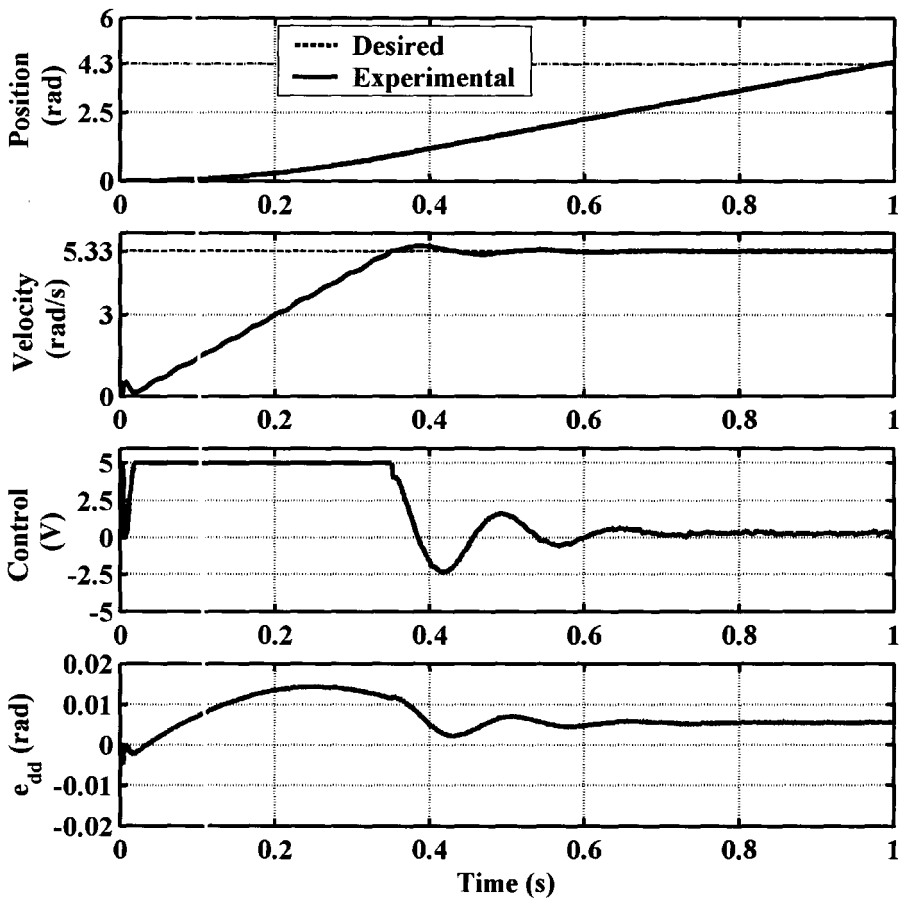


Fig A.6. The closed-loop experiment result for the direct-drive manipulator for a constant desired velocity of 5.2 rad/s.

A.3 Control system for the pneumatic cylinder acting as the human torso

The pneumatic cylinder acts as the human torso in the head-neck-torso apparatus (see Figs. 4.1 and 4.2). A control system is required to make it move at a constant velocity before the impact. The design of the control system for the pneumatic cylinder is based on the thesis of Ning (2004). The controller is the FF plus PD controller:

$$u_{pnm} = K_{ffa}\ddot{p}_{d,pnm} + K_{ffv}\dot{p}_{d,pnm} + K_{p,pnm}\left(p_{d,pnm} - p_{pnm}\right) + K_{d,pnm}\left(\dot{p}_{d,pnm} - \dot{p}_{pnm}\right) \quad \text{(A.3.1)}$$

where $u_{d,pnm}$ is the control command for the valve of the pneumatic cylinder; $p_{d,pnm}$ is the desired position of the piston in the pneumatic cylinder; p_{pnm} is the actual position piston of the cylinder, and according to Ning (2004) $K_{ffa,pnm} = 0.34$, $K_{ffv,pnm} = 0.44$, $K_{p,pnm} = 208.5$ and $K_{d,pnm} = 14.9$. The sample time of the control system is $T_s = 0.001$ s.

Fig. A.7 presents a closed-loop experiment result with (A.3.1). We can see that the velocity of the piston oscillates. Since the simulated human head is connected with springs to the piston and slides on it, accelerations of the piston will cause oscillations of the head. The oscillations will make the head acceleration vary during the impact experiments and will influence the measured head acceleration attributed to the impact. Therefore, to prevent significantly influencing the experimental results, during the impact experiments the manipulator must collide with the apparatus after the head oscillation settles down. We can see that in Fig. A.7 the oscillation after $p_{pnm} > 0.23$ m (*i.e.* when $t > 0.33\,s$) has nearly settled down. Therefore, the impact must happen after the cylinder moves over 0.23 m. 0.26 m (*i.e.* when $t = 0.36\,s$) is selected in the impact experiments. At that position, the head acceleration is nearly zero. Note the piston position is limited to [0, 0.33] m.

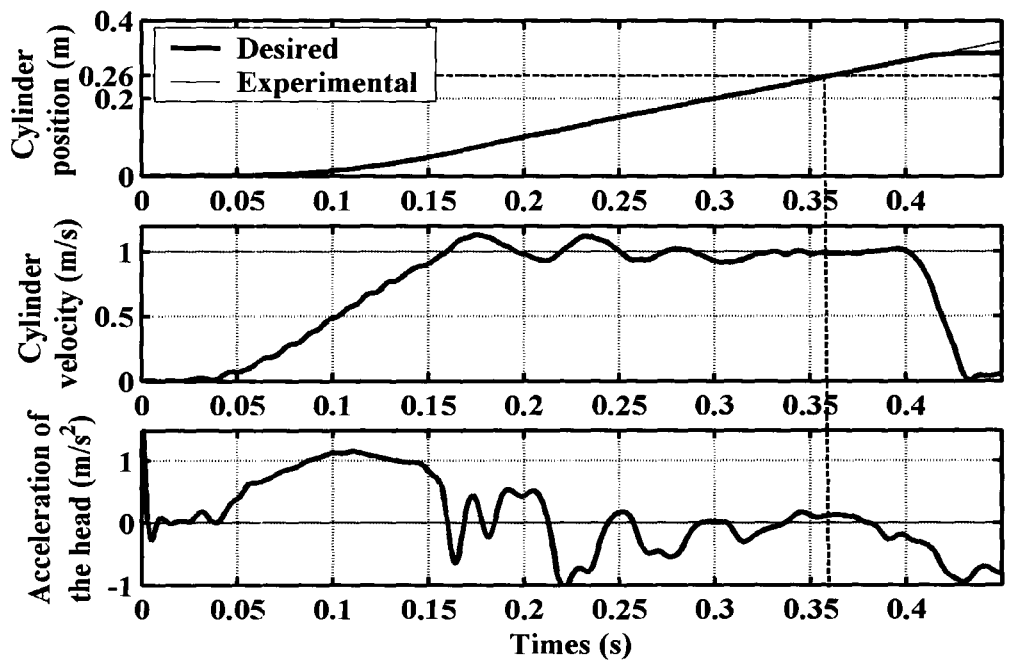


Fig. A.7. The position, velocity of the piston and the head acceleration in a closed-loop experiment for a constant desired velocity of 1 m/s.

Appendix B.

Stability and continuity analysis for the new VFF

B.1 The Lyapunov function for the new VFF

Since the robot is modeled as a point mass, stability of our VFF implies stability of the robot motion. Our stability analysis is based on Lyapunov's second method (Lyapunov 1892). We will first analyze the stability for the case of a single obstacle. The following form of Lyapunov function candidate will be used:

$$V_i(\mathbf{X}_i) = \tfrac{1}{2}\mathbf{X}_i^{\mathrm{T}}\mathbf{Q}_i\mathbf{X}_i \tag{B.1.1}$$

where $\mathbf{X}_i$ is the state vector of the VFF in the i^{th} condition and $\mathbf{Q}_i$ is a positive definite matrix. Recall that our VFF is a piecewise force field. When $\mathbf{P}_r \in \mathbb{C}_1$, $\mathbf{F}_v = K_1\mathbf{E} + K_2\dot{\mathbf{E}}$ only depends on $\mathbf{E} = \begin{bmatrix} E_X & E_Y \end{bmatrix}^{\mathrm{T}}$ and $\dot{\mathbf{E}} = \begin{bmatrix} \dot{E}_X & \dot{E}_Y \end{bmatrix}^{\mathrm{T}}$ so the state vector is $\mathbf{X}_1 = \begin{bmatrix} \mathbf{E}^{\mathrm{T}} & \dot{\mathbf{E}}^{\mathrm{T}} \end{bmatrix}^{\mathrm{T}}$ and the only equilibrium point is the origin. The proof is as follows:

With the robot dynamics, (5.1.1), since the goal is stationary we have:

$$\ddot{\mathbf{E}} = -\mathbf{a}_{mr} = -\frac{\mathbf{F}_v}{m_{mr}} = -\frac{K_1}{m_{mr}}\mathbf{E} - \frac{K_2}{m_{mr}}\dot{\mathbf{E}} \tag{B.1.2}$$

Next, we can represent the control system (*i.e.* $\mathbf{P}_r \in \mathbb{C}_1$) in the state space format:

$$\begin{bmatrix} \dot{E}_x \\ \dot{E}_y \\ \ddot{E}_x \\ \ddot{E}_y \end{bmatrix} = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ -K_1/m_{mr} & 0 & -K_2/m_{mr} & 0 \\ 0 & -K_1/m_{mr} & 0 & -K_2/m_{mr} \end{bmatrix} \begin{bmatrix} E_x \\ E_y \\ \dot{E}_x \\ \dot{E}_y \end{bmatrix} \tag{B.1.3}$$

From (B.1.3), $\begin{bmatrix}\dot{E}_x & \dot{E}_y & \ddot{E}_x & \ddot{E}_y\end{bmatrix} = \mathbf{0} \Rightarrow E_x = 0,\ E_y = 0,\ \dot{E}_x = 0$ and $\dot{E}_y = 0$. The origin therefore is proven to be the only equilibrium point when $\mathbf{P}_r \in \mathbb{C}_1$.

Note that being at the origin of the state space is equivalent to the robot being stopped at its goal. When $\mathbf{P}_r \in \mathbb{C}_2$, the VFF state vector is $\mathbf{X}_2 = \begin{bmatrix}\mathbf{E}^{\mathrm{T}} & \dot{\mathbf{E}}^{\mathrm{T}} & \Lambda & \dot{\Lambda}^* & \psi & \dot{\psi}^*\end{bmatrix}^{\mathrm{T}}$. The piecewise Lyapunov function candidate for our VFF is:

$$V = \begin{cases} V_1 = \frac{1}{2}\mathbf{X}_1^{\mathrm{T}}\mathbf{Q}_1\mathbf{X}_1, & \text{if } P_r \in \mathbb{C}_1 \\ V_2 = \frac{1}{2}\mathbf{X}_2^{\mathrm{T}}\mathbf{Q}_2\mathbf{X}_2, & \text{if } P_r \in \mathbb{C}_2 \end{cases} \tag{B.1.4}$$

where $\mathbf{Q}_2 = \begin{bmatrix}K_1^2 & K_1^2 & M_r K_1 & M_r K_1 & K_3^2 & h_\Lambda & K_5^2 & h_\psi\end{bmatrix} \cdot \mathbf{I}_8$ for $h_\Lambda = M_r K_3 \dfrac{(d - r_3)^2}{(r_2 - d)^2}$, $h_\psi = M_r K_5 \dfrac{d}{(r_2 - d)^2}$ are positive time varying coefficients since $d > r_3$; and $\mathbf{Q}_1 = \begin{bmatrix}K_1^2 & K_1^2 & M_r K_1 & M_r K_1\end{bmatrix} \cdot \mathbf{I}_4$. Note that when $\mathbf{P}_r \in \mathbb{C}_3$ the robot is commanded to stop, and VFF is not used. Johansson (2002) stated in chapter 4 of his book that a piecewise Lyapunov function candidate must be continuous to analyze the stability of the piecewise system. Eq. (B.1.4) is continuous at the boundary between $\mathbb{C}_1$ and $\mathbb{C}_2$. The proof is as follows:

When $\mathbf{P}_r \in \mathbb{C}_1$

$$\lim_{d \to r_2, P_r \in \mathbb{C}_1} V = \lim_{d \to r_2} V_1 = \lim_{d \to r_2} \tfrac{1}{2}K_1^2\mathbf{E}^{\mathrm{T}}\mathbf{E} + \tfrac{1}{2}M_r K_1 \dot{\mathbf{E}}^{\mathrm{T}}\dot{\mathbf{E}} = \tfrac{1}{2}K_1^2\mathbf{E}^{\mathrm{T}}\mathbf{E} + \tfrac{1}{2}M_r K_1 \dot{\mathbf{E}}^{\mathrm{T}}\dot{\mathbf{E}} \tag{B.1.5}$$

When $\mathbf{P}_r \in \mathbb{C}_2$

$$\lim_{d\to r_2, P_r\in\mathbb{C}_2} V = \lim_{d\to r_2} V_2 = \lim_{d\to r_2}\left(\tfrac{1}{2}K_1^2\mathbf{E}^{\mathrm{T}}\mathbf{E} + \tfrac{1}{2}M_rK_1\dot{\mathbf{E}}^{\mathrm{T}}\dot{\mathbf{E}} + \tfrac{1}{2}K_3^2\Lambda^2 + \tfrac{1}{2}M_rK_3h_\Lambda\dot{\Lambda}^{*2} + \tfrac{1}{2}K_5^2\psi^2 + \tfrac{1}{2}M_rK_5h_\psi\dot{\psi}^{*2}\right) \tag{B.1.5}$$

Since

$$\lim_{d\to r_2}\Lambda = \lim_{d\to r_2}(r_2-d)^2/(d-r_3) = 0, \tag{B.1.6a}$$

$$\lim_{d\to r_2}\dot{\Lambda}^* = \lim_{d\to r_2} -\dot{d}(r_2-d)^2/(d-r_3)^2 = 0, \tag{B.1.6b}$$

$$\lim_{d\to r_2}\psi = \lim_{d\to r_2}(r_2-d)^2\Phi = 0, \text{ and} \tag{B.1.6c}$$

$$\lim_{d\to r_2}\dot{\psi}^* = \lim_{d\to r_2}(r_2-d)^2\dot{\Phi} = 0, \tag{B.1.6d}$$

we have:

$$\lim_{d\to r_2, P_r\in\mathbb{C}_2} V = \tfrac{1}{2}K_1^2\mathbf{E}^{\mathrm{T}}\mathbf{E} + \tfrac{1}{2}M_rK_1\dot{\mathbf{E}}^{\mathrm{T}}\dot{\mathbf{E}} = \lim_{d\to r_2, P_r\in\mathbb{C}_1} V \tag{B.1.7}$$

Eq. (B.1.7) proves the continuity of (B.1.4).

B.2 Stability analysis of the piecewise Lyapunov function candidate

B.2.1 Stability analysis for V_1

The first derivative of (B.1.4) when $\mathbf{P}_r \in \mathbb{C}_1$ is:

$$\begin{aligned}\dot{V}_1 &= K_1^2\mathbf{E}^{\mathrm{T}}\dot{\mathbf{E}} + M_rK_1\dot{\mathbf{E}}^{\mathrm{T}}\ddot{\mathbf{E}}\\ &= K_1^2\mathbf{E}^{\mathrm{T}}\dot{\mathbf{E}} + K_1\dot{\mathbf{E}}^{\mathrm{T}}\left(-K_1\mathbf{E} - K_2\dot{\mathbf{E}}\right)\\ &= -K_1K_2\dot{\mathbf{E}}^{\mathrm{T}}\dot{\mathbf{E}} \le 0\end{aligned} \tag{B.2.1}$$

Therefore the function $\dot{V}_1$ is negative semi-definite. Applying Lasalle's theorem (Spong and Vidyasagar 1989), if $\dot{V}_1 \equiv 0$ then $\dot{\mathbf{E}} \equiv 0$ and hence $\ddot{\mathbf{E}} \equiv 0$. From (B.7), if $\ddot{\mathbf{E}} = 0$ then $\mathbf{F}_v = 0$. Since $\dot{\mathbf{E}} = 0$ and $\mathbf{F}_v = 0$, (5.4.2) and (5.4.7) imply that $\mathbf{E} = 0$. Hence we can

conclude the VFF is asymptotically stable at the origin subject to the condition of the robot being inside $\mathbb{C}_1$.

B.2.2 Stability analysis for V_2

When $\mathbf{P}_r \in \mathbb{C}_2$, (B.1.4) can be rewritten as:

$$V_2 = \tfrac{1}{2}K_1^2\mathbf{E}^{\mathrm{T}}\mathbf{E} + \tfrac{1}{2}M_rK_1\dot{\mathbf{E}}^{\mathrm{T}}\dot{\mathbf{E}} + \tfrac{1}{2}K_3^2\Lambda^2 - \tfrac{1}{2}M_rK_3\dot{\Lambda}^*\dot{d} + \tfrac{1}{2}K_5^2\psi^2 + \tfrac{1}{2}M_rK_5d\dot{\psi}^*\Phi \quad \text{(B.2.2)}$$

Note that $\dot{\Lambda}^* = -K_v\dot{d}$ and $K_v > 0$, so $-\tfrac{1}{2}M_rK_3\dot{\Lambda}^*\dot{d} \geq 0$ in (B.2.2). The first derivative of V_2 is:

$$\begin{aligned}\dot{V}_2 = {} & K_1^2\mathbf{E}^{\mathrm{T}}\dot{\mathbf{E}} + M_rK_1\dot{\mathbf{E}}^{\mathrm{T}}\ddot{\mathbf{E}} + K_3^2\Lambda\dot{\Lambda}^* - K_3^2\frac{2\dot{d}(r_2-d)}{d-r_3}\Lambda - M_rK_3\dot{\Lambda}^*\ddot{d} \\ & + K_5^2\psi(r_2-d)^2\dot{\Phi} + M_rK_5(r_2-d)^2d\dot{\Phi}\ddot{\Phi} - \tfrac{1}{2}M_rK_3\dot{K}_\Lambda\dot{d}^2 - \\ & K_5^2(r_2-d)\dot{d}\Phi^2 + \tfrac{1}{2}M_rK_5(r_2-d)^2\dot{d}\dot{\Phi}^2 - M_rK_5(r_2-d)d\dot{d}\dot{\Phi}^2\Big]\end{aligned} \quad \text{(B.2.3)}$$

Since

$$\ddot{d} = (\mathbf{a}_o - \mathbf{a}_{mr})^{\mathrm{T}}\mathbf{u}_\Lambda \quad \text{(B.2.4)}$$

and

$$\begin{aligned}\ddot{\Phi} = \ddot{\alpha} - \ddot{\beta} &= \left(-\mathbf{a}_o^{\mathrm{T}}\mathbf{u}_\psi\right)/W - \left(\mathbf{a}_{mr}^{\mathrm{T}}\mathbf{u}_\psi - \mathbf{a}_o^{\mathrm{T}}\mathbf{u}_\psi\right)/d \\ &= -\left[-\mathbf{a}_{mr}^{\mathrm{T}}\mathbf{u}_\psi - \mathbf{a}_o^{\mathrm{T}}\mathbf{u}_\psi\left(1-\frac{d}{W}\right)\right]/d = \left(\mathbf{a}_o^{*\mathrm{T}}\mathbf{u}_\psi - \mathbf{a}_{mr}^{\mathrm{T}}\mathbf{u}_\psi\right)/d\end{aligned} \quad \text{(B.2.5)}$$

where $\mathbf{a}_o$ is the acceleration vector of the obstacle; and $W = \|\mathbf{W}\|_2$; we have :

$$\begin{aligned}\dot{V}_2 = {} & K_1^2\mathbb{E}^{\mathrm{T}}\dot{\mathbf{E}} + K_1\dot{\mathbf{E}}^{\mathrm{T}}\left(-K_1\mathbf{E} - K_2\dot{\mathbf{E}} - K_4\dot{\Lambda}^*\mathbf{u}_\Lambda - K_6\dot{\psi}^*\mathbf{u}_\psi\right) + K_3^2\Lambda\dot{\Lambda}^* - \\ & K_3\dot{\Lambda}^*\left(K_2\dot{\mathbf{E}}^{\mathrm{T}}\mathbf{u}_\Lambda + K_3\Lambda + K_4\dot{\Lambda}^*\right) + K_5^2\psi\dot{\psi}^* + \\ & K_5\dot{\psi}^*\left(-K_2\dot{\mathbf{E}}^{\mathrm{T}}\mathbf{u}_\psi - K_5\psi - K_6\dot{\psi}^*\right) - \left(K_1\dot{\mathbf{E}} + K_3\dot{\Lambda}^*\mathbf{u}_\Lambda + K_5\dot{\psi}^*\mathbf{u}_\psi\right)^{\mathrm{T}}\mathbf{F}_L + \Omega\end{aligned} \quad \text{(B.2.6)}$$

where

$$\Omega = -K_1\dot{\mathbf{E}}^{\mathrm{T}}\left(K_3\Lambda\mathbf{u}_\Lambda + K_5\psi\mathbf{u}_\psi\right) - K_3^2\frac{2\dot{d}(r_2-d)}{d-r_3}\Lambda + \tfrac{1}{2}M_rK_3\dot{K}_\Lambda\dot{d}^2 +$$
$$K_5\dot{\psi}^*\left(\mathbf{a}_o^{*\mathrm{T}}\mathbf{u}_\psi - K_1\mathbf{E}^{\mathrm{T}}\mathbf{u}_\psi\right) - K_3\dot{\Lambda}^*\left(K_1\mathbf{E}^{\mathrm{T}}\mathbf{u}_\Lambda - \mathbf{a}_o^{\mathrm{T}}\mathbf{u}_\Lambda\right) + \tfrac{1}{2}M_rK_5(r_2-d)^2\dot{d}\dot{\Phi}^2 \quad \text{(B.2.7)}$$
$$-M_rK_5(r_2-d)d\dot{d}\dot{\Phi}^2 - K_5^2\left(r_2-d\right)\dot{d}\Phi^2$$

and

$$\dot{K}_\Lambda = -2\dot{d}\left(\frac{\left(r_2-d\right)^2}{\left(d-r_3\right)^3} + \frac{\left(r_2-d\right)}{\left(d-r_3\right)^2}\right) \quad \text{(B.2.8)}$$

Assuming:

$$\frac{K_2}{K_1} = \frac{K_4}{K_3} = \frac{K_6}{K_5} = b \quad \text{(B.2.9)}$$

where $b>0$; and substituting $(\dot{\Lambda}^*)^2 = \dot{\Lambda}^*\mathbf{u}_\Lambda^{\mathrm{T}}\dot{\Lambda}^*\mathbf{u}_\Lambda$, $\left(\dot{\psi}^*\right)^2 = \dot{\psi}^*\mathbf{u}_\psi^{\mathrm{T}}\dot{\psi}^*\mathbf{u}_\psi$, $\mathbf{u}_\psi^{\mathrm{T}}\mathbf{u}_\Lambda = \mathbf{u}_\Lambda^{\mathrm{T}}\mathbf{u}_\psi = 0$, $\dot{\psi}^* = (r_2-d)^2\dot{\Phi}$ and $\mathbf{v}_L = K_1\dot{\mathbf{E}} + K_3\dot{\Lambda}^*\mathbf{u}_\Lambda + K_5\dot{\psi}^*\mathbf{u}_\psi$ into (B.2.6), (B.2.6) can be simplified as:

$$\dot{V}_2 = -b\left\|\mathbf{v}_L\right\|_2^2 - \mathbf{v}_L{}^{\mathrm{T}}\mathbf{F}_L + \Omega \quad \text{(B.2.10)}$$

To make $\dot{V}_2$ negative semi-definite, we need to set:

$$\mathbf{F}_L = \begin{cases} \dfrac{\Omega\mathbf{u}_L}{\left\|\mathbf{v}_L\right\|_2} & \Omega > 0 \\ 0 & \Omega \le 0 \end{cases} \quad \text{(B.2.11)}$$

Comparing to (5.4.6), we have:

$$K_L = \begin{cases} \Omega & \Omega > 0 \\ 0 & \Omega \le 0 \end{cases} \quad \text{(B.2.12)}$$

Then the first derivative of (B.2.2) is:

$$\dot{V}_2 = \begin{cases} -b\|\mathbf{v}_L\|_2^2 & \Omega > 0 \\ -b\|\mathbf{v}_L\|_2^2 + \Omega & \Omega \leq 0 \end{cases} \tag{B.2.13}$$

We have:

$$\dot{V}_2 \leq -b\|\mathbf{v}_L\|_2^2 \leq 0 \tag{B.2.14}$$

From inequality (B.2.14), $\dot{V}_2 \leq 0$ and $\dot{V}_2 = 0$ only when $\mathbf{v}_L = 0$. It will now be explained how the stabilizing virtual force, $\mathbf{F}_L$, prevents $\mathbf{v}_L = \mathbf{0}$ except for the case $\dot{\mathbf{E}} = \mathbf{0}$, $\dot{\Lambda} = 0$ and $\dot{\psi}^* = 0$. When $\mathbf{v}_L \to 0$, and $\dot{\mathbf{E}} \neq \mathbf{0}, \dot{\Lambda}^* \neq 0$ and $\dot{\psi}^* \neq 0$, the magnitude of $\mathbf{F}_L$ will increase significantly since $K_L \neq 0$ and the denominator of (5.4.6) will approach zero. The force will drive the robot such that $\|\mathbf{v}_L\|_2 \neq 0$. The proof is as follows:

When $\mathbf{v}_L \to 0$, $K_L \neq 0$ and $\|\mathbf{F}_L\|_2 \to \infty$, so $\mathbf{F}_L$ will be much larger than the other forces. So, we have:

$$\mathbf{F}_v \approx \mathbf{F}_L \tag{B.2.15}$$

With the robot dynamics, (5.1.1), the robot acceleration is:

$$\mathbf{a}_{mr} = \mathbf{F}_v / m_{mr} \approx \mathbf{F}_L / m_{mr} \tag{B.2.16}$$

Then with this acceleration of the robot, after an infinitesimal time interval Δt, we have:

$$\begin{aligned} \mathbf{v}_{L,\Delta t} &= K_1\left(\dot{\mathbf{E}} - \mathbf{a}_{mr}\Delta t\right) + K_3\left\{\dot{\Lambda}^*\mathbf{u}_\Lambda + K_\Lambda\left[\left(\mathbf{a}_o - \mathbf{a}_{mr}\right)^{\mathrm{T}}\mathbf{u}_\Lambda\right]\mathbf{u}_\Lambda\Delta t\right\} + \\ &\quad K_3\left[\dot{\psi}^*\mathbf{u}_\psi + K_\psi\left[\left(\mathbf{a}_o - \mathbf{a}_{mr}\right)^{\mathrm{T}}\mathbf{u}_\psi\right]\mathbf{u}_\psi\Delta t\right] \\ &= \mathbf{v}_L + \Delta t\left[-K_1\mathbf{a}_{mr} + \right. \\ &\quad \left. K_3K_\Lambda\left[\left(\mathbf{a}_o - \mathbf{a}_{mr}\right)^{\mathrm{T}}\mathbf{u}_\Lambda\right]\mathbf{u}_\Lambda + K_3K_\psi\left[\left(\mathbf{a}_o - \mathbf{a}_{mr}\right)^{\mathrm{T}}\mathbf{u}_\psi\right]\mathbf{u}_\psi\right] \end{aligned} \tag{B.2.17}$$

where $K_\psi = \dfrac{(r_2 - d)^2}{d} \geq 0$. when $\mathbf{v}_L \to 0$, $\|\mathbf{a}_{mr}\|_2 \to \infty$ since $\|\mathbf{F}_L\|_2 \to \infty$ and $\mathbf{a}_{mr} \gg \mathbf{a}_o$ and $\mathbf{v}_{L,\Delta t}$ is

$$\mathbf{v}_{L,\Delta t} \approx \left(K_1 \mathbf{a}_{mr} + K_3 K_\Lambda \left(\mathbf{a}_{mr}^{\mathrm{T}} \mathbf{u}_\Lambda\right)\mathbf{u}_\Lambda + K_5 K_\psi \left(\mathbf{a}_{mr}^{\mathrm{T}} \mathbf{u}_\psi\right)\mathbf{u}_\psi\right)\Delta t \qquad \text{(B.2.18a)}$$

The norm of $\mathbf{v}_{L,\Delta t}$ is

$$\left\|\mathbf{v}_{L,\Delta t}\right\|_2 \approx \left\|K_1 \mathbf{a}_{mr} + K_3 K_\Lambda \left(\mathbf{a}_{mr}^{\mathrm{T}} \mathbf{u}_\Lambda\right)\mathbf{u}_\Lambda + K_5 K_\psi \left(\mathbf{a}_{mr}^{\mathrm{T}} \mathbf{u}_\psi\right)\mathbf{u}_\psi\right\|_2 \Delta t \qquad \text{(B.2.18b)}$$

In (B.2.18b), $K_1 \mathbf{a}_{mr} + K_3 K_\Lambda \left(\mathbf{a}_{mr}^{\mathrm{T}} \mathbf{u}_\Lambda\right)\mathbf{u}_\Lambda + K_5 K_\psi \left(\mathbf{a}_{mr}^{\mathrm{T}} \mathbf{u}_\psi\right)\mathbf{u}_\psi = \mathbf{0}$ only happens when the three elements in this equation generate a right triangle, as shown in Fig. B.1a. Assuming the vector $\mathbf{a}_{mr}$ has been given, $\mathbf{u}_\Lambda$ and $\mathbf{u}_\psi$ will have four different directions, as shown in Figs. B.1b – B.1e. If the vectors of $\mathbf{a}_{mr}$, $\mathbf{u}_\Lambda$ and $\mathbf{u}_\psi$ are as shown in Fig. B.1b, we can see $\left(\mathbf{a}_{mr}^{\mathrm{T}} \mathbf{u}_\Lambda\right) < 0$ and $\left(\mathbf{a}_{mr}^{\mathrm{T}} \mathbf{u}_\psi\right) < 0$, then the vectors of $K_3 K_\Lambda \left(\mathbf{a}_{mr}^{\mathrm{T}} \mathbf{u}_\Lambda\right)\mathbf{u}_\Lambda$ and $K_5 K_\psi \left(\mathbf{a}_{mr}^{\mathrm{T}} \mathbf{u}_\psi\right)\mathbf{u}_\psi$ must be along the negative direction of $\mathbf{u}_\Lambda$ and $\mathbf{u}_\psi$. So the two vectors are opposite to the direction shown in Fig. B.1a; the right triangle cannot be formed. If the vectors of $\mathbf{a}_{mr}$, $\mathbf{u}_\Lambda$ and $\mathbf{u}_\psi$ is as in Fig. B.1c, we can see $\left(\mathbf{a}_{mr}^{\mathrm{T}} \mathbf{u}_\Lambda\right) > 0$ and $\left(\mathbf{a}_{mr}^{\mathrm{T}} \mathbf{u}_\psi\right) > 0$, $K_3 K_\Lambda \left(\mathbf{a}_{mr}^{\mathrm{T}} \mathbf{u}_\Lambda\right)\mathbf{u}_\Lambda$ and $K_5 K_\psi \left(\mathbf{a}_{mr}^{\mathrm{T}} \mathbf{u}_\psi\right)\mathbf{u}_\psi$ must be along the positive direction of $\mathbf{u}_\Lambda$ and $\mathbf{u}_\psi$ in this figure. So the two vectors are also opposite to the direction shown in Fig. B.1a. For Fig. B.1d and B.1e, the same conclusions can be drawn. So, $K_1 \mathbf{a}_{mr} + K_3 K_\Lambda \left(\mathbf{a}_{mr}^{\mathrm{T}} \mathbf{u}_\Lambda\right)\mathbf{u}_\Lambda + K_5 K_\psi \left(\mathbf{a}_{mr}^{\mathrm{T}} \mathbf{u}_\psi\right)\mathbf{u}_\psi \neq \mathbf{0}$. Then

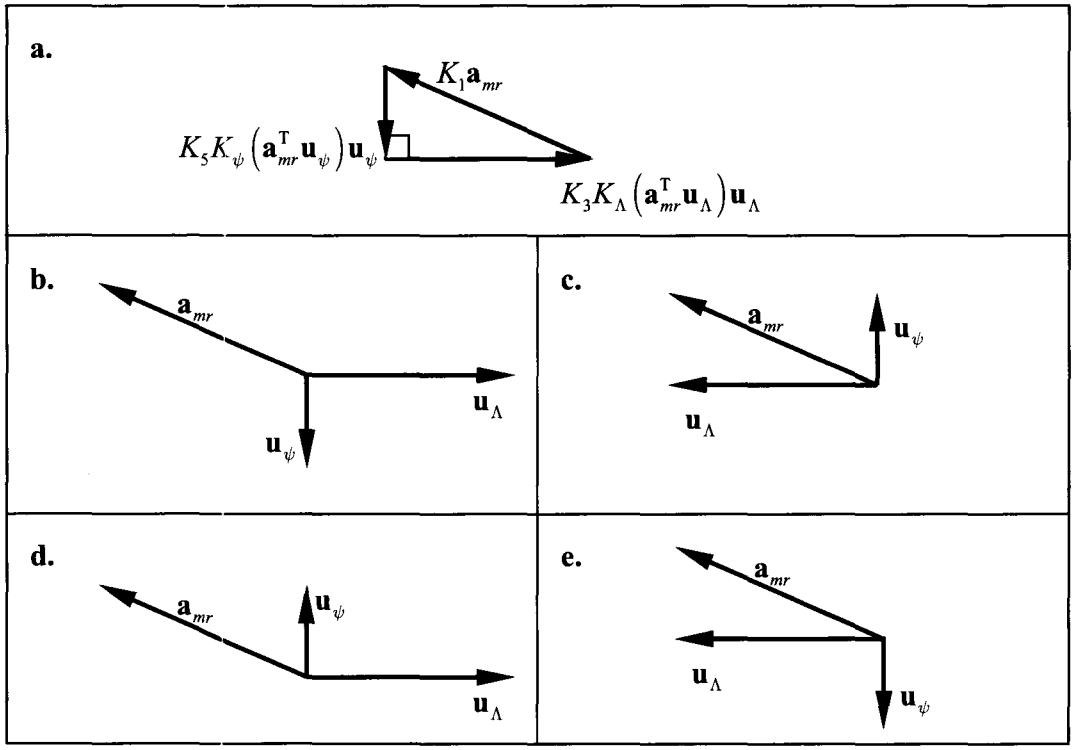


Fig. B.1. The conditions of $K_1\mathbf{a}_{mr}+K_3K_\Lambda\left(\mathbf{a}_{mr}^{\mathrm{T}}\mathbf{u}_\Lambda\right)\mathbf{u}_\Lambda+K_5K_\psi\left(\mathbf{a}_{mr}^{\mathrm{T}}\mathbf{u}_\psi\right)\mathbf{u}_\psi\neq\mathbf{0}$

$$\left\|\mathbf{v}_{L,\Delta t}\right\|_2\neq 0 \tag{B.2.19}$$

With (B.2.19), that $\mathbf{F}_L$ will drive the robot such that $\left\|\mathbf{v}_L\right\|_2\neq 0$ when $\mathbf{v}_L\to 0$, $K_L\neq 0$ is now proven. If $\dot{\mathbf{E}}=\mathbf{0}$, $\dot{\Lambda}=0$ and $\dot{\psi}^*=0$, this means that the robot is stopped and the obstacle is stationary or stopped. Therefore, $\dot{d}=0$ and $\dot{\Phi}=0$. In this case, we can see that $\Omega=0$ since every element in (B.14) is a function of either $\dot{\mathbf{E}}$, $\dot{\Lambda}^*$, $\dot{\psi}^*$, $\dot{d}$ or $\dot{\Phi}$. Then $K_L=0$ and $\mathbf{F}_L=\mathbf{0}$. At this condition, if $\mathbf{F}_v=\mathbf{F}_a+\mathbf{F}_\Lambda+\mathbf{F}_\psi=\mathbf{0}$ is also true, then the robot is stopped and will not be restarted. Since $\dot{\mathbf{E}}$, $\dot{\Lambda}^*$, $\dot{\psi}^*$ are all zeros, according to the force functions of $\mathbf{F}_a$, $\mathbf{F}_\Lambda$ and $\mathbf{F}_\psi$, $K_1\mathbf{E}+K_3\Lambda\mathbf{u}_\Lambda+K_5\psi\mathbf{u}_\psi=\mathbf{0}$. This rare situation is an

equilibrium point. From the above analysis, we can conclude $\dot{V}_2$ is negative semi-definite, and that $\dot{V}_2 = 0$ will rarely occur.

B.3 The continuity of the new VFF for multiple obstacles

If the robot is sharing its work area with N obstacles (which could include other robots), the force field will be:

$$\mathbf{F}_v = \begin{cases} \mathbf{F}_a & \text{if } \mathbf{P}_r \in \mathbb{C}_1 \\ \text{undefined} & \text{if } \mathbf{P}_r \in \mathbb{C}_{3,n}\ i \in [1,...,N] \\ \mathbf{F}_a + \sum_{i=1}^{N} c_i \left(\mathbf{F}_{\Lambda,i} + \mathbf{F}_{\psi,i}\right) + \mathbf{F}_{L,N} & \text{otherwise} \end{cases} \tag{B.3.1}$$

where $c_i = \begin{cases} 1 & \text{if } \mathbf{P}_r \in \mathbb{C}_{2,i} \\ 0 & \text{otherwise} \end{cases}$; $\mathbf{F}_{\Lambda,i}$ and $\mathbf{F}_{\psi,i}$ are the repulsive and detour forces for the i^{th} obstacle, respectively. $\mathbf{F}_{L,N} = \dfrac{\Omega_N \mathbf{u}_{L,N}}{\left\|\mathbf{v}_{L,N}\right\|}$, $\mathbf{v}_{L,N} = K_1 \dot{\mathbf{E}} + \sum_{i=1}^{N} c_i \left(K_{3,i} \dot{\Lambda}_i^* \mathbf{u}_{\Lambda,i} + K_{5,i} \dot{\psi}_i^* \mathbf{u}_{\psi,i}\right)$ and

$$\begin{aligned} \Omega_N = & \sum_{i=1}^{N} c_i K_{1,i} \dot{\mathbf{E}}^{\mathrm{T}} \left(K_{3,i} \Lambda_i \mathbf{u}_{\Lambda,i} + K_{5,i} \psi_i \mathbf{u}_{\psi,i}\right) - \sum_{i=1}^{N} c_i K_{3,i}^2 \frac{2\dot{d}_i (r_{2,i} - d_i)}{d_i - r_{3,i}} \Lambda_i \\ & + \tfrac{1}{2} \sum_{i=1}^{N} c_i M_r K_{3,i} \dot{K}_{\Lambda,i} \dot{d}_i^2 + \sum_{i=1}^{N} c_i K_{5,i} \dot{\psi}_i^* \left(\mathbf{a}_{o,i}^{*\mathrm{T}} \mathbf{u}_{\psi,i} - K_{1,i} \mathbf{E}_i^{\mathrm{T}} \mathbf{u}_{\psi,i}\right) \\ & - \sum_{i=1}^{N'} c_i K_{3,i} \dot{\Lambda}_i \left(K_{1,i} \mathbf{E}_i^{\mathrm{T}} \mathbf{u}_{\Lambda,i} - \mathbf{a}_{o,i}^{\mathrm{T}} \mathbf{u}_{\Lambda,i}\right) + \tfrac{1}{2} \sum_{i=1}^{N} c_i M_r K_{5,i} (r_{2,i} - d_i)^2 \dot{d}_i \dot{\Phi}_i^2 \\ & - \sum_{i=1}^{N'} c_i M_r K_{5,i} (r_{2,i} - d_i) d_i \dot{d}_i \dot{\Phi}_i^2 - \sum_{i=1}^{N} c_i K_{5,i}^2 \left(r_{2,i} - d_i\right) \dot{d}_i \Phi_i^2 \ . \end{aligned} \tag{B.3.2}$$

With (B.3.1), the force field is piecewise. The VFF for multiply obstacles (B.3.1) is also continuous at boundaries of every $\mathbb{C}_2$. The proof is as follows:

We already knew that when $N = 1$, the new VFF is continuous. Let us examine

the continuity when $N = k+1 \ \forall k > 0$. At the boundary of the active region between $\mathbb{C}_1$ and $\mathbb{C}_{2,k+1}$ of the $k+1^{\text{th}}$ obstacle, we have:

$$\mathbf{F}_{v,\mathbf{P}_r \in \mathbb{C}_1} = K_1 \mathbf{E} + K_2 \dot{\mathbf{E}} = \mathbf{F}_a \tag{B.3.3}$$

$$\lim_{d_{k+1} \to r_{2,k+1}} \Lambda_{k+1} = \lim_{d_{k+1} \to r_{2,k+1}} \frac{\left(r_{2,k+1} - d_{k+1}\right)^2}{d_{k+1} - r_{3,k+1}} = 0 \tag{B.3.4a}$$

$$\lim_{d_{k-1} \to r_{2,k+1}} \dot{\Lambda}^*_{k+1} = \lim_{d_{k+1} \to r_{2,k+1}} -\frac{\dot{d}_{k+1}(r_{2,k+1} - d_{k+1})^2}{(d_{k+1} - r_{3,k+1})^2} = 0 \tag{B.3.4b}$$

$$\lim_{d_{k+1} \to r_{2,k+1}} \psi_{k+1} = \lim_{d_{k+1} \to r_{2,k-1}} \left(r_{2,k+1} - d_{k+1}\right)^2 \Phi_{k+1} = 0 \tag{B.3.4c}$$

$$\lim_{d_{k+1} \to r_{2,k+1}} \dot{\psi}^*_{k+1} = \lim_{d_{k+1} \to r_{2,k-1}} \left(r_{2,k+1} - d_{k+1}\right)^2 \dot{\Phi}_{k+1} = 0 \tag{B.3.4d}$$

Substituting (B.3.4a) – (B.3.4d) into (B.3.2), we have:

$$\lim_{d_k \to r_{2,k+1}} \mathbf{F}_{L,k+1} = \mathbf{F}_{L,k} \tag{B.3.4e}$$

With (B.3.4a) – (B.3.4e), we have:

$$\lim_{d_{k+1} \to r_{2,k+1}} \mathbf{F}_v = \lim_{d_{k+1} \to r_{2,k+1}} \mathbf{F}_a + \sum_{i=1}^{k} c_i \left(\mathbf{F}_{\Lambda,i} + \mathbf{F}_{\psi,i}\right) + \mathbf{F}_{L,k} \tag{B.3.5}$$

If the boundary is only between $\mathbb{C}_1$ and $\mathbb{C}_{2,k+1}$, $c_i = 0$ for $i = 1, 2, 3, \ldots, k$ and $\mathbf{F}_{L,k} = 0$:

$$\lim_{d_{k+1} \to r_{2,k+1}} \mathbf{F}_v = \lim_{d_{k+1} \to r_{2,k+1}} \mathbf{F}_a + \sum_{i=1}^{k} c_i \left(\mathbf{F}_{\Lambda,i} + \mathbf{F}_{\psi,i}\right) + \mathbf{F}_{L,k} = \mathbf{F}_a \tag{B.3.6a}$$

So the new VFF is continuous at this boundary. If the boundary is also the boundary between $\mathbb{C}_1$ and $\mathbb{C}_{2,j}$ for $j = 1, 2, \ldots, k$ (*i.e.* the boundaries of active regions of obstacles have overlaps and intersections), since the VFF is also continuous when $N = k$, we must

have $\sum_{i=1}^{k} c_i\left(\mathbf{F}_{\Lambda,i}+\mathbf{F}_{\psi,i}\right)+\mathbf{F}_{L,k}=\mathbf{0}$. Thus:

$$\lim_{d_j\to r_{2,j}} \mathbf{F}_{V,\mathbf{P}_r\in\mathbb{C}_{2,j}} = \mathbf{F}_{\mathbf{a}} = \mathbf{F}_{V,\mathbf{P}_r\in\mathbb{C}_1} \tag{B.3.6B}$$

The VFF is also continuous at this boundary.

For the boundary between $\mathbb{C}_{2,k+1}$ and $\mathbb{C}_{2,j}$ for j = 1, 2, 3, …, k, let us first re-organize the subscript sequence to make $j = k$ for simplicity. We have:

$$\mathbf{F}_{V,\mathbf{P}_r\in\mathbb{C}_{2,k}} = \mathbf{F}_a + \sum_{i=1}^{k} c_i\left(\mathbf{F}_{\Lambda,i}+\mathbf{F}_{\psi,i}\right)+\mathbf{F}_{L,k} \tag{B.3.7}$$

$$\lim_{d_{k+1}\to r_{2,k+1}} \mathbf{F}_v = \lim_{d_{k+1}\to r_{2,k+1}} \mathbf{F}_a + \sum_{i=1}^{k+1} c_i\left(\mathbf{F}_{\Lambda,i}+\mathbf{F}_{\psi,i}\right)+\mathbf{F}_{L,k+1} \tag{B.3.8}$$

Substituting (B.3.4a) – (B.3.4d) into (B.3.8),

$$\lim_{d_{k+1}\to r_{2,k+1}} \mathbf{F}_v = \lim_{d_{k+1}\to r_{2,k+1}} \mathbf{F}_a + \sum_{i=1}^{k} c_i\left(\mathbf{F}_{\Lambda,i}+\mathbf{F}_{\psi,i}\right)+\mathbf{F}_{L,k} \tag{B.3.9}$$

So

$$\mathbf{F}_{V,\mathbf{P}_r\in\mathbb{C}_{2,n}} = \mathbf{F}_a + \sum_{i=1}^{n-1} c_i\left(\mathbf{F}_{\Lambda,i}+\mathbf{F}_{\psi,i}\right)+\mathbf{F}_{L,n-1} = \mathbf{F}_{V,\mathbf{P}_r\in\mathbb{C}_{2,n-1}} \tag{B.3.10}$$

By recursive application of (B.3.3) – (B.3.10), starting from N = 1, the continuity of the new VFF with multiple obstacles is proven.

B.4 The Lyapunov stability analysis for multiple obstacles

After the proof of the continuity of the new VFF with multiple obstacles, the stability analysis for multiple obstacles will be performed in this section. The proof by Lyapunov's second method used in Section B.2 can be expanded to multiple obstacles. If the robot is in the active regions of N obstacles, the VFF state is:

$$\left[\mathbf{E}^{\mathrm{T}},\ \dot{\mathbf{E}}^{\mathrm{T}},\ \Lambda_1,\ \dot{\Lambda}_1,\ \psi_1,\ \dot{\psi}_1,\ \ldots\ ,\Lambda_N\ \ \dot{\Lambda}_N\ \ \psi_N\ \ \dot{\psi}_N\right]^{\mathrm{T}}. \quad \text{(B.4.1)}$$

Similar to (B.1.4), the Lyapunov function candidate for the VFF with N obstacles can be built. An example for $N = 2$ is presented as follows:

Using subscripts $_1$ and $_2$ to denote the states of Obstacles 1 and 2, respectively, the state vectors are:

$$\text{If } P_r \in \mathbb{C}_1 : \mathbf{X}_1 = \left[\mathbf{E}^{\mathrm{T}} \quad \dot{\mathbf{E}}^{\mathrm{T}}\right]^{\mathrm{T}} \quad \text{(B.4.2a)}$$

$$\text{If } P_r \in \mathbb{C}_{2,1} \ \&\ P_r \notin \mathbb{C}_{2,2} : \mathbf{X}_2 = \left[\mathbf{E}^{\mathrm{T}} \quad \dot{\mathbf{E}}^{\mathrm{T}} \quad \Lambda_1 \quad \dot{\Lambda}_1^* \quad \psi_1 \quad \dot{\psi}_1^*\right]^{\mathrm{T}} \quad \text{(B.4.2b)}$$

$$\text{If } P_r \notin \mathbb{C}_{2,1} \ \&\ P_r \in \mathbb{C}_{2,2} : \mathbf{X}_3 = \left[\mathbf{E}^{\mathrm{T}} \quad \dot{\mathbf{E}}^{\mathrm{T}} \quad \Lambda_2 \quad \dot{\Lambda}_2^* \quad \psi_2 \quad \dot{\psi}_2^*\right]^{\mathrm{T}} \quad \text{(B.4.2c)}$$

$$\text{If } P_r \in \mathbb{C}_{2,1} \ \&\ P_r \in \mathbb{C}_{2,2} : \mathbf{X}_4 = \left[\mathbf{E}^{\mathrm{T}} \quad \dot{\mathbf{E}}^{\mathrm{T}} \quad \Lambda_1 \quad \dot{\Lambda}_1^* \quad \psi_1 \quad \dot{\psi}_1^* \quad \Lambda_2 \quad \dot{\Lambda}_2^* \quad \psi_2 \quad \dot{\psi}_2^*\right]^{\mathrm{T}} \quad \text{(B.4.2d)}$$

Similar to the analysis for a single obstacle from (B.1.1) to (B.1.4), we will have:

$$V = \begin{cases} V_1 = \frac{1}{2}\mathbf{X}_1^{\mathrm{T}}\mathbf{Q}_1\mathbf{X}_1, & \text{if } P_r \in \mathbb{C}_1 \\ V_2 = \frac{1}{2}\mathbf{X}_2^{\mathrm{T}}\mathbf{Q}_2\mathbf{X}_2, & \text{if } P_r \in \mathbb{C}_{2,1} \ \&\ P_r \notin \mathbb{C}_{2,2} \\ V_3 = \frac{1}{2}\mathbf{X}_3^{\mathrm{T}}\mathbf{Q}_3\mathbf{X}_3, & \text{if } P_r \notin \mathbb{C}_{2,1} \ \&\ P_r \in \mathbb{C}_{2,2} \\ V_4 = \frac{1}{2}\mathbf{X}_4^{\mathrm{T}}\mathbf{Q}_4\mathbf{X}_4, & \text{if } P_r \in \mathbb{C}_{2,1} \ \&\ P_r \in \mathbb{C}_{2,2} \end{cases} \quad \text{(B.4.3)}$$

where $\mathbf{Q}_1 = \left[K_1^2 \quad K_1^2 \quad M_r K_1 \quad M_r K_1\right] \cdot \mathbf{I}_4$;

$\mathbf{Q}_2 = \left[K_1^2 \quad K_1^2 \quad M_r K_1 \quad M_r K_1 \quad K_{3,1}^2 \quad h_{\Lambda,1} \quad K_{5,1}^2 \quad h_{\psi,1}\right] \cdot \mathbf{I}_8$;

$\mathbf{Q}_3 = \left[K_1^2 \quad K_1^2 \quad M_r K_1 \quad M_r K_1 \quad K_{3,2}^2 \quad h_{\Lambda,2} \quad K_{5,2}^2 \quad h_{\psi,2}\right] \cdot \mathbf{I}_8$; and

$\mathbf{Q}_4 = \left[K_1^2 \quad K^2 \quad M_r K_1 \quad M_r K_1 \quad K_{3,1}^2 \quad h_{\Lambda,1} \quad K_{5,1}^2 \quad h_{\psi,1} \quad K_{3,2}^2 \quad h_{\Lambda,2} \quad K_{5,2}^2 \quad h_{\psi,2}\right] \cdot \mathbf{I}_8$

for $h_{\Lambda,1} = M_r K_{3,1} \dfrac{\left(d - r_{3,1}\right)^2}{\left(r_{2,1} - d_1\right)^2}$, $h_{\psi,1} = M_r K_{5,1} \dfrac{d_1}{\left(r_{2,1} - d_1\right)^2}$, $h_{\Lambda,2} = M_r K_{3,2} \dfrac{\left(d_2 - r_{3,2}\right)^2}{\left(r_{2,2} - d_2\right)^2}$ and $h_{\psi,2} = M_r K_{5,2} \dfrac{d_2}{\left(r_{2,2} - d_2\right)^2}$. Note the piecewise Lyapunov function candidate (B.4.3) is also continuous. It is clear since $\Lambda_1, \dot{\Lambda}_1^*, \psi_1, \dot{\psi}_1^*, \Lambda_2, \dot{\Lambda}_2^*, \psi_2$, and $\dot{\psi}_2^*$ are all equal to zero at the corresponding boundaries between $\mathbb{C}_1$, $\mathbb{C}_{2,1}$ and $\mathbb{C}_{2,2}$.

