COST OPTIