

**MULTI-POINT PARAMETER EXTRACTION:
THE CONCEPT AND ITS IMPLEMENTATION**

M.H. Bakr

SOS-98-24-R

August 1998

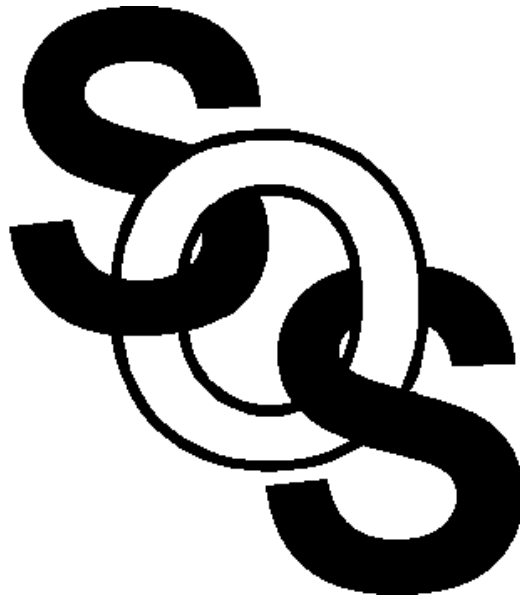
© M.H. Bakr 1998

No part of this document may be copied, translated, transcribed or entered in any form into any machine without written permission. Address inquiries in this regard to Dr. J.W. Bandler. Excerpts may be quoted for scholarly purposes with full acknowledgment of source. This document may not be lent or circulated without this title page and its original cover.

MULTI-POINT PARAMETER EXTRACTION: THE CONCEPT AND ITS IMPLEMENTATION

Mohamed Bakr

Simulation Optimization Systems Research Laboratory
and Department of Electrical and Computer Engineering
McMaster University, Hamilton, Canada L8S 4L7



presented at

SOS Research Laboratory Meeting, Hamilton, August 18, 1998.



Single-Point Parameter Extraction

given a set of measured responses \mathbf{R}_m of a certain system, parameter extraction obtains the corresponding system parameters that regenerate these responses

parameter extraction can be formalized as

$$\underset{\mathbf{x}}{\text{minimize}} \|\mathbf{R}_m - \mathbf{R}(\mathbf{x})\|$$

where \mathbf{x} is the vector of system parameters and \mathbf{R} is the vector of system responses

a suitable optimizer is then used to solve this optimization problem



Parameter Extraction and Space Mapping

parameter extraction is a crucial step in any space mapping algorithm

given a set of fine model parameters \mathbf{x}_{em} with corresponding fine model responses \mathbf{R}_{em} , parameter extraction aims at obtaining the set of coarse model parameters \mathbf{x}_{os} whose response $\mathbf{R}_{os}(\mathbf{x}_{os})$ matches the vector \mathbf{R}_{em}

it follows that the parameter extraction problem can be formulated as

$$\underset{\mathbf{x}_{os}}{\text{minimize}} \left\| \mathbf{R}_{em}(\mathbf{x}_{em}) - \mathbf{R}_{os}(\mathbf{x}_{os}) \right\|$$