In Sections B.2, V_1 and V_2 have been analyzed. To make the following analysis clear, we will add a subscript to Ω from (B.2.7). Ω_i denotes Ω for V_i. To analyze V_3, since there is one obstacle (obstacle 2), we can solve that by following the steps for the analysis of V_2, from (B.2.2) to (B.2.14). The result is:

$$\dot{V}_3 \leq -b\left\|\mathbf{v}_{L,3}\right\|_2^3 \leq 0 \tag{B.4.4}$$

where $\mathbf{v}_{L,3} = K_1\dot{\mathbf{E}} + K_{3,2}\dot{\Lambda}_2^* + K_{5,2}\dot{\psi}_2^*$. The VFF is also stable in sense of Lyapunov when $P_r \notin \mathbb{C}_{2,1}$ & $P_r \in \mathbb{C}_{2,2}$. To achieve (B.4.4), we require:

$$\mathbf{F}_L = \begin{cases} \dfrac{\Omega_3 \mathbf{u}_{L,3}}{\left\|\mathbf{v}_{L,3}\right\|_2} & \Omega_3 > 0 \\ 0 & \Omega_3 \leq 0 \end{cases} \tag{B.4.5}$$

where $\mathbf{u}_{L,3}$ is the unit vector along the direction of $\mathbf{v}_{L,3}$ and:

$$\Omega_3 = -K_1\dot{\mathbf{E}}^{\mathrm{T}}\left(K_{3,2}\Lambda_2\mathbf{u}_{\Lambda,2} + K_{5,2}\psi_2\mathbf{u}_{\psi,2}\right) - K_{3,2}^2\frac{2\dot{d}_2(r_{2,2}-d_2)}{d_2-r_{3,2}}\Lambda_2 + \\ \tfrac{1}{2}M_r K_{3,2}\dot{K}_{\Lambda,2}\dot{d}_2^2 + K_5\dot{\psi}_2^*\left(\mathbf{a}_{o,2}^{*\mathrm{T}}\mathbf{u}_{\psi,2} - K_1\mathbf{E}^{\mathrm{T}}\mathbf{u}_{\psi,2}\right) - K_{3,2}\dot{\Lambda}_2^*\left(K_1\mathbf{E}^{\mathrm{T}}\mathbf{u}_{\Lambda,2} - \mathbf{a}_{o,2}^{\mathrm{T}}\mathbf{u}_{\Lambda,2}\right) + \\ + \tfrac{1}{2}M_r K_{5,2}(r_{2,2}-d_2)^2\dot{d}_2\dot{\Phi}_2^2 - M_r K_{5,2}(r_{2,2}-d_2)d_2\dot{d}_2\dot{\Phi}_2^2 - K_{5,2}^2\left(r_{2,2}-d_2\right)\dot{d}_2\Phi_2^2 \tag{B.4.6}$$

For V_4, we have:

$$V_4 = \tfrac{1}{2}K_1^2\mathbf{E}^{\mathrm{T}}\mathbf{E} + \tfrac{1}{2}M_r K_1\dot{\mathbf{E}}^{\mathrm{T}}\dot{\mathbf{E}} + \tfrac{1}{2}K_{3,1}^2\Lambda_1^2 - \tfrac{1}{2}M_r K_{3,1}\dot{\Lambda}_1^*\dot{d}_1 + \tfrac{1}{2}K_{5,1}^2\psi_1^2 + \\ \tfrac{1}{2}M_r K_{5,1}d_1\dot{\psi}_1^{*2}\dot{\Phi}_1 + \tfrac{1}{2}K_{3,2}^2\Lambda_2^2 - \tfrac{1}{2}M_r K_{3,2}\dot{\Lambda}_2^*\dot{d}_2 + \tfrac{1}{2}K_{5,2}^2\psi_2^2 + \tfrac{1}{2}M_r K_{5,2}d_2\dot{\psi}_2^{*2}\dot{\Phi}_2 \tag{B.4.7}$$

Its first derivative is

$$\dot{V}_4 = K_1^2\mathbf{E}^{\mathrm{T}}\dot{\mathbf{E}} + K_1\dot{\mathbf{E}}^{\mathrm{T}}\left(-K_1\mathbf{E} - K_2\dot{\mathbf{E}}\right) + \\ K_1\dot{\mathbf{E}}^{\mathrm{T}}\left(-K_{4,1}\dot{\Lambda}_1^*\mathbf{u}_{\Lambda,1} - K_{6,1}\dot{\psi}_1^*\mathbf{u}_{\psi,1} - K_{4,2}\dot{\Lambda}_2^*\mathbf{u}_{\Lambda,2} - K_{6,2}\dot{\psi}_2^*\mathbf{u}_{\psi,2}\right) + \\ K_{3,1}^2\Lambda_1\dot{\Lambda}_1^* - K_{3,1}\dot{\Lambda}_1^*\left(K_{2,1}\dot{\mathbf{E}}^{\mathrm{T}}\mathbf{u}_{\Lambda} + K_{3,1}\Lambda_1 + K_{4,1}\dot{\Lambda}_1^*\right) + \\ K_{5,1}^2\psi_1\dot{\psi}_1^* + K_{5,1}\dot{\psi}_1^*\left(-K_{2,1}\dot{\mathbf{E}}^{\mathrm{T}}\mathbf{u}_{\psi} - K_{5,1}\psi_1 - K_{6,1}\dot{\psi}_1^*\right) + \\ K_{3,2}^2\Lambda_2\dot{\Lambda}_2^* - K_{3,2}\dot{\Lambda}_2^*\left(K_{2,2}\dot{\mathbf{E}}^{\mathrm{T}}\mathbf{u}_{\Lambda} + K_{3,2}\Lambda_1 + K_{4,2}\dot{\Lambda}_1^*\right) + \\ K_{5,2}^2\psi_2\dot{\psi}_2^* + K_{5,2}\dot{\psi}_2^*\left(-K_{2,2}\dot{\mathbf{E}}^{\mathrm{T}}\mathbf{u}_{\psi} - K_{5,2}\psi_2 - K_{6,2}\dot{\psi}_2^*\right) - \\ \left(K_1\dot{\mathbf{E}} + K_{3,1}\dot{\Lambda}_1^*\mathbf{u}_{\Lambda,1} + K_{5,1}\dot{\psi}_1^*\mathbf{u}_{\psi,1} + K_{3,2}\dot{\Lambda}_2^*\mathbf{u}_{\Lambda,2} + K_{5,2}\dot{\psi}_2^*\mathbf{u}_{\psi,2}\right)^{\mathrm{T}}\mathbf{F}_L + \Omega_4 \tag{B.4.8}$$

where $\Omega_4 = \Omega_2 + \Omega_3$. Let us set:

$$\mathbf{v}_{L,4} = K_1\dot{\mathbf{E}} + K_{3,1}\dot{\Lambda}_1^*\mathbf{u}_{\Lambda,1} + K_{5,1}\dot{\psi}_1^*\mathbf{u}_{\psi,1} + K_{3,2}\dot{\Lambda}_2^*\mathbf{u}_{\Lambda,2} + K_{5,2}\dot{\psi}_2^*\mathbf{u}_{\psi,2} \tag{B.4.9}$$

Eq. (B.4.8) can be rewritten as:

$$\dot{V}_4 = -b\left\|\mathbf{v}_{L,4}\right\|_2^2 + \mathbf{v}_{L,4}^{\mathrm{T}}\mathbf{F}_L + \Omega_4 \tag{B.4.10}$$

If we have:

$$\mathbf{F}_L = \begin{cases} \dfrac{\Omega_4 \mathbf{u}_{L,4}}{\left\|\mathbf{v}_{L,4}\right\|_4} & \Omega_4 > 0 \\ 0 & \Omega_4 \le 0 \end{cases} \tag{B.4.11}$$

where $\mathbf{u}_{L,4}$ is the unit vector along the direction of $\mathbf{v}_{L,4}$, the first derivation of V_4 will be:

$$\dot{V}_4 \le -b\left\|\mathbf{v}_{L,4}\right\|_2^2 \tag{B.4.12}$$

The VFF is also stable in sense of Lyapunov when $P_r \in \mathbb{C}_{2,1}$ or $P_r \in \mathbb{C}_{2,2}$. With the above analysis, we can conclude that the VFF is asymptotically stable at the origin when $\mathbf{P}_r \in \mathbb{C}_1$ and stable in sense of Lyapunov when $\mathbf{P}_r \in \mathbb{C}_{2,1} \cup \mathbb{C}_{2,2}$.

Appendix C

VFF gain sensitivity study

C.1 Introduction

The performance of a VFF-based navigation system is influenced by the gains of the VFF. It is clear that a larger K_1 tends to produce a smaller T_{arrive} and a larger K_3 tends to increase the distance between the robot the obstacle(s). In Section 5.6, the navigation performance criterion, H, was presented. In this appendix, we will study the influence of the VFF gains on the H values. We will first use a heuristic optimization method to find the VFF gains that maximize H for the three VFF-based algorithms used in Chapters 6 (specially the new VFF, the VFF of Masoud 2007, and the VFF of Ge & Cui 2002). Then, the sensitivity of the optimal solution to changes in the VFF gains will be studied numerically.

C.2 Defining the VFF gains to maximize *H*

The H values are related to the obstacle configurations. Since the obstacles configurations are extremely diverse, it is impossible to consider all configurations. In this section, four obstacle configurations, Configuration 1: avoiding a stationary obstacle, Configuration 2: avoiding a walking human, Configuration 3: avoiding two stationary obstacles, and Configuration 4: avoiding a walking human and two stationary obstacles, as shown in Figs. 6.3a, 6.3b, 6.3c and 6.3d, will be utilized. The first three configurations are fundamental for navigation systems in manufacturing and service applications. More

complex configurations, such as Configuration 4, can be produced by combining the three configurations. We will optimize the gain set (K_1, K_3, K_5 and b) used in the new VFF-based algorithm for those three fundamental configurations. The optimized gains for the other two existing algorithms will be also obtained for fairly comparing the two algorithms with the new algorithm. Since the collision may be caused by those two algorithms in Configuration 2, the optimizations for those two algorithms use Configurations 1, 2 and 4. To mitigate the local minima problem in the optimization, using the new VFF as an example, the following heuristic optimization method was used:

1) Select a large range for each gain to initialize the optimization. We selected $K_1 \in [0, 50]$, $K_3 \in [0, 200]$, $K_5 \in [0, 500]$ and $b \in [0, 3]$

2) Split the ranges of the four gains into a four-dimensional grid using a set of fixed increments. We used $\Delta K_1 = 5$, $\Delta K_3 = 10$, $\Delta K_5 = 25$ and $\Delta b = 0.2$.

3) Simulations were performed to obtain the values of H_{mean} for every point of the grid (for the new VFF-based algorithm, H_{mean} is the mean value of the H values of Configurations 1, 2 and 3; and for the two conventional VFF-based algorithms, it is the mean value of the H values of Configurations 1, 2 and 4).

4) The values of H_{mean} for all grid points were sorted in descending order. The new gain ranges were reduced to include the points with the 500 largest H_{mean} values. The values of the increments were also decreased to build a new grid for the new ranges of the gains. To limit the computation time, the values of the increments were selected to make the grid have $11 \times 21 \times 21 \times 16$ points.

5) Performed the simulations needed to obtain the values of H_{mean} for every point of the new grid. Find new gain ranges to include the points with the 500 largest H_{mean} values.

6) Repeat Step 5 two more times (i.e. for the new VFF, the increments then are $\Delta K_1 = 0.4, \Delta K_3 = 0.8$, $\Delta K_5 = 1.9$ and $\Delta b = 0.02$).

7) The gain sets with the 20 largest H_{mean} were selected.

8) The 20 gain sets were used as the initial gains for the *fmincon* function in Matlab to locally maximize H_{mean}.

9) The gain set with the largest H_{mean} from the 20 results from Step 8 was chosen as the set of optimized gains.

Using the above procedure, the optimization results for the three algorithms were obtained and are listed in Table C.1. Note that in the simulations, the holonomic robot and its velocity and acceleration limits from Chapter 6 were employed. The optimization procedure was programmed in Matlab and required one week to run on a standard PC.

Table C.1. The optimized gains of the new VFF and the two conventional VFFs

Source	*Attractive force gain*	*b*	*Repulsive force gain*	*Detour force gain*	H_{opt}
New VFF	4.3	0.68	20.3	240.7	0.69
Masoud 2007	5	-	24	36	0.66
Ge and Cui 2002	1.0	0.75	14.2	14.2	0.52

C.3 The sensitivity analysis for the new VFF-based algorithm

In this section, we will study the sensitivity of the optimal solution for the new VFF-based algorithm. The relative sensitivities (Tomovic and Vukobratovic 1972) can be

computed with:

$$S_{K_1} = \frac{1-H_{mean}/H_{opt}}{1-K_1/K_{1,opt}} = \frac{H_{opt}-H_{mean}}{K_{1,opt}-K_1} \times \frac{K_{1,opt}}{H_{opt}} = \frac{\Delta H}{\Delta K_1} \times \frac{K_{1,opt}}{H_{opt}} \quad \text{(C.3.1)}$$

$$S_{k_3} = \frac{1-H_{mean}/H_{opt}}{1-K_3/K_{3,opt}} = \frac{\Delta H}{\Delta K_3} \times \frac{K_{3,opt}}{H_{opt}} \quad \text{(C.3.2)}$$

$$S_{k_5} = \frac{1-H_{mean}/H_{opt}}{1-K_5/K_{5,opt}} = \frac{\Delta H}{\Delta K_5} \times \frac{K_{5,opt}}{H_{opt}} \quad \text{(C.3.3)}$$

$$S_b = \frac{1-H_{mean}/H_{opt}}{1-b/b_{opt}} = \frac{\Delta H}{\Delta b} \times \frac{b_{opt}}{H_{opt}} \quad \text{(C.3.4)}$$

where S_{K_1}, S_{K_3}, S_{K_5} and S_b are the relative sensitivity of H_{mean} to K_1, K_3 K_5 and b, respectively; $K_{1,opt} = 4.3$ N/m, $K_{3,opt} = 20.3$ N/m, $K_{5,opt} = 240$ N/m^2 and $b_{opt} = 0.68$ are the optimized gains. Due to the complexity of the new VFF, the analytical equations for the sensitivity cannot be derived. A numerical solution was used to reveal the sensitivity of the VFF gains. Each gain was seperately increased by the series of percentages:

$$[-100 \quad -50 \quad -20 \quad -10 \quad -5 \quad -2 \quad 0 \quad 2 \quad 5 \quad 10 \quad 20 \quad 50 \quad 100]\%.$$

The values of H_{mean} were obtained by simulating the three configurations with those altered gains. The T_{arrive} (unit: s), L_{energy} (unit: J) and $d_{s,min}$ (unit: m) values for those simulations are listed in Tables C.2, C.4, C.6, and C.8 for the four gains, respectively. The sensitivity results, H_1, H_2, H_3 and H_{mean} values are listed in Table C.3, C.5., C.7, and C.9. Note that H_i denotes the H value obtained with the i^{th} configuration.

From Table C.2, a larger attractive force gain, K_1, reduces T_{arrive} and that tends to increase the H value. However, it may also reduce distance $d_{s,min}$, which will make the H

value smaller. From Table C.3, the maximum sensitivity of H_{mean} to K_1 is 2%, and the average sensitivity is 1%. If $K_1 = 0$, the robot cannot reach its goal.

Table C.2. T_{arrive}, L_{energy} and $d_{s,\min}$ for Configuration 1, 2 and 3 with different K_1

%	Configuration 1			Configuration 2			Configuration 3			
	T_{arrive}	L_{energy}	$d_{s,\min,cir}$	T_{arrive}	L_{energy}	$d_{s,\min,h}$	T_{arrive}	L_{energy}	$d_{s,\min,cir}$	$d_{s,\min,rect}$
-50	6.44	3.61	0.562	7.64	5.53	0.860	6.32	4.13	0.538	0.560
-20	6.35	3.62	0.556	7.48	5.47	0.859	6.22	4.10	0.539	0.560
-10	6.33	3.63	0.554	7.45	5.45	0.859	6.20	4.10	0.539	0.559
-5	6.32	3.64	0.553	7.45	5.45	0.859	6.19	4.09	0.539	0.559
-2	6.31	3.64	0.552	7.44	5.45	0.858	6.19	4.09	0.539	0.559
0	6.31	3.64	0.552	7.44	5.45	0.858	6.19	4.09	0.539	0.559
2	6.31	3.65	0.552	7.44	5.46	0.858	6.18	4.08	0.539	0.559
5	6.30	3.65	0.551	7.44	5.46	0.858	6.18	4.03	0.539	0.559
10	6.30	3.66	0.549	7.44	5.47	0.858	6.18	4.02	0.539	0.559
20	6.29	3.66	0.535	7.45	5.48	0.857	6.17	4.00	0.539	0.559
50	6.26	3.69	0.525	7.49	5.51	0.857	6.15	4.00	0.540	0.558
100	6.24	3.75	0.518	7.63	5.62	0.855	6.13	3.90	0.540	0.558

Table C.3. Sensitivity analysis results for the attractive force gain - K_1

$\frac{\Delta K_1}{K_1}$ %	S_{K_1} for H_{mean} %	Performance criterion values			
		H_1	H_2	H_3	H_{mean}
-100	Avoid obstacle(s) but fail to reach the goal for Configurations 1, 2 and 3				
-50	1.57	0.7431	0.4689	0.7681	0.6600
-20	0.55	0.7430	0.4759	0.7743	0.6644
-10	0.47	0.7423	0.4765	0.7752	0.6647
-5	0.31	0.7410	0.4773	0.7759	0.6647
-2	0.78	0.7407	0.4773	0.7763	0.6648
0	0	0.7403	0.4774	0.7767	0.6648
2	0.78	0.7397	0.4774	0.7769	0.6647
5	0.94	0.7390	0.4773	0.7768	0.6644
10	0.94	0.7381	0.4772	0.7769	0.6641
20	1.33	0.7370	0.4765	0.7767	0.6634
50	1.54	0.7298	0.4740	0.7768	0.6602
100	1.94	0.7176	0.4655	0.7750	0.6527

The repulsive force gain K_3 has an opposite effect to K_1. A large K_3 tends to increase the distances between the robot and the obstacles larger and tends to increase T_{arrive} since the robot has to travel a longer path. Those trends are seen in Table C.4. From Table C.5, we can see that the H values for those three configurations vary only 0.4% with 150% different K_3 values. The maximum sensitivity of H to K_3 is 0.52%, and the average sensitivity is only 0.2%. If K_3 = 0, the human-robot collision happens in Configuration 2 since the velocity of the robot towards the human is required to be reduced by the repulsive force before the detour force drags the robot sideways. But for stationary obstacles (*i.e.* Configurations 1 and 3), by only using the detour force, the navigation also can be completed.

Table C.4. T_{arrive}, L_{energy} and $d_{s,\min}$ for Configuration 1, 2 and 3 with different K_3

%	Configuration 1			Configuration 2			Configuration 3			
	T_{arrive}	L_{energy}	$d_{s,\min,cir}$	T_{arrive}	L_{energy}	$d_{s,\min,h}$	T_{arrive}	L_{energy}	$d_{s,\min,cir}$	$d_{s,\min,rect}$
-50	6.30	3.64	0.550	7.41	5.41	0.854	6.18	4.08	0.540	0.559
-20	6.31	3.64	0.552	7.42	5.44	0.856	6.18	4.09	0.539	0.559
-10	6.31	3.64	0.552	7.43	5.44	0.857	6.18	4.09	0.539	0.559
-5	6.31	3.64	0.552	7.44	5.45	0.858	6.19	4.09	0.539	0.559
-2	6.31	3.64	0.552	7.44	5.45	0.858	6.19	4.09	0.539	0.559
0	6.31	3.64	0.552	7.44	5.45	0.858	6.19	4.09	0.539	0.559
2	6.31	3.64	0.552	7.44	5.45	0.858	6.20	4.09	0.539	0.559
5	6.31	3.64	0.552	7.45	5.45	0.858	6.20	4.09	0.539	0.559
10	6.31	3.64	0.552	7.45	5.46	0.859	6.20	4.10	0.539	0.559
20	6.31	3.64	0.552	7.45	5.46	0.860	6.21	4.10	0.539	0.559
50	6.31	3.65	0.553	7.47	5.48	0.862	6.21	4.10	0.539	0.559
100	6.32	3.65	0.553	7.50	5.52	0.866	6.22	4.10	0.539	0.560

Table C.5. Sensitivity analysis results for the repulsive force gain – K_3

$\frac{\Delta K_3}{K_3}$ %	S_{k_3} for H_{mean} %	Performance criterion values			
		H_1	H_2	H_3	H_{mean}
-100	Avoid obstacle(s) and reach the goal for Configurations 1 and 3 Fail on Configuration 2				
-50	0.07	0.7406	0.4776	0.7770	0.6651
-20	0.03	0.7407	0.4776	0.7769	0.6651
-10	0.05	0.7408	0.4775	0.7769	0.6651
-5	0.10	0.7409	0.4775	0.7768	0.6651
-2	0.20	0.7410	0.4774	0.7768	0.6651
0	-	0.7411	0.4774	0.7768	0.6652
2	0.52	0.7411	0.4773	0.7767	0.6650
5	0.31	0.7411	0.4772	0.7767	0.6650
10	0.26	0.7411	0.4771	0.7766	0.6649
20	0.18	0.7412	0.4769	0.7765	0.6649
50	0.18	0.7413	0.4762	0.7761	0.6645
100	0.17	0.7414	0.4752	0.7755	0.6640

The detour force drags the robot to detour around the obstacle. The robot will move sideways earlier with a larger K_5; then $d_{s,\min}$ will be increased. At the same time, as mentioned in Chapter 5, the new VFF-based algorithm with a detour force helps reduce T_{arrive} in comparison with the existing VFF-based algorithms without detour forces. However, with a larger K_5 the robot will detour farther around the obstacle(s) and a longer path is required. So a larger detour force may cause a slightly larger T_{arrive} (less than 2% larger for 150% larger K_5; see the values of T_{arrive} in Table C.6). From Table C.7, the maximum sensitivity of H_{mean} to K_5 is 1.8% and the average sensitivity of H_{mean} to K_5 is 0.7%. If $K_5 = 0$, the robot cannot avoid the walking human in Configuration 2 since it is at the collinear condition (see Table C.7). Severe path oscillations also occur in Configuration 1 and 3 when $K_5 = 0$.

Table C.6. T_{arrive}, L_{energy} and $d_{s,\min}$ for Configuration 1, 2 and 3 with different K_5

%	Configuration 1			Configuration 2			Configuration 3			
	T_{arrive}	L_{energy}	$d_{s,\min,cir}$	T_{arrive}	L_{energy}	$d_{s,\min,h}$	T_{arrive}	L_{energy}	$d_{s,\min,cir}$	$d_{s,\min,rect}$
-50	6.26	3.52	0.529	7.46	5.44	0.860	6.16	3.96	0.539	0.559
-20	6.28	3.60	0.542	7.45	5.45	0.858	6.18	4.06	0.539	0.559
-10	6.29	3.62	0.548	7.44	5.45	0.858	6.18	4.07	0.539	0.559
-5	6.30	3.63	0.550	7.44	5.45	0.858	6.19	4.08	0.539	0.559
-2	6.31	3.64	0.551	7.44	5.45	0.858	6.19	4.09	0.539	0.559
0	6.31	3.64	0.552	7.44	5.45	0.858	6.19	4.09	0.539	0.559
2	6.31	3.65	0.553	7.44	5.45	0.858	6.19	4.10	0.539	0.559
5	6.32	3.65	0.554	7.44	5.45	0.858	6.19	4.11	0.539	0.559
10	6.32	3.66	0.555	7.44	5.45	0.858	6.19	4.13	0.539	0.559
20	6.32	3.67	0.557	7.44	5.46	0.859	6.20	4.15	0.539	0.559
50	6.34	3.70	0.563	7.44	5.48	0.860	6.21	4.25	0.539	0.559
100	6.36	3.73	0.570	7.43	5.97	0.860	6.23	4.33	0.538	0.560

Table C.7. Sensitivity analysis results for the detour force Gain – K_5

$\frac{\Delta K_5}{K_5}$ %	S_{k5} for H_{mean} %	Performance criterion values			
		H_1	H_2	H_3	H_{mean}
-100	Severe path oscillation and fail to reach the goal for Configuration 1 and 3. Fail to avoid obstacle(s) in Configuration 2.				
-50	0.56	0.7277	0.4772	0.7852	0.6634
-20	0.29	0.7376	0.4773	0.7795	0.6648
-10	0.26	0.7396	0.4773	0.7781	0.6650
-5	0.52	0.7404	0.4773	0.7773	0.6650
-2	1.04	0.7407	0.4774	0.7770	0.6650
0	-	0.7411	0.4774	0.7770	0.6651
2	1.83	0.7411	0.4773	0.7764	0.6649
5	0.94	0.7413	0.4773	0.7760	0.6649
10	0.57	0.7418	0.4773	0.7753	0.6648
20	0.52	0.7425	0.4772	0.7738	0.6645
50	0.50	0.7443	0.4769	0.7695	0.6636
100	0.42	0.7462	0.4763	0.7649	0.6625

The VFF gain ratio, b, affects all three virtual forces. From Tables C.8 and C.9, its influences on the performance are less significant than K_1, K_2 and K_3. The maximum sensitivity of H_{mean} to b is 0.4%, and the average sensitivity is 0.2%. If $b = 0$, the robot is able to move close to the goal and cannot reach the goal since b is the source of damping in the new VFF.

From Tables C.2, C.4 and C.6, we can see that larger K_1, K_3 and K_5 increase L_{energy} since they cause larger virtual forces and larger control commands. From Table C.8, a larger b decreases L_{energy}.

Table C.8. T_{arrive}, L_{energy} and $d_{s,\min}$ for the simulations with different b

%	Configuration 1			Configuration 2			Configuration 3			
	T_{arrive}	L_{energy}	$d_{s,\min,cir}$	T_{arrive}	L_{energy}	$d_{s,\min,h}$	T_{arrive}	L_{energy}	$d_{s,\min,cir}$	$d_{s,\min,rect}$
-50	6.30	3.66	0.552	7.45	5.45	0.854	6.16	4.14	0.531	0.567
-20	6.30	3.66	0.552	7.44	5.45	0.856	6.17	4.11	0.535	0.562
-10	6.30	3.65	0.552	7.44	5.45	0.857	6.18	4.10	0.538	0.560
-5	6.31	3.65	0.552	7.44	5.45	0.858	6.18	4.10	0.539	0.560
-2	6.31	3.65	0.552	7.44	5.45	0.858	6.19	4.09	0.539	0.559
0	6.31	3.64	0.552	7.44	5.45	0.858	6.19	4.09	0.539	0.559
2	6.31	3.64	0.552	7.44	5.45	0.859	6.19	4.09	0.539	0.559
5	6.32	3.64	0.552	7.44	5.45	0.859	6.19	4.09	0.539	0.559
10	6.32	3.63	0.552	7.44	5.45	0.859	6.19	4.09	0.539	0.559
20	6.34	3.62	0.552	7.44	5.45	0.860	6.19	4.09	0.539	0.559
50	6.41	3.58	0.548	7.44	5.45	0.863	6.19	4.09	0.539	0.559
100	6.45	3.55	0.546	7.44	5.45	0.865	6.19	4.09	0.539	0.559

Table C.9. Sensitivity analysis results for the gain ratio – b

$\frac{\Delta b}{b}$ %	S_b for H_{mean} %	Performance criterion values			
		H_1	H_2	H_3	H_{mean}
-100	Avoid the obstacles in Configuration 1, 2 and 3. Fail to reach the goals in Configuration 1, 2 and 3.				
-50	0.17	0.7412	0.4761	0.7763	0.6645
-20	0.16	0.7412	0.4768	0.7766	0.6649
-10	0.21	0.7412	0.4770	0.7766	0.6649
-5	0.21	0.7411	0.4772	0.7767	0.6650
-2	0.40	0.7411	0.4772	0.7767	0.6650
0	-	0.7411	0.4774	0.7767	0.6651
2	0.26	0.7410	0.4774	0.7767	0.6650
5	0.10	0.7410	0.4775	0.7766	0.6650
10	0.16	0.7408	0.4775	0.7766	0.6650
20	0.08	0.7403	0.4780	0.7766	0.6649
50	0.26	0.7374	0.4787	0.7766	0.6642
100	0.25	0.7349	0.4790	0.7766	0.6635

Appendix D:

D.1 Kinematics of the holonomic mobile robot

There are several designs for the holonomic robots. Watanabe et al. (1998) developed a holonomic robot with three omnidirection wheels. The three wheels are independently driven and each wheel is located at one corner of an equilateral triangle, as shown in Fig. D.1. This design is relatively simpler to built, but special omnidirectional wheels (see Fig. D.2) are required. Ostrovskaya, Angeles and Spiteri (2000) used three ball wheels to create a holonomic robot. With their robot, six electric motors are used, three for the rolling of the wheels and three for the orientations of the wheels. This design is mechanically complex and very hard to control. Holmberg and Khatib (2000) utilized a

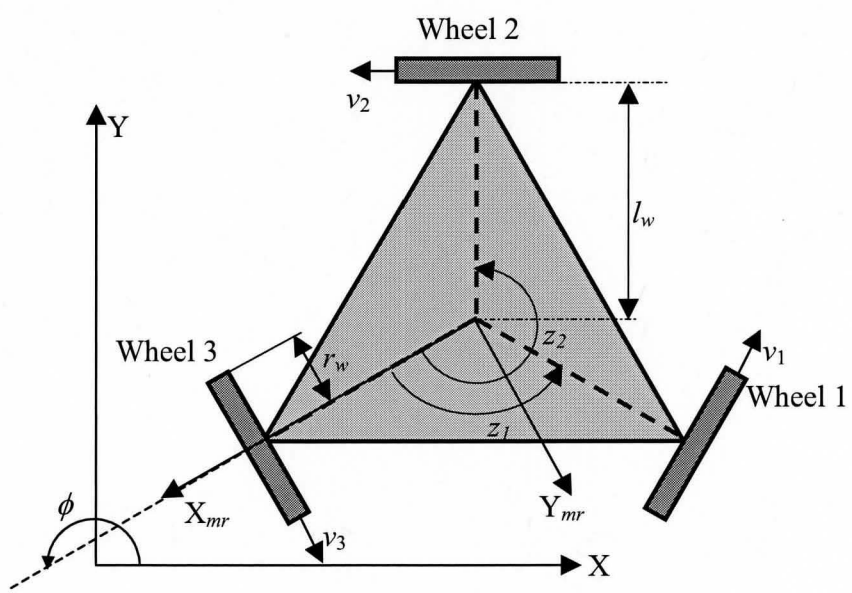


Fig. D.1. The design scheme of the holonomic robot used in Watanabe (1998) and this thesis.

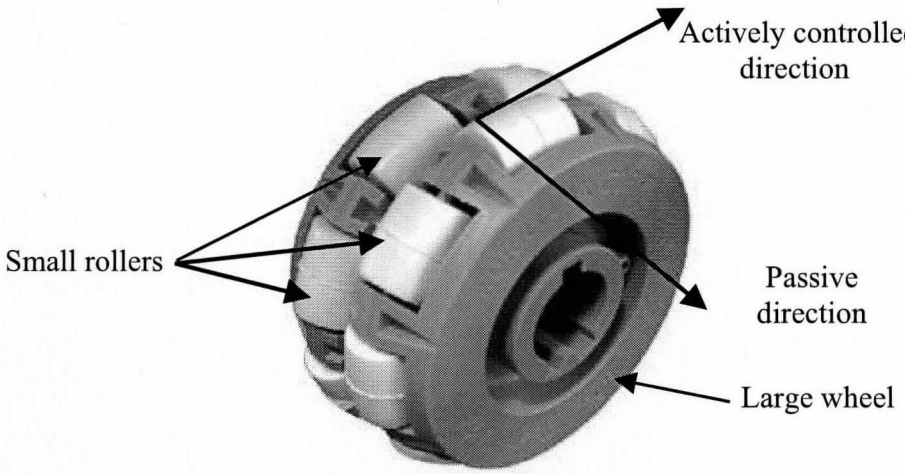


Fig. D.2. An omnidirectional wheel.

powered caster to control the orientation of their holonomic robot. This scheme provides better traction for the wheels, but suffers from the wheel singularity problem.

The robot used in this thesis adapts the design shown in Fig. D.1. Wheels 1, 2 and 3 are omnidirectional wheels. An omnidirectional wheel has two independent moving directions. The actively controlled direction is driven by a DC motor via a gearbox and a drive shaft. The passive direction allows the wheels to move laterally and is controlled by the motion of other wheels. By actively controlling the three wheels the robot can be independently moved forward/backward, left/right and rotated about the Z-axis. The robot is moving in the world coordinates X-Y. We need to first investigate the rotation of a holonomic wheel to derive the kinematic equations of the robot. The velocity of a holonomic wheel is a sum of the velocities of the larger wheel and small peripheral rollers. The rotation of the larger wheel and smaller rollers are shown in Fig. D.3. Let us define the controlled linear velocities of the wheels as $\begin{bmatrix} v_1 & v_2 & v_3 \end{bmatrix}^T =$

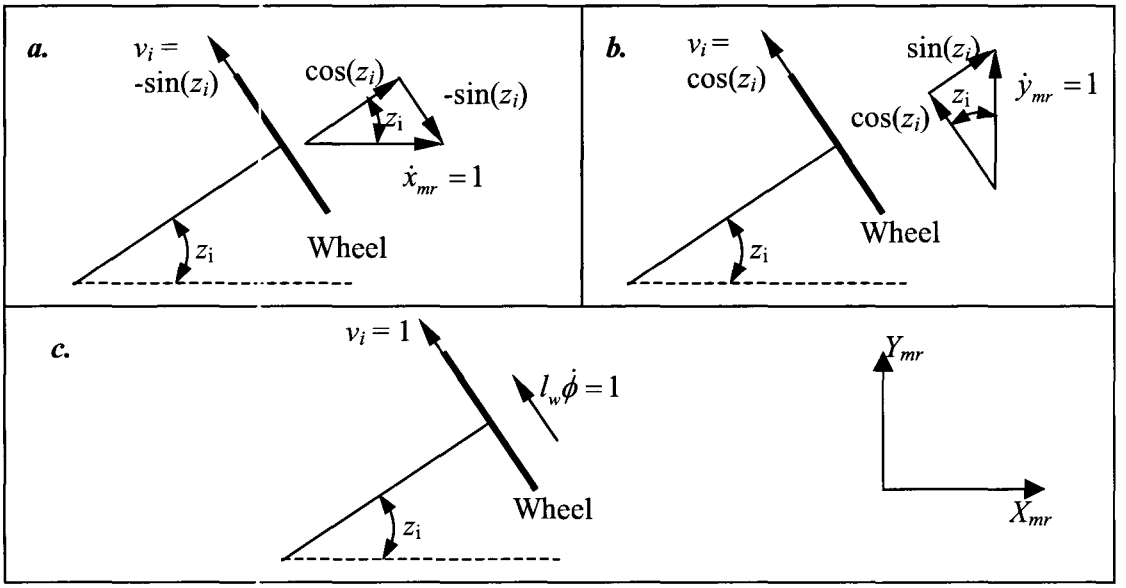


Fig. D.3. Rotation of large wheel and small rollers.

$\left[r_w\dot{\theta}_1 \quad r_w\dot{\theta}_2 \quad r_w\dot{\theta}_3\right]^T$ where r_w is the radius of the wheels; and $\dot{\theta}_1$, $\dot{\theta}_2$ and $\dot{\theta}_3$ are the angular velocities of the wheels. According to Fig. D.3a, if the robot velocity is $[\dot{x}_{mr}\ 0\ 0]^T$, we have:

$$v_i = -\sin(z_i)\dot{x}_{mr} \tag{D.1.1a}$$

where v_i is the actively controlled linear velocity of the i^{th} wheel, and z_i is the angle between the i^{th} wheel axis and X_{mr}. Similarly, if the robot velocity is $[0\ \ \dot{y}_{mr}\ \ 0]^T$, from Fig. D.3b, we have:

$$v_i = \cos(z_i)\dot{y}_{mr}. \tag{D.1.1b}$$

If the robot velocity is $[0\ \ 0\ \ l_w\dot{\phi}]^T$ where $\dot{\phi}$ is the angular velocity of the robot, from Fig. D.3c:

$$v_i = l_w\dot{\phi} \tag{D.1.1c}$$

Then, if the robot velocity is $[\dot{x}_{mr} \quad \dot{y}_{mr} \quad l_w\dot{\phi}]$, we have the actively controlled wheel velocity:

$$v_i = -\dot{x}_{mr}\sin(z_i) + \dot{y}_{mr}\cos(z_i) + l_w\dot{\phi} \tag{D.1.2}$$

For the symmetric design: $z_1 = 120°$, $z_2 = 240°$ $z_3 = 0°$. The inverse kinematics equation of the robot is

$$r_w\begin{bmatrix}\dot{\theta}_1\\ \dot{\theta}_2\\ \dot{\theta}_3\end{bmatrix} = \begin{bmatrix}-\frac{\sqrt{3}}{2} & -\frac{1}{2} & l_w\\ \frac{\sqrt{3}}{2} & -\frac{1}{2} & l_w\\ 0 & 1 & l_w\end{bmatrix}\begin{bmatrix}\dot{x}_{mr}\\ \dot{y}_{mr}\\ \dot{\phi}\end{bmatrix} \tag{D.1.3}$$

Eq. (D.1.3) is related to the robot coordinates X_{mr}-Y_{mr}. The inverse kinematics in world coordinates X-Y is:

$$r_w\begin{bmatrix}\dot{\theta}_1\\ \dot{\theta}_2\\ \dot{\theta}_3\end{bmatrix} = \begin{bmatrix}-\frac{\sqrt{3}}{2} & -\frac{1}{2} & l_w\\ \frac{\sqrt{3}}{2} & -\frac{1}{2} & l_w\\ 0 & 1 & l_w\end{bmatrix}\begin{bmatrix}\cos\phi & \sin\phi & 0\\ -\sin\phi & \cos\phi & 0\\ 0 & 0 & 1\end{bmatrix}\begin{bmatrix}\dot{x}\\ \dot{y}\\ \dot{\phi}\end{bmatrix} \tag{D.1.4}$$

D.2 Identification of the dynamic model of the holonomic robot

As in Watanabe (1998), the dynamic equation for the i^{th} wheel is defined as:

$$\ddot{\theta}_i = \frac{K_{mr,i}}{J_{mr,i}}u_{mr,i} - \frac{B_{mr,i}}{J_{mr,i}}\dot{\theta}_i - \frac{\tau_{mr,i}}{J_{mr,i}}\mathrm{sgn}(\dot{\theta}_i) \tag{D.2.1}$$

where $K_{mr,i}$ is the product of the motor torque constant, the amplifier gains (*i.e.* 1 A/V) and the gear ratio (*i.e.* 10:1); $\tau_{mr,i}$ is the static friction torque of the i^{th} motor and gear; $B_{mr,i}$ is the viscous friction coefficient of the i^{th} motor and gear; $J_{mr,i}$ is the effective inertia of the robot plus the inertia of the motor, gear and wheel; $\dot{\theta}_i$ is the angular velocity

of i^{th} wheel and $\ddot{\theta}_i$ is its acceleration. In (D.2.1), we assume that every wheel is independently driven since the wheels are non-back-drivable with worm gear sets.