the extracted coarse model parameters are then used to predict the next iterate

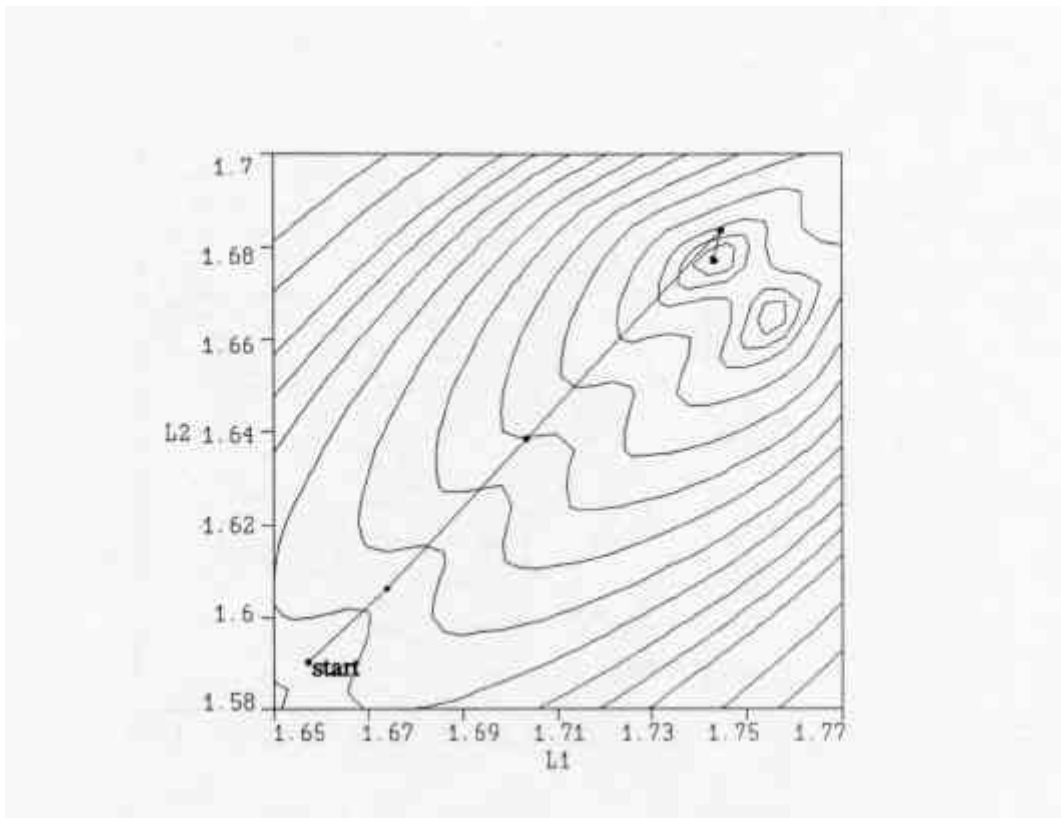
the extracted coarse model parameters may be nonunique

this nonuniqueness may lead to the divergence or oscillation of the space mapping algorithm



Example of a Nonunique Extracted Parameters

This figure shows the contours of the objective function for a single-point parameter extraction process



(Bandler et al., 1996)



The Multi-Point Parameter Extraction

to overcome problems caused by nonuniqueness of the parameter extraction step a multi-point parameter extraction procedure was suggested (*Bandler et al., 1996*)

the step aims at matching not only the response but also the first-order derivative of the two models

the extracted coarse model point \mathbf{x}_{os} matches the responses of both models at a number of points

it follows that \mathbf{x}_{os} should satisfy

$$\mathbf{R}_{os}(\mathbf{x}_{os} + \Delta\mathbf{x}^{(i)}) = \mathbf{R}_{em}(\mathbf{x}_{em} + \Delta\mathbf{x}^{(i)})$$

simultaneously for a set of perturbations $\Delta\mathbf{x}^{(i)} \in V_p$; the set of perturbations used for the multi-point parameter extraction, where $\mathbf{0} \in V_p$

this step is likely to improve the uniqueness of the parameter extraction step



Comments on the Original Multi-Point Extraction

the original multi-point parameter extraction (*Bandler et al., 1996*) suffers from two main drawbacks

the first drawback is that the fine model points were arbitrarily selected

the second drawback is the assumption that a perturbation of $\Delta\mathbf{x}$ in the fine model space corresponds to an equal perturbation in the coarse model space

in the j th iteration, the most recent information about the mapping between the two spaces is given by the matrix $\mathbf{B}^{(j)}$, which should be integrated with the multi-point parameter extraction step



The Recursive Multi-Point Parameter Extraction

(Bakr et al., 1998)

this approach was developed within the context of the trust region aggressive space mapping algorithm (TRASAM)

every fine model point that is generated by the algorithm is fed back to improve the uniqueness of the extraction step

the available information about the mapping between the two spaces is exploited in the extraction step

in the j th iteration the extracted coarse model parameters should satisfy

$$\mathbf{R}_{os}(\mathbf{x}_{os} + \mathbf{B}^{(j)} \Delta \mathbf{x}^{(i)}) = \mathbf{R}_{em}(\mathbf{x}_{em} + \Delta \mathbf{x}^{(i)})$$

simultaneously for a set of perturbations $\Delta \mathbf{x}^{(i)} \in V_p$; the set of perturbations used for the multi-point parameter extraction where $\mathbf{0} \in V_p$

this step is assumed to be superior to the originally suggested multi-point parameter extraction as it takes into account the available information about the mapping between the two spaces



Multi-Point Parameter Extraction:the Current Implementation

currently the multi-point parameter extraction is implemented in OSA90/hope

the utilized fine model points and the corresponding fine model responses are stored in two different matrices

two indices are used to determine the response and the fine model point under consideration