Open-loop tests were performed to obtain $\frac{K_{mr,i}}{J_{mr,i}}$, $\frac{B_{mr,i}}{J_{mr,i}}$ and $\frac{\tau_{mr,i}}{J_{mr,i}}$ for different values of the input $u_{mr,i}$. During the tests, due to the usage of the worm gear set, at least two wheels need to be moving. At first, Motor 1 was given a positive signal to drive Wheel 1 forwards and Motor 2 was given a negative signal to drive Motor 2 backwards, and the robot moved along the negative direction of X_{mr}-axis, as shown in figure D.1. Next, Motor 1 was given a negative signal and Motor 3 was given a positive signal; and the movement direction of the robot was $\phi \approx 30°$. Next, Motor 3 was given a negative signal and Motor 2 was given a positive signal; and (the movement direction of the robot was $\phi \approx -60°$). The tests for driving those three motors both forwards and backwards were performed in this fashion. In every test, control commands with the identical magnitudes were sent to the two motors. The procedure used is as follows:

1. Perform open-loop tests by sending different commands (*i.e.* $u_{mr,i}$ = 0.25, 0.3, 0.35, 0.4, 0.45 and 0.5 V), send each command twice. Obtain position profiles θ_i vs. time t for those tests.

2. Let us set $\kappa_{mr,i} = \frac{B_{mr,i}}{J_{mr,i}}$. $u_{mr,i}$ is constant for these open-loop tests and we assume the static friction torque, $\tau_{mr,i}$, is also a constant value. Then $A_{mr,i} = \frac{K_{mr,i}}{J_{mr,i}} + \frac{\tau_{mr,i}}{J_{mr,i}}$ is a constant, and we can rewrite (D.2.1) as

$$\ddot{\theta}_i = A_{mr,i} - \kappa_{mr,i}\dot{\theta}_i . \tag{D.2.2}$$

3. Use the Laplace transform and inverse Laplace transform to solve (D.2.2), to obtain θ_i in time domain:

$$\theta_i = \kappa_{mr,i}\frac{A_{mr,i}t - 1 + \exp\left(-A_{mr,i}t\right)}{A_{mr,i}} \tag{D.2.3}$$

4. Use the function *lspcurvefit* in MATLAB to fit the position profiles, θ_i vs. t in the least-squares sense to find the values of $A_{mr,i}$ and $\kappa_{mr,i}$. For the fitting, the initial values of those parameters $A_{mr,i}$ and $\kappa_{mr,i}$ are calculated with the values of $K_{mr,i}$, $B_{mr,i}$ and $\tau_{mr,i}$ from the datasheet of the motor (Maxon motor, RE40-148877). In this step, we will obtain a series of $A_{mr,i}$ and $\kappa_{mr,i}$ values.

5. The mean values of $\kappa_{mr,i}$ are obtained from the series of $\kappa_{mr,i}$ values. They are listed in Table D.1.

6. To obtain the parameters $\frac{K_{mr,i}}{J_{mr,i}}$ and $\frac{\tau_{mr,i}}{J_{mr,i}}$

$$A_{mr,i} = \frac{K_{mr,i}}{J_{mr,i}}u_{mr,i} - \frac{\tau_{mr,i}}{J_{mr,i}}, \tag{D.2.4}$$

We first define $\mathbf{Y}_{mr,i}$ as the vector of $A_{mr,i}$ values from step 4; and $\mathbf{u}_{mr,i}$ = [0.25, 0.3, 0.35, 0.4, 0.45, 0.5]^T V. Then (D.2.4) can be rewritten in the format of the first order linear function $\mathbf{Y}_{mr,i} = \left(\frac{K_{mr,i}}{J_{mr,i}}\right)\mathbf{U}_{mr,i} - \left(\frac{\tau_{mr,i}}{J_{mr,i}}\right)$. We use the linear regression method as follows:

$$\frac{K_{mr,i}}{J_{mr,i}} = \frac{6\mathbf{U}_{mr,i}\mathbf{Y}_{mr,i}^{\mathrm{T}} - \sum\mathbf{U}_{mr,i}\sum\mathbf{Y}_{mr,i}}{6\mathbf{U}_{mr,i}\mathbf{U}_{mr,i}^{\mathrm{T}} - \left(\sum\mathbf{U}_{mr,i}\right)^2}; \tag{D.2.5a}$$

$$\frac{\tau_{mr,i}}{J_{mr,i}} = \frac{1}{6}\left(\sum\mathbf{Y}_{mr,i} - \frac{K_{mr,i}}{J_{mr,i}}\sum\mathbf{U}_{mr,i}\right) \tag{D.2.5b}$$

The results from (D.2.5a) and (D.2.5b) are listed in Table D.1

Table D.1. Estimated dynamic model parameters

Motor No.	Direction	$\kappa_{mr,i}$	$\frac{K_{mr,i}}{J_{mr,i}}$	$\frac{\tau_{mr,i}}{J_{mr,i}}$
1	Forward	0.241	137	28.0
	Backward	0.281	136	23.3
2	Forward	0.219	137	30.9
	Backward	0.320	134	29.4
3	Forward	0.392	152	29.1
	Backward	0.353	157	27.4

Since the differences between the forward and backward parameters for the wheels are rather small, the dynamic model for each wheel will be described with only one model and the dynamic model parameters are the average values of the forward and backward data:

$$\begin{aligned}
\ddot{\theta}_1 &= 137u_{mr,1} - 0.26\dot{\theta}_1 - 25.7\,\mathrm{sgn}\left(\dot{\theta}_1\right) \\
\ddot{\theta}_2 &= 136u_{mr,2} - 0.27\dot{\theta}_2 - 30.2\,\mathrm{sgn}\left(\dot{\theta}_2\right) \\
\ddot{\theta}_3 &= 155u_{mr,3} - 0.37\dot{\theta}_3 - 28.3\,\mathrm{sgn}\left(\dot{\theta}_3\right)
\end{aligned} \tag{D.2.6}$$

D.3 Design of the wheel controllers

The controller for each wheel is a FF plus PD plus friction compensation controller. The controller is

$$u_{mr,i} = \underbrace{\left(K_{P,mr,i}e_{mr,i} + K_{D,mr,i}\dot{e}_{mr,i}\right)}_{PD\ Feedback} + \underbrace{\frac{J_{mr,i}}{K_{mr,i}}\ddot{\theta}_{d,i}}_{\substack{Acceleration \\ Feedforward}} + \underbrace{\frac{B_{mr,i}}{K_{mr,i}}\dot{\theta}_{i,d}}_{\substack{Viscous\ Friction \\ Feedforward}} + \underbrace{\frac{\tau_{mr,i}}{K_{mr,i}}\mathrm{sgn}(\dot{\theta}_{i,d})}_{\substack{Static\ Friction \\ Feedforward}} \quad (D.3.1)$$

where $e_{mr,i} = \theta_{i,d} - \theta_i$, $\theta_{i,d}$ is the desired wheel angle; $\dot{\theta}_{i,d}$ is the desired wheel angular velocity; $\ddot{\theta}_{i,d}$ is the desired wheel angular acceleration; $K_{P,mr,i}$ is the positive proportional control gain for i^{th} wheel; and $K_{D,mr,i}$ is the positive derivative control gain. The two gains for the three wheels were obtained by manually tuning, and are listed in Table D.2. The feedforward parameters are from (D.2.6).

Table D.2. The proportional and derivative control gains; and feedforward parameters.

Gains	Wheel 1	Wheel 2	Wheel 3
$K_{P,mr,i}$ (V)	37.0	34.0	35.0
$K_{D,mr,i}$ (VS)	1.10	1.10	1.10
$J_{mr,i}/K_{mr,i}$ (VS^2)	0.0073	0.0074	0.0065
$B_{mr,i}/K_{mr,i}$ (VS)	0.0019	0.0020	0.0024
$\tau_{mr,i}/K_{mr,i}$ (V)	0.188	0.223	0.183

Using quadrature counting, the rotary encoders supplied with the DC motors have a resolution of 2000 counts/rev. The quantization errors of the incremental encoders influence the performance of the wheel controllers. Those errors cause severe oscillations in the $\dot{e}_{mr,i}$ values and significantly influence the values of $K_{D,mr,i}\dot{e}_{mr,i}$ in (D.3.1). This leads to oscillations of the robot. Luenberger observer was used to estimate the angular positions of those wheels and smoothen the $\dot{e}_{mr,i}$. Assuming the friction has been compensated, the i^{th} wheel dynamics are:

$$\begin{cases} \dot{\mathbf{x}}_{mr,i} = \mathbf{A}_{mr,i}\mathbf{x}_{mr,i} + \mathbf{B}_{mr,i}u_{mr,i} \\ \theta_i = \mathbf{C}_{mr,i}\mathbf{x}_{mr,i} \end{cases} \tag{D.3.2}$$

where $\mathbf{x}_{mr,i} = \begin{bmatrix} \theta_i & \dot{\theta} \end{bmatrix}^{\mathrm{T}}$, $\mathbf{C}_{mr,i} = \begin{bmatrix} 1 & 0 \end{bmatrix}$, $\mathbf{A}_{mr,i}$ and $\mathbf{B}_{mr,i}$ are related to (D.2.6). For the three wheels, $\mathbf{A}_{mr,1} = \begin{bmatrix} 0 & 1 \\ 0 & 0.26 \end{bmatrix}$, $\mathbf{B}_{mr,1} = \begin{bmatrix} 0 \\ 137 \end{bmatrix}$, $\mathbf{A}_{mr,2} = \begin{bmatrix} 0 & 1 \\ 0 & 0.27 \end{bmatrix}$, $\mathbf{B}_{mr,2} = \begin{bmatrix} 0 \\ 136 \end{bmatrix}$, $\mathbf{A}_{mr,3} = \begin{bmatrix} 0 & 1 \\ 0 & 0.37 \end{bmatrix}$ and $\mathbf{B}_{mr,3} = \begin{bmatrix} 0 & 155 \end{bmatrix}^{\mathrm{T}}$. The Luenberger observer is:

$$\begin{cases} \dot{\hat{\mathbf{x}}}_{mr,i} = \mathbf{A}_{mr,i}\hat{\mathbf{x}}_{mr,i} + \mathbf{B}_{mr,i}u_{mr,i} + \mathbf{L}_{mr,i}(\theta_i - \hat{\theta}_i) \\ \hat{\theta}_i = \mathbf{C}_{mr,i}\hat{\mathbf{x}}_{mr,i} \end{cases} \tag{D.3.3}$$

where $\hat{\theta}_i$ is the estimated angular position of the i^{th} wheel; $\hat{\mathbf{x}}_{mr,i} = \begin{bmatrix} \hat{\theta}_i & \dot{\hat{\theta}}_i \end{bmatrix}$ and $\mathbf{L}_i$ is the observer parameter vector that is determined by desired poles $\begin{bmatrix} p_{1,i} & p_{2,i} \end{bmatrix}$. The poles are the roots of $\mathbf{A}_{mr,i} - \mathbf{L}_{mr,i}\mathbf{C}_{mr,i}$. Faster $\begin{bmatrix} p_{1,i} & p_{2,i} \end{bmatrix}$ make $\mathbf{L}_i$ larger and $\hat{\theta}_i$ closer to the measured position, θ_i. Based on several experiments, $p_{1,i} = -10$ and $p_{2,i} = -10$ were chosen. Example experiment results are presented in Fig. D.4.

Note that (D.3.1) with $\mathbf{i} = \begin{bmatrix} i_1 & i_2 & i_3 \end{bmatrix}^{\mathrm{T}}$ are used to compute the values of L_{energy} and H for the three VFF-based algorithms (the new VFF, the VFF in Borenstein & Koren 1991 and the VFF in Ge & Cui 2002) in Chapter 6. For those simulations in Chapter 6, we assumed the tracking is perfect (*i.e.* $e_{mr,i} = 0$ and $\dot{e}_{mr,i} = 0$); so the feedback PD controllers do not contribute to the control command. From (D.3.1) and (D.2.6), we have:

$$u_{mr,1} = 0.0073\ddot{\theta}_{1,d} + 0.0019\dot{\theta}_{1,d} + 0.19\,\mathrm{sgn}\left(\dot{\theta}_{1,d}\right) \tag{D.3.2a}$$

$$u_{mr,2} = 0.0074\ddot{\theta}_{2,d} + 0.0020\dot{\theta}_{2,d} + 0.22\,\mathrm{sgn}\left(\dot{\theta}_{2,d}\right) \tag{D.3.2b}$$

$$u_{mr,3} = 0.0065\ddot{\theta}_{3,d} + 0.0024\dot{\theta}_{3,d} + 0.18\,\mathrm{sgn}\left(\dot{\theta}_{3,d}\right) \tag{D.3.2c}$$

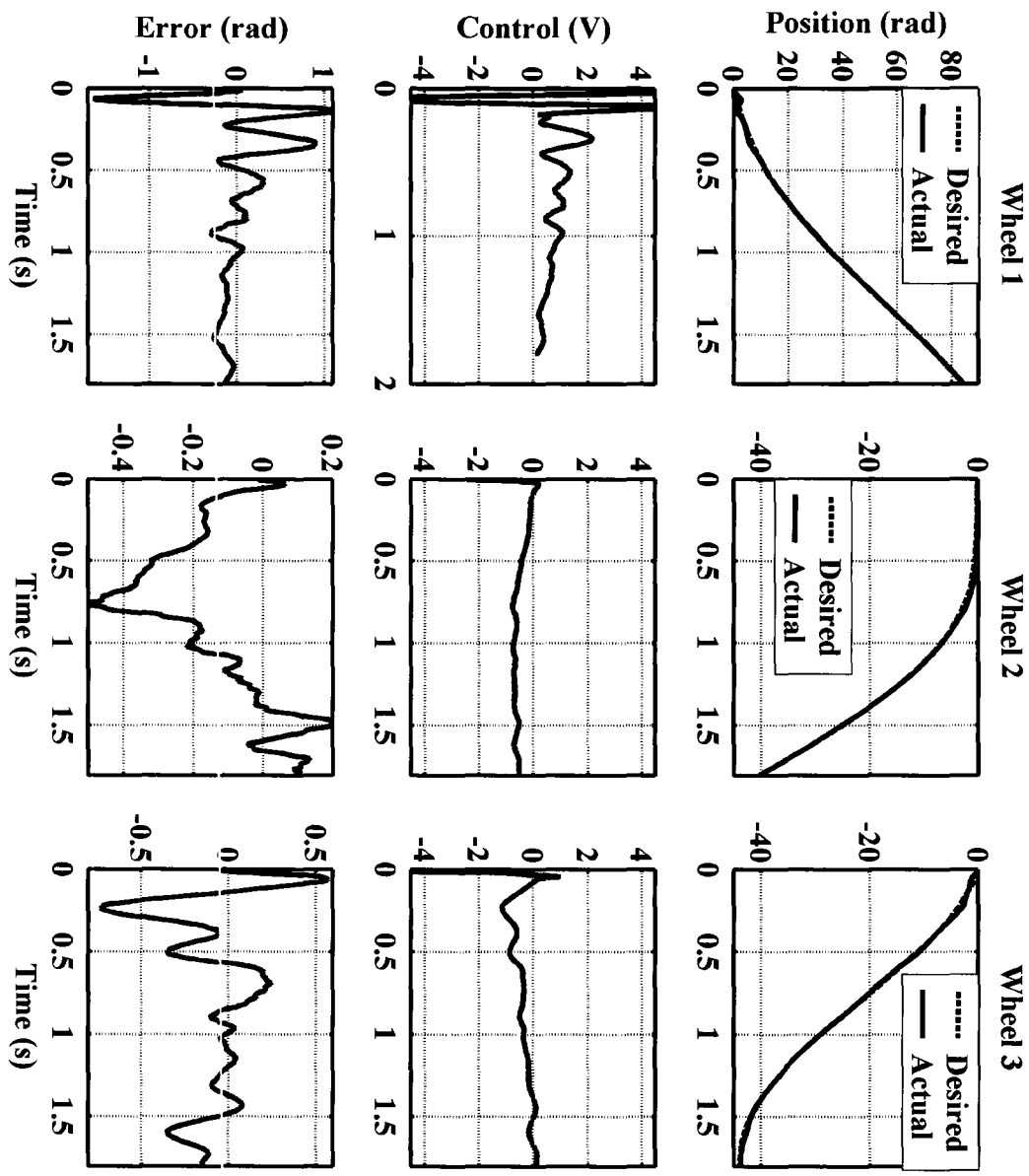


Fig. D.4. Example closed-loop control experimental results for the holonomic robot.

After the amplifier, the input current to the motors are:

$$\mathbf{i} = \begin{bmatrix} i_1 \\ i_2 \\ i_3 \end{bmatrix} = \begin{bmatrix} K_{amp} & 0 & 0 \\ 0 & K_{amp} & 0 \\ 0 & 0 & K_{amp} \end{bmatrix} \begin{bmatrix} u_{mr,1} \\ u_{mr,2} \\ u_{mr,3} \end{bmatrix} \tag{D.3.3}$$

where K_{amp} = 1 A/V is the amplifier gain in our implementation. Eq. (D.3.3) was used in (5.6.5) to calculate L_{energy}. In the experiments, the PD controllers contributed to the control commands. We directly recorded the motor current data and used them in (5.6.5).

D.4 Camera calibration

The objective of camera calibration is to find the external parameters (position and orientation relative to the world coordinate system), and the internal parameters of the camera (image centre, focal length and distortion coefficients) to build transformation functions to estimate the positions in world coordinates corresponding to the pixel positions in the images. One of the most used camera calibration techniques is the one proposed by Tsai (1986). The parameters required in Tsai's calibration method are as follows

f_{cam} : Focal length of the camera,

k_{rad} : Radial lens distortion coefficient,

C_x, C_y : Coordinates of the centre of radial lens distortion,

S_{cam} : Scale factor to account for the pixel size,

$\mathbf{R}_{cam}$: 3-by-3 rotation matrix for the transformation between the world and camera coordinates,

$\mathbf{T}_{cam} = [T_x, T_y, T_z]^T$: Translation components for the transformation between the world and camera coordinates.

In this thesis, we used this calibration method. The calibration image is shown in Fig. D.5. Twenty-six circular markers were used, and numbered from 0 to 25 in this figure. 20 of those markers have zero *Z*-axis heights and 6 of them have different non-zero Z-axis heights. The centre of each marker formed the set of calibration points.

Fig. D.5. Camera Calibration picture with 26 calibration points.

The calibration results with the 26 calibration points are as follows:

$f_{cam} = 4.76$ mm

$k_{rad} = 1.68\text{e-}002$ 1/mm^2

$S_{cam} = 0.987$

$C_x = 355.3$, $C_y = 292.1$ pixel

$$\mathbf{T}_{cam} = [2645.8,\ 357.38,\ 4250.18]^{\mathrm{T}}\ \text{mm}$$

$$\mathbf{R}_{cam} = \begin{bmatrix} -0.941126 & -0.318152 & -0.114288 \\ -0.053498 & 0.473980 & -0.878909 \\ 0.333797 & -0.821050 & -0.463095 \end{bmatrix}.$$

After the calibration, the maximum positioning error between the actual positions and the estimated positions is 0.025 m, and the mean positioning error is 0.004 m.

D.5 3D reconstruction from the pixel positions

In Fig. D.6, Tsai Camera 3D reconstruction model is presented. The pixel positions of the color patch centriods for the human and robot in the navigation experiments is denoted as $[X_{im}, Y_{im}]^{\mathrm{T}}$. The reconstruction procedure is as follows:

1) Transform from pixel positions, $[X_{im}, Y_{im}]^{\mathrm{T}}$, to the position in distorted image plane coordinates, $[X_{ds}, Y_{ds}]^{\mathrm{T}}$:

$$\begin{bmatrix} X_{ds} \\ Y_{ds} \end{bmatrix} = \begin{bmatrix} s_{pix,x}/S_{cam} & 0 \\ 0 & s_{pix,y} \end{bmatrix} \begin{bmatrix} X_{im} - C_x \\ Y_{im} - C_y \end{bmatrix} \tag{6.4.1}$$

where $s_{pix,x}$ and $s_{pix,y}$ are fixed parameters of the camera; they depend only

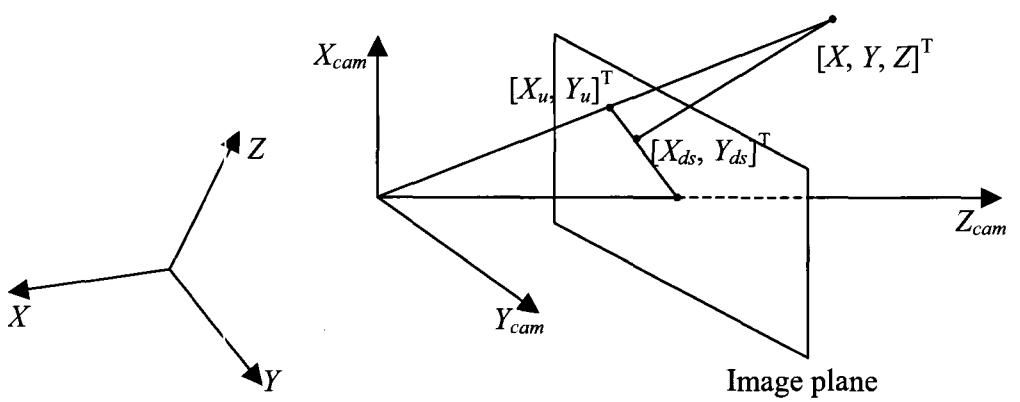


Fig. D.6. Tsai's Camera 3D reconstruction model

on the size of the imaging sensor (in mm) and the image resolution.

2) Transform from distorted coordinates, $[X_{ds}, Y_{ds}]^T$, to the undistorted image plane coordinates, $[X_u, Y_u]^T$:

$$\begin{bmatrix} X_u \\ Y_u \end{bmatrix} = \begin{bmatrix} \left(1+k_{rad}l_{cam,ds}\right)^{-1} & 0 \\ 0 & \left(1+k_{rad}l_{cam,ds}\right)^{-1} \end{bmatrix} \begin{bmatrix} X_{ds} \\ Y_{ds} \end{bmatrix} \tag{6.4.2}$$

where $l_{cam,ds} = \sqrt{X_{ds}^2 + Y_{ds}^2}$.

3) Transform from undistorted coordinates, $[X_u, Y_u]^T$, to camera coordinates, $[X_{cam}, Y_{cam}, Z_{cam}]^T$:

$$\begin{bmatrix} X_{cam} \\ Y_{cam} \end{bmatrix} = \begin{bmatrix} X_u R_{cam,31} - f_{cam} & Y_u R_{cam,32} \\ X_u R_{cam,31} & Y_u R_{cam,32} - F_{ocus} \end{bmatrix}^{-1} \begin{bmatrix} X_u \left(ZR_{cam,33} + T_z\right) \\ Y_u \left(ZR_{cam,33} + T_z\right) \end{bmatrix} \tag{6.4.3}$$

where $R_{cam,ij}$ is the i^{th} row and j^{th} column element of $\mathbf{R}_{cam}$. Note that in my experimental setup, the human and robot Z-axis heights are known. We have: $Z_{cam} = R_{cam,32}^{-1}\left(Z + T_z - X_u R_{cam,31} - Y_u R_{cam,32}\right)$.

4) Transform from camera coordinates, $[X_{cam}, Y_{cam}, Z_{cam}]^T$ to world coordinates, $[X, Y, Z]$ is:

$$\begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = \mathbf{R}_{cam}^{-1} \left(\begin{bmatrix} X_{cam} \\ Y_{cam} \\ Z_{cam} \end{bmatrix} - \mathbf{T} \right) \tag{6.4.4}$$

By applying the above procedure, the color patch centroid positions of the human and robot in world coordinates are estimated from their pixel positions.

Appendix E:

E.1 Design of the nonholonomic mobile robot

Nonholonomic mobile robots have two controllable DOFs. One DOF translates the robot with a linear velocity, v; the other turns the robot with an angular velocity, $\omega = \dot{\phi}$, as shown in Fig. E.1. Nonholonomic robots are much more popular than holonomic robots. They only require two motors rather than the three motors required for holonomic robots. In this thesis, the nonholonomic robot has a differential-drive design. Two electric motors are used to rotate Wheel 1 and Wheel 2, respectively. Controlling the speeds of those two wheels will make the robot turn left/right and move forwards/backwards. Wheel 3 is an omni-directional wheel used to provide support. The designed and manufactured robot is shown in Fig. 7.1. The kinematics equation of the robot is expressed as:

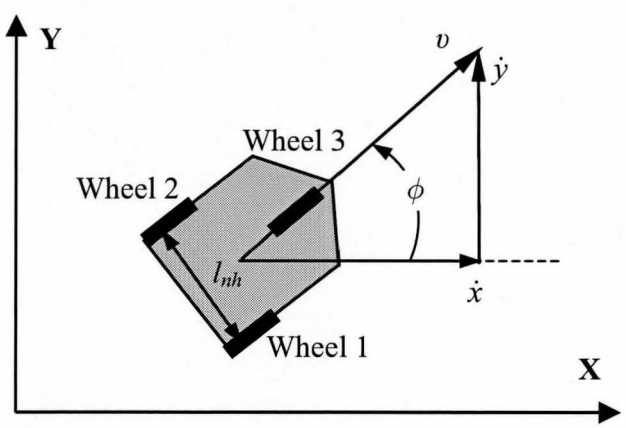


Fig. E.1. Kinematics of a differential-drive nonholonomic mobile robot.

$$\begin{bmatrix}\dot{x}\\ \dot{y}\\ \dot{\phi}\end{bmatrix}=\begin{bmatrix}\cos\phi & 0\\ \sin\phi & 0\\ 0 & 1\end{bmatrix}\begin{bmatrix}\upsilon\\ \omega\end{bmatrix} \tag{E.1.1}$$

where $\begin{bmatrix}\dot{x} & \dot{y}\end{bmatrix}^{\mathrm{T}}$ is the linear velocity of the robot and $\dot{\phi}$ is the angular velocity of the robot. According to Fig. E.1, the equation of the nonholonomic constraints is:

$$\dot{x}\sin\phi+\dot{y}\cos\phi=0 \tag{E.1.2}$$

This physical meaning of this constraint is the nonholonomic robot cannot move sideways without turning first. Due to this constraint, the force component of the virtual force, $\mathbf{F}_v$, perpendicular to υ cannot drag the robot move sideways. The angular velocities of Wheel 1 and Wheel 2 are obtained as:

$$\begin{bmatrix}\dot{\theta}_1\\ \dot{\theta}_2\end{bmatrix}=\frac{1}{r_{w,nh}}\begin{bmatrix}1 & \frac{1}{2}l_{nh}\\ 1 & -\frac{1}{2}l_{nh}\end{bmatrix}\begin{bmatrix}\upsilon\\ \omega\end{bmatrix} \tag{E.1.3}$$

where $r_{w,nh}$ equals the radii of Wheels 1 and 2 (*i.e.* 0.025 m for our robot); l_{nh} is the span between Wheel 1 and 2 (*i.e.* 0.28 m for our robot); and $\dot{\theta}_1$ and $\dot{\theta}_2$ are the angular velocities of Wheel 1 and Wheel 2, respectively. With (E.1.3), the relationship between $\begin{bmatrix}\upsilon & \omega\end{bmatrix}^{\mathrm{T}}$ and the angular velocities of Wheel 1 and Wheel 2 is:

$$\begin{bmatrix}\upsilon\\ \omega\end{bmatrix}=\frac{r_{w,nh}}{4}\begin{bmatrix}2 & 2\\ l_{nh}^{-1} & -l_{nh}^{-1}\end{bmatrix}\begin{bmatrix}\dot{\theta}_1\\ \dot{\theta}_2\end{bmatrix} \tag{E.1.4}$$

Substituting (E.1.4) into (E.1.1), the forward kinematics of the robot is:

$$\begin{bmatrix}\dot{x}\\ \dot{y}\\ \dot{\phi}\end{bmatrix}=\frac{r_{w,nh}}{4}\begin{bmatrix}2\cos\phi & 2\cos\phi\\ 2\sin\phi & 2\sin\phi\\ l_{nh}^{-1} & -l_{nh}^{-1}\end{bmatrix}\begin{bmatrix}\dot{\theta}_1\\ \dot{\theta}_2\end{bmatrix} \tag{E.1.5}$$

The inverse kinematics equation of the robot then is:

$$\begin{bmatrix}\dot{\theta}_1\\ \dot{\theta}_2\end{bmatrix}=\frac{1}{r_{w,nh}}\begin{bmatrix}\cos\phi & \sin\phi & 0.5l_{nh}\\ \cos\phi & \sin\phi & -0.5l_{nh}\end{bmatrix}\begin{bmatrix}\dot{x}\\ \dot{y}\\ \dot{\phi}\end{bmatrix} \tag{E.1.6}$$

E.2 Identification of the dynamic model of the nonholonomic robot

The dynamic equation for the i^{th} wheel of the nonholonomic robot is defined as:

$$\ddot{\theta}_i=\frac{K_{nh,i}}{J_{nh,i}}u_{nh,i}-\frac{B_{nh,i}}{J_{nh,i}}\dot{\theta}_i-\frac{\tau_{nh,i}}{J_{nh,i}}\mathrm{sgn}\left(\dot{\theta}_i\right) \tag{E.2.1}$$

where $K_{nh,i}$ is the product of the motor torque constant, the amplifier gains (*i.e.* 1 A/V) and the gear ratio (*i.e.* 10:1); $\tau_{nh,i}$ is the static friction torque of the i^{th} motor and gear; $B_{nh,i}$ is the viscous friction coefficient of the i^{th} motor and gear; $J_{nh,i}$ is the effective inertia of the robot plus the inertia of the motor, gear and wheel; $\dot{\theta}_i$ is the angular velocity of i^{th} wheel; and $\ddot{\theta}_i$ is its acceleration.

Open-loop tests were performed to obtain $\frac{K_{nh,i}}{J_{nh,i}}$, $\frac{B_{nh,i}}{J_{nh,i}}$ and $\frac{\tau_{nh,i}}{J_{nh,i}}$ for different values of the input $u_{nh,i}$. During the tests, Wheels 1 and 2 were moved at same time. At first, both motors were given a positive signal to drive the robot move forwards. Next, both motors were given a negative signal to drive the robot move backwards. The tests for driving the two motors both forwards and backwards direction were performed in this fashion. In every test, control commands with the identical magnitudes were sent to the two motors. The procedure used is as follows:

1. Perform open-loop tests by sending different commands (*i.e.* $u_{nh,i}$ = 0.25, 0.3, 0.35, 0.4, 0.45 and 0.5 V), send each command twice. Obtain position profiles θ_i vs. time t for those tests.

2. Let us set $\kappa_{nh,i} = \frac{B_{nh,i}}{J_{nh,i}}$. $u_{nh,i}$ is constant for these open-loop tests and we assume the static friction torque, $\tau_{nh,i}$, is also a constant value. Then, $A_{nh,i} = \frac{K_{nh,i}}{J_{nh\,i}} u_{nh,i} + \frac{\tau_{nh,i}}{J_{nh,i}}$ is a constant, and we can rewrite (E.2.1) as

$$\ddot{\theta}_i = A_{nh,i} - \kappa_{nh,i}\dot{\theta}_i. \tag{E.2.2}$$

3. Use the Laplace transform and inverse Laplace transform to solve (E.2.2), to obtain θ_i in time domain:

$$\theta_i = \kappa_{nh,i} \frac{A_{nh,i}t - 1 + \exp\left(-A_{nh,i}t\right)}{A_{nh,i}} \tag{E.2.3}$$

4. Use the function *lspcurvefit* in MATLAB to fit the position profiles, θ_i vs. t in the least-squares sense to find the values of $A_{nh,i}$ and $\kappa_{nh,i}$. For the fitting, the initial values of those parameters $A_{nh,i}$ and $\kappa_{nh,i}$ are calculated with the values of $K_{nh,i}$, $B_{nh,i}$ and $f_{nh,i}$ from the datasheet of the motor (Maxon motor, RE40-148877). In this step, we will obtain a series of $A_{nh,i}$ and $\kappa_{nh,i}$ values.

5. The mean values of $\kappa_{nh,i}$ are obtained from the series of $\kappa_{nh,i}$ values. They are listed in Table E.1.

6. To obtain the parameters $\frac{K_{nh,i}}{J_{nh,i}}$ and $\frac{\tau_{nh,i}}{J_{nh,i}}$,

$$A_{nh,i} = \frac{K_{nh,i}}{J_{nh,i}} u_{nhi} - \frac{\tau_{nh,i}}{J_{nh,i}}, \tag{E.2.4}$$

We first define $\mathbf{Y}_{nh,i}$ as the vector of $A_{nh,i}$ values from step 4; and $\mathbf{u}_{nh,i}$ = [0.25, 0.3, 0.35, 0.4, 0.45, 0.5]^{T} V. Then (E.2.4) can be rewritten in the format of the first order linear function $\mathbf{Y}_{nh,i} = \left(\frac{K_{nh,i}}{J_{nh,i}}\right)\mathbf{U}_{nh,i} - \left(\frac{\tau_{nh,i}}{J_{nh,i}}\right)$. We use the linear regression method as follows:

$$\frac{K_{nh,i}}{J_{nh,i}} = \frac{6\mathbf{U}_{nh,i}\mathbf{Y}_{nh,i}^{\mathrm{T}} - \sum\mathbf{U}_{nh,i}\sum\mathbf{Y}_{nh,i}}{6\mathbf{U}_{nh,i}\mathbf{U}_{nh,i}^{\mathrm{T}} - \left(\sum\mathbf{U}_{nh,i}\right)^{2}}; \tag{E.2.5a}$$

$$\frac{\tau_{nh,i}}{J_{nh,i}} = \frac{1}{6}\left(\sum\mathbf{Y}_{nh,i} - \frac{K_{nh,i}}{J_{nh,i}}\sum\mathbf{U}_{nh,i}\right) \tag{E.2.5b}$$

The results from (E.2.5a) and (E.2.5b) are listed in Table E.1

Table E.1. Estimated dynamic model parameters for the two motors

Motor No.	***Direction***	$\kappa_{nh,i}$	$\frac{K_{nh,i}}{J_{nh,i}}$	$\frac{\tau_{nh,i}}{J_{nh,i}}$
1	Forward	0.43	130	20.5
	Backward	0.38	125	25.3
2	Forward	0.44	176	46.4
	Backward	0.59	173	52.1

Since the differences between the forward and backward parameters for the wheels are rather small, the dynamic model for each wheel will be described with only one model and the dynamic model parameters are the average values of the forward and backward data; so we have:

$$\begin{aligned}\ddot{\theta}_1 &= 128u_{nh,1} - 0.41\dot{\theta}_1 - 22.9\,\mathrm{sgn}\left(\dot{\theta}_1\right)\\ \ddot{\theta}_2 &= 175u_{nh,2} - 0.52\dot{\theta}_2 - 49.3\,\mathrm{sgn}\left(\dot{\theta}_2\right)\end{aligned} \tag{E.2.6}$$

E.3 Design of the wheel controllers

The controller for each wheel is a FF plus PD controller. The controller is

$$u_{nh,i} = \underbrace{\left(K_{P,nh,i}e_{nh,i} + K_{D,nh,i}\dot{e}_{nh,i}\right)}_{PD\ Feedback} + \underbrace{\frac{J_{nh,i}}{K_{nh,i}}\ddot{\theta}_{d,i}}_{\substack{Acceleration\\ Feedforward}} + \underbrace{\frac{B_{nh,i}}{K_{nh,i}}\dot{\theta}_{d,i}}_{\substack{Viscous\ Friction\\ Feedforward}} + \underbrace{\frac{\tau_{nh,i}}{K_{nh,i}}\mathrm{sgn}(\dot{\theta}_{d,i})}_{\substack{Static\ Friction\\ Feedforward}} \tag{E.3.1}$$

where $e_{nh,i} = \theta_{d,i} - \vartheta_i$, $\theta_{d,i}$ is the desired wheel angle; $\dot{\theta}_{d,i}$ is the desired wheel angular velocity; $\ddot{\theta}_{d,i}$ is the desired wheel angular acceleration; $K_{P,nh,i}$ is the positive proportional control gain for i^{th} wheel; and $K_{D,nh,i}$ is the positive derivative control gain. The two gains for the two wheels were obtained by manually tuning, and are listed in Table E.2. The feedforward parameters come from (E.2.6).

Table E.2. The proportional and derivative control gains; and feedforward parameters.