File Template

```
Model
  Ros= ; !coarse model as function of the vector Xos
end

expression
  i:1; ! Index of fine model points
  n= ; ! Number of Extracted parameters
  m= ; ! Number of responses
  N_points= ; ! Number of points used for the multi-point extraction
  j= ; ! j as a function of frequency (Index of responses)
  Xos_Matrix[1,n]=[ ? ? ? ? ]; ! Starting Point of optimization
  X_fine_Matrix[(n+1),n]=[
                                     ]; ! The Points used for the multi-point
                                     ! extraction
  X_fine_Responses[(n+1),m]=[
                                     ]; ! The corresponding fine model responses
  B[n,n]=[
           ]; ! The current B matrix
  X_fine[1:n]=row(X_fine_Matrix,i);
  X_fine_1[1:n]=row(X_fine_Matrix,1);
  X_coarse1[1:n]=row(Xos_Matrix,1);
  X_fine_Difference[1:n]=X_fine-X_fine_1;
  X_coarse_Difference[1:n]=product(B,X_fine_Difference);
  Xos[1:n]=X_coarse_Difference+X_coarse1;
end

specification
  i: from 1 to N_points step 1
    freq: from startf to endf step=stepf
    Ros=X_fine_Responses[i,j];
end

Control
  Optimizer=Huber;
  N_iterations=99;
  Huber_threshold=0.01;
  accuracy=1.0e-6;
  two_sided_jacobian;
end

report
  %12.9f$ P = [ $ Xos_matrix[1,1]$
                $ Xos_matrix[1,2]$ ]
end
```



Example: a Quadratic Function

the coarse model for this problem is given by

$$R_{os} = x_1^2 + x_2^2$$

the fine model for this problem is given by

$$R_{em} = (0.9x_1 + 0.1x_2)^2 + (0.1x_1 + 0.9x_2)^2$$

it is required in this problem to extract the coarse model parameters corresponding to the fine model parameters

$$\mathbf{x}_{em} = [2.0 \quad 1.0]^T$$

four fine model points were needed to ensure the uniqueness of the extracted parameters



OSA90 File for the Quadratic Function Problem

```
Model
  Ros=(Xos[1]*Xos[1])+(Xos[2]*Xos[2]);
end
expression
  i:1;
  n=2;
  m=1;
  N_Points=4;
  j=1;
  Xos_Matrix[1,n]=[ 1.89571?  1.10868? ];!coarse model points
  X_fine_Matrix[(n+2),n]= [   2.0      1.0
                             1.8217   0.9094
                             2.1414   1.1414
                             1.8586   0.8586 ];
  X_fine_Responses[(n+2),m]=[ 4.82
                              3.9958
                              5.7085
                              4.0115 ];

  B[n,n]=[1  0
          0  1];
  X_fine[1:n]=row(X_fine_Matrix,i);
  X_fine_1[1:n]=row(X_fine_Matrix,1);
  X_coarse1[1:n]=row(Xos_Matrix,1);
  X_fine_Difference[1:n]=X_fine-X_fine_1;
  X_coarse_Difference[1:n]=product(B,X_fine_Difference);
  Xos[1:n]=X_coarse_Difference+X_coarse1;
end
specification
  i: from 1 to N_points step 1
    Ros=X_fine_Responses[i,j];
end

Control
  Optimizer=Huber;
  N_iterations=99;
  Huber_threshold=0.01;
  accuracy=1.0e-6;
  two_sided_jacobian;
end

report
  %12.9f$ P = [ $ Xos_matrix[1,1]$
               $ Xos_matrix[1,2]$ ]
end
```



The Variation of the Extracted Parameters with the Number of Fine Model Points for the Quadratic Function

the following table shows the variation of the extracted coarse model parameters with the number of fine model points used in the multi-point parameter extraction

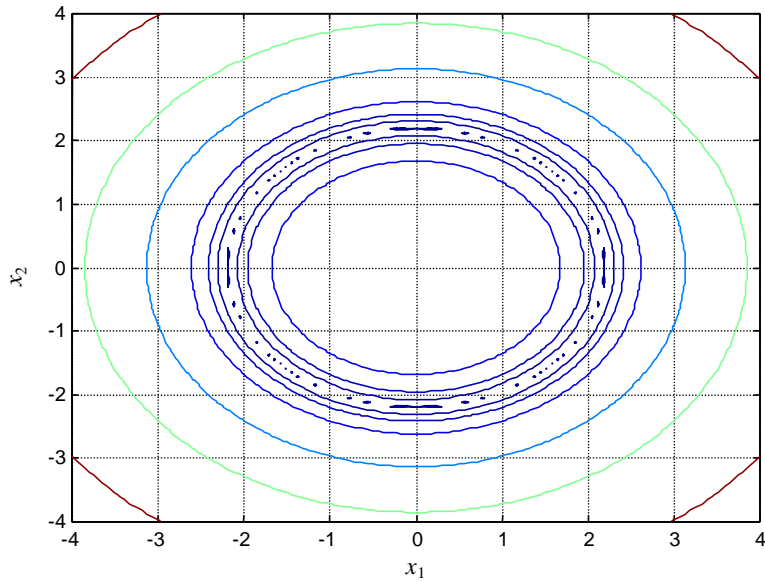
| Number of Points | $x_{os,1}$ | $x_{os,2}$ |
|------------------|------------|------------|
| 1 | 1.95724 | 0.99458 |
| 2 | 2.10283 | 0.63094 |
| 3 | 1.92787 | 1.05337 |
| 4 | 1.89571 | 1.10868 |