Gains	Wheel 1	Wheel 2
$K_{P,nh,i}$	34.0	35.0
$K_{D,nh,i}$	1.0	1.0
$J_{nh,i}/K_{nh,i}$	0.0079	0.0057
$B_{nh,i}/K_{nh,i}$	0.0032	0.0030
$\tau_{nh,i}/K_{nh,i}$	0.18	0.28

The Luenberger observers presented in Section D.3 of Appendix D are also used in the control system for the nonholonomic robot. Based on several experiments, the

observer poles were chosen as $p_{1,i} = -10$ and $p_{2,i} = -10$. Fig. E.2 shows the closed-loop control experiment results for the nonholonomic robot.

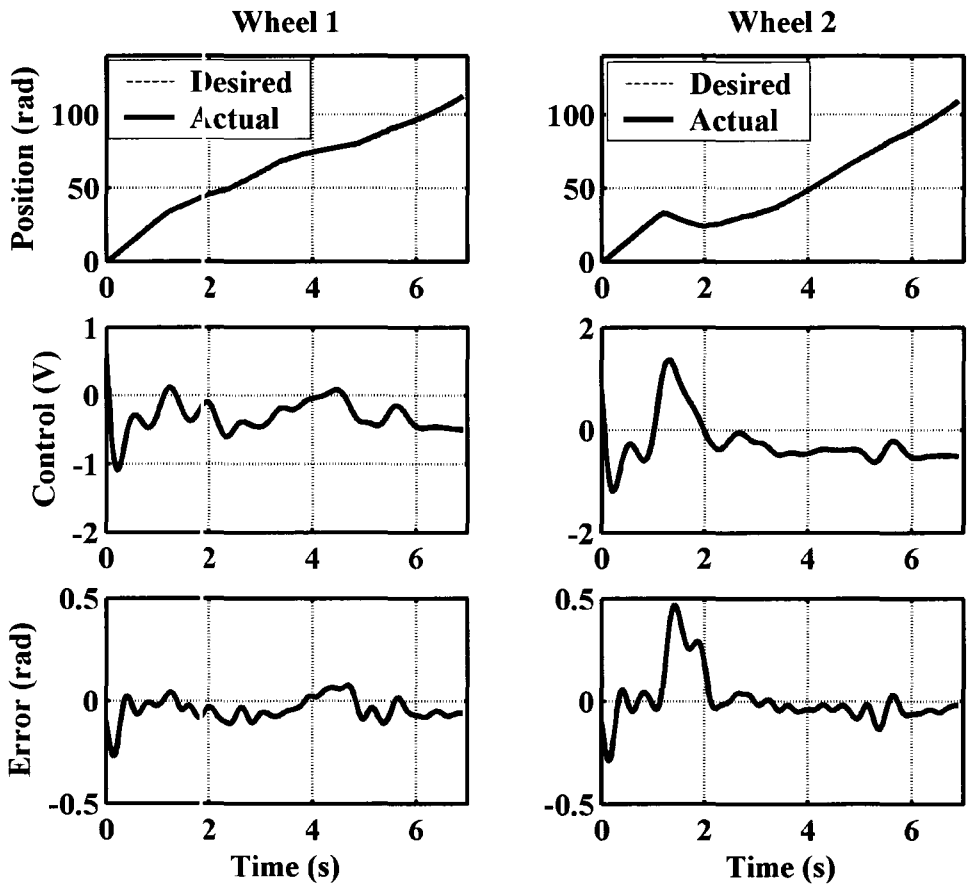


Fig. E.2. Example closed-loop experiment results for the nonholonomic robot.

Note that (E.3.1) with $\mathbf{i} = [i_1 \quad i_2]^T$ are used to compute the values of L_{energy} and H for simulations with the nonholonomic robot in Chapter 7. For those simulations, we assumed the tracking is perfect (*i.e.* $e_{nh,i} = 0$ and $\dot{e}_{nh,i} = 0$); so the feedback PD controllers do not contribute to the control commands. From (E.3.1) and (E.2.6), we have:

$$u_{nh,1} = 0.0079\ddot{\theta}_{d,1} + 0.0032\dot{\theta}_{d,1} + 0.18\,\mathrm{sgn}\left(\dot{\theta}_{d,1}\right) \tag{E.3.2a}$$

$$u_{nh,2} = 0.0057\ddot{\theta}_{d,2} + 0.0030\dot{\theta}_{d,2} + 0.28\,\mathrm{sgn}\left(\dot{\theta}_{d,2}\right) \tag{E.3.2b}$$

After the amplifier, the input currents to the motors are:

$$\mathbf{i} = \begin{bmatrix} i_1 \\ i_2 \end{bmatrix} = \begin{bmatrix} K_{amp} & 0 \\ 0 & K_{amp} \end{bmatrix} \begin{bmatrix} u_{nh,1} \\ u_{nh,2} \end{bmatrix} \tag{E.3.3}$$

where K_{amp} = 1 A/V is the amplifier gain. Eq. (E.3.3) will be used in (5.6.5) to calculate L_{energy}. In the experiments, the PD controllers contribute to the control commands. We directly recorded the motor current data and used them in (5.6.5).

Appendix F:

F.1 Repeatability of the experiments with the holonomic robot

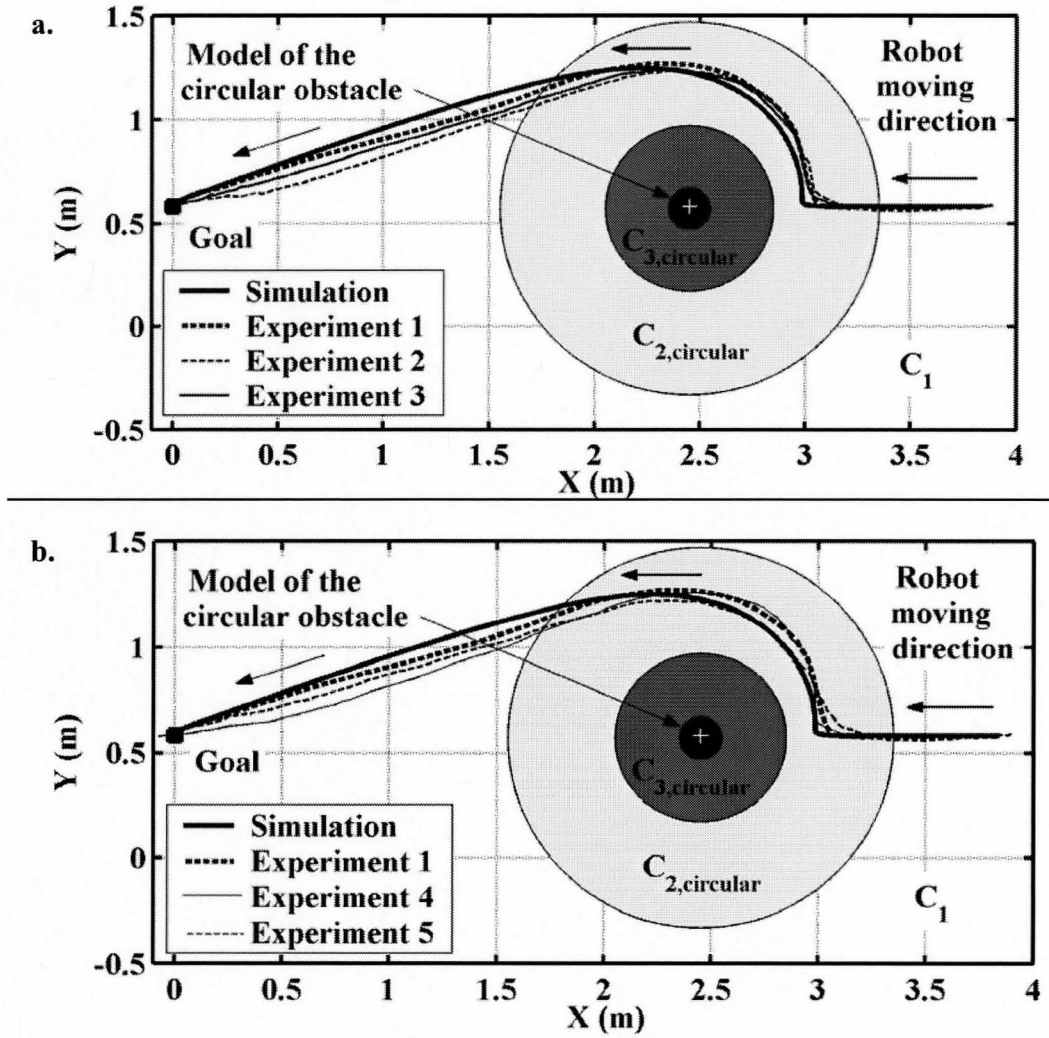


Fig. F.1. Robot paths in five experiments with the holonomic robot for Configuration 1.

Fig F.1 shows the robot paths in five experiments with the holonomic robot for Configuration 1. e and d for these experiments are shown in Fig. F.2. We can see the experiments are all similar to the simulation and to each other. The differences of the

paths and values of e and d among those experiments are mainly from the sliding of the wheels. Note that in these figures, Experiment 1 denotes the experiment results we presented in Chapter 6.

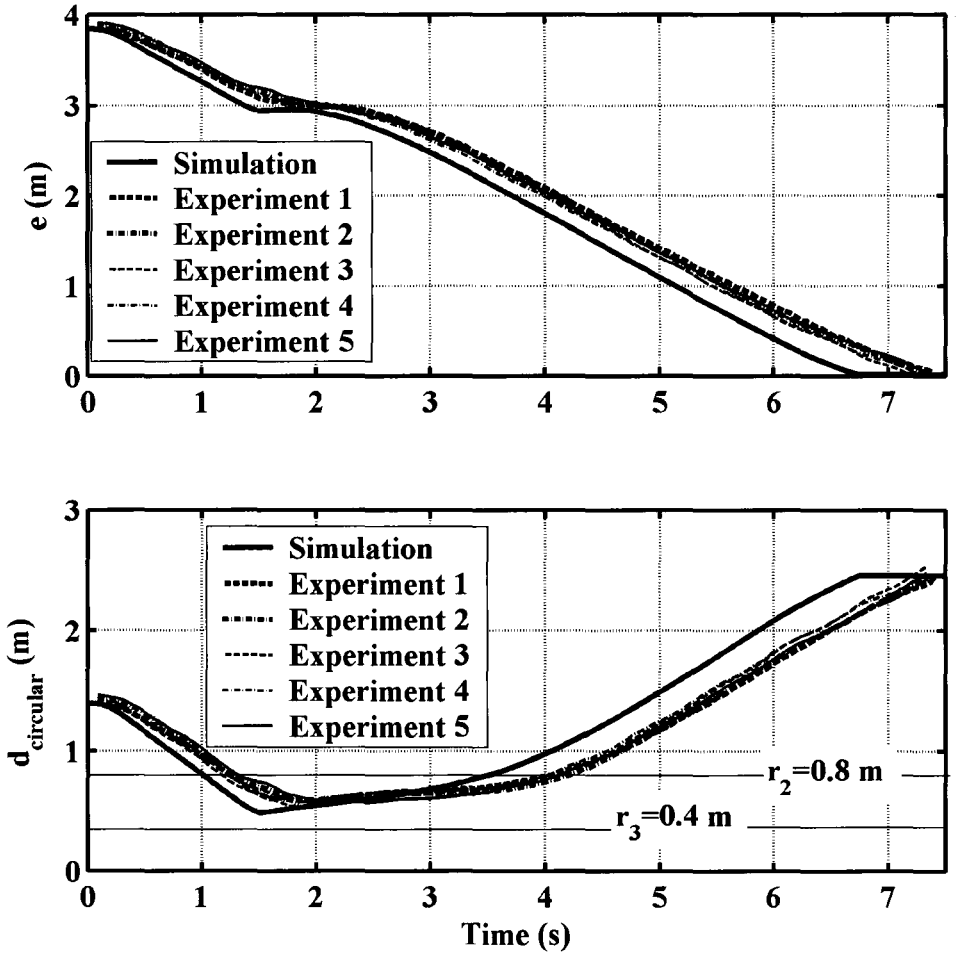


Fig. F.2. e and d of the five experiments in Fig. F.1.

F.2 Repeatability of the experiments with the nonholonomic robot

Fig F.3 shows the robot paths in five experiments with the nonholonomic robot for Configuration 1. e and d for these experiments are shown in Fig. F.4. We can see the

experiments are all similar to the simulation and to each other. Due to less sliding of the wheels, the differences of the paths and values of e and d among these experiments are small. Note that in these figures, Experiment 1 denotes the experiment results we presented in Chapter 7.

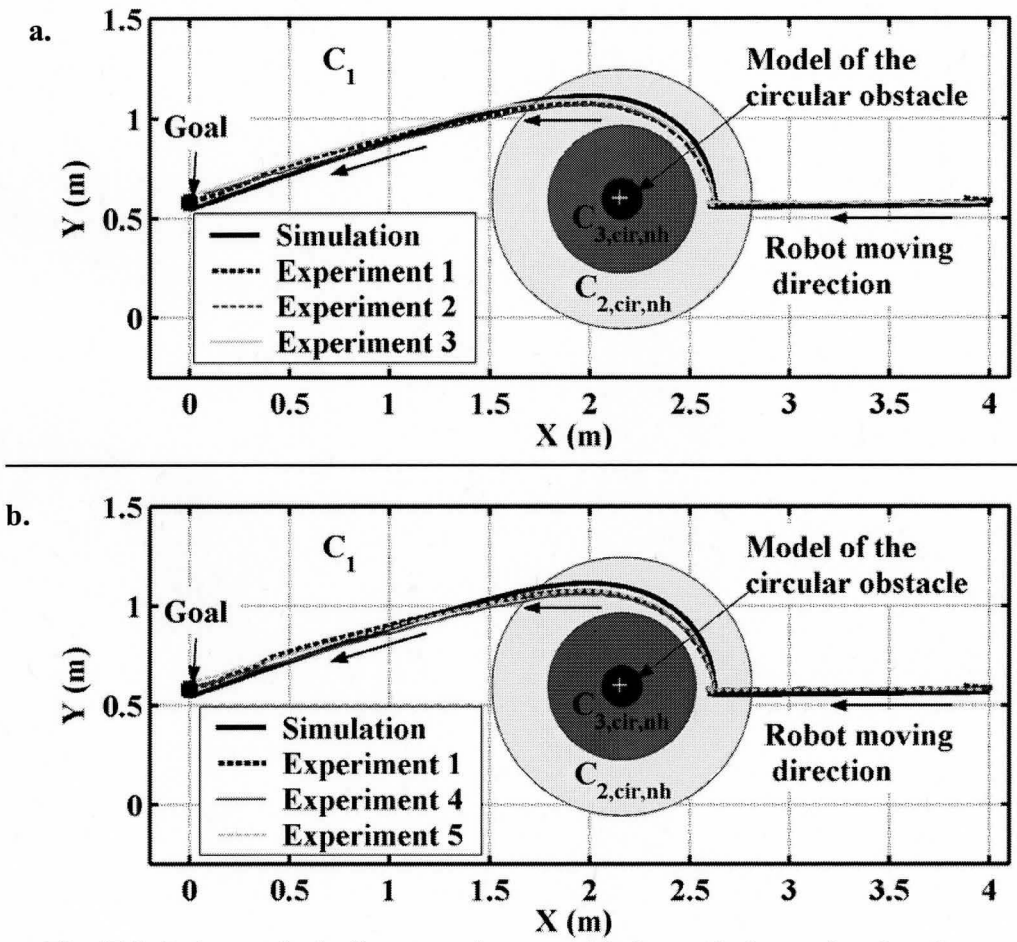


Fig. F.3. Robot paths in five experiments with the nonholonomic robot for Configuration 1.

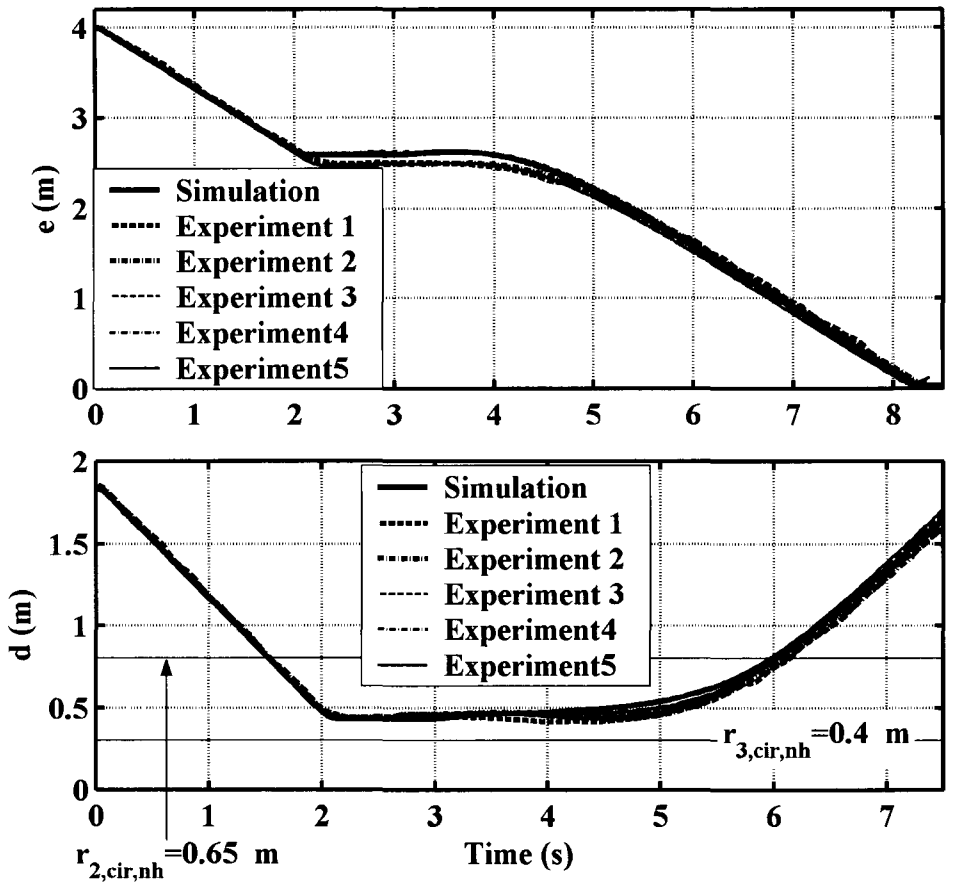


Fig. F.4. *e* and *d* of the five experiments in Fig. F.3.