the exact solution for the parameter extraction problem is

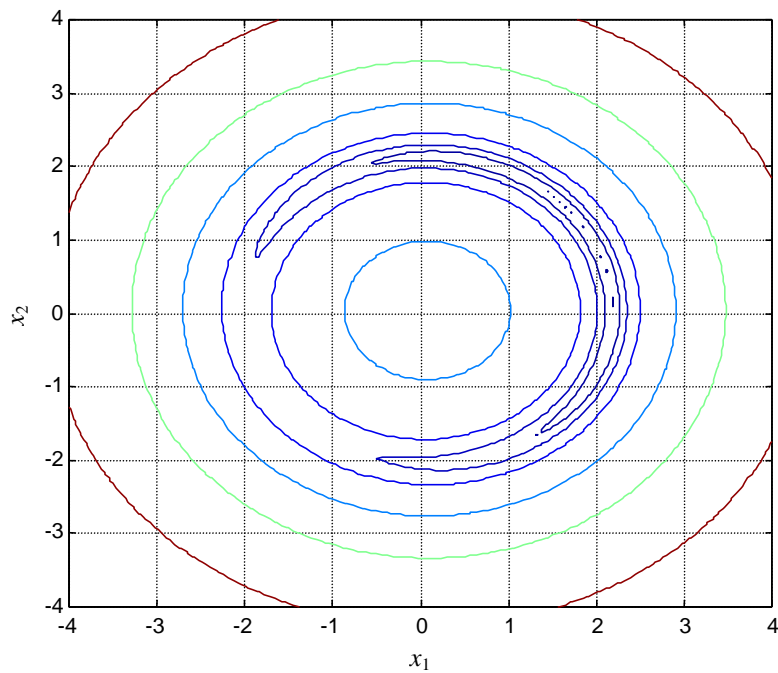
$$\mathbf{x}_{os} = [1.9 \quad 1.1]^T$$



The Contours of the L_2 Objective Function for the Quadratic Function



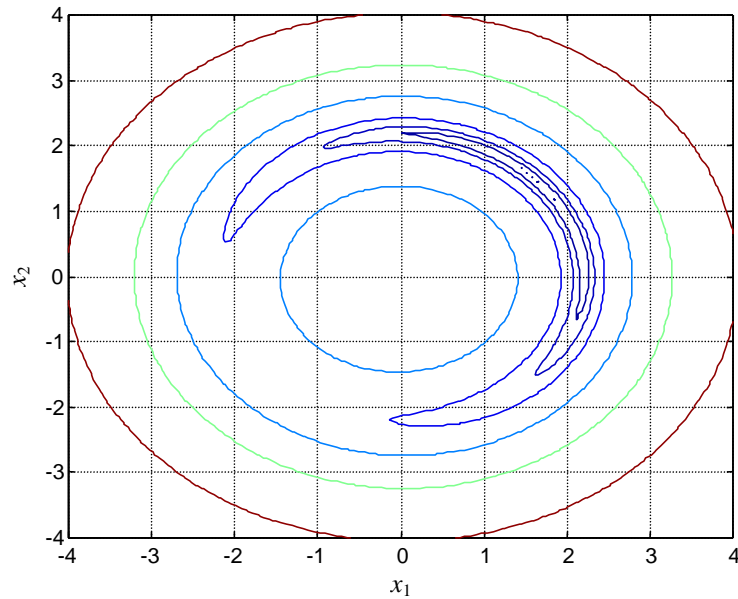
(a) single-point extraction



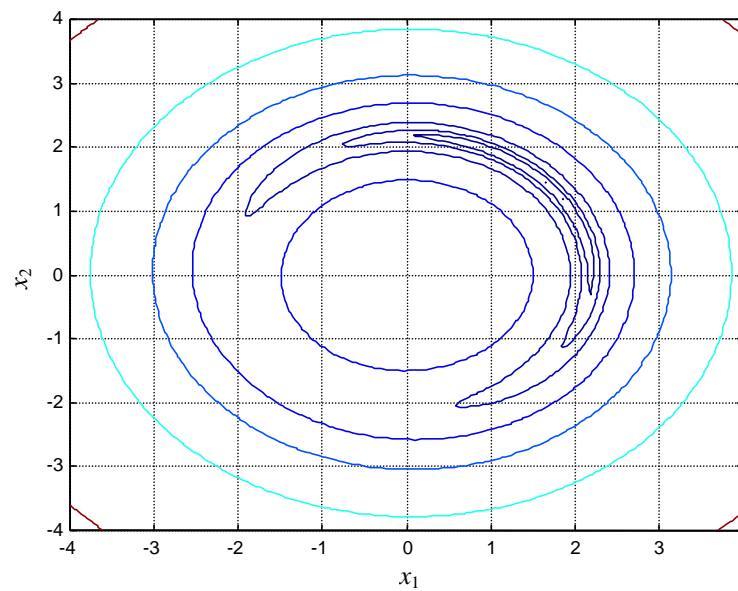
(b) two-point extraction



The Contours of the L_2 Objective Function for the Quadratic Function



(c) three-point extraction



(d) four-point extraction



OSA90 File for a Frequency Dependent Response

```
Model
  ICONS = 0;
  ideal: 0;
  M1: 2;
  M2 = M1 + 2;
  A[M2] = [3.48488 3.6 3.8 4.0386];
  B1[M2] = [0.508 Xos[1] Xos[2] 2.0193];
  L1[M2] = [0 Xos[3] Xos[4] 0];
  Datapipe: SIMD FILE="simwgd"
              N_INPUT=(3 * M2 + 4)
              INPUT=(M1, A, B1, L1, ICONS, ideal, FREQ)
              N_OUTPUT=1 OUTPUT=(VSWR_ideal);
end
expression
  i:1; n=4;
  m=17;
  N_Points=1;
  Kx: ((freq-5.8)/0.05)+1;
  j=nint(Kx);
  Xos_Matrix[1,n]=[ ?0.711967? ?1.39486? ?1.65733? ?1.59002? ];
  X_fine_Matrix[(n+1),n]= [ ];
  X_fine_Responses[(n+1),m]= [ ];
  B[n,n]=[ ];
  X_fine[1:n]=row(X_fine_Matrix,i);
  X_fine_1[1:n]=row(X_fine_Matrix,1);
  X_coarse1[1:n]=row(Xos_Matrix,1);
  X_fine_Difference[1:n]=X_fine-X_fine_1;
  X_coarse_Difference[1:n]=product(B,X_fine_Difference);
  Xos[1:n]=X_coarse_Difference+X_coarse1;
end
specification
  i: from 1 to N_points step 1
      freq: from 5.8 to 6.6 step=0.05
          VSWR_ideal=X_fine_Responses[i,j];
end
sweep
  freq: from 5.8 to 6.6 step=0.05
      VSWR_ideal X_fine_Responses[1,j] {Xsweep Y=VSWR_ideal.white&
X_fine_Responses[1,j].white.circle Ymin=1 Ymax=1.04 Nyticks=4 xmin=5.7
xmax=6.7 Nxticks=10} ;
end
Control
  Optimizer=Huber;
  N_iterations=99;
  Huber_threshold=0.01;
  accuracy=1.0e-6;
  two_sided_jacobian;
end
report
  %12.9f$ P = [ $ Xos_matrix[1,1]$
                $ Xos_matrix[1,2]$
                $Xos_matrix[1,3]$
                $Xos_matrix[1,4]$ ]
```




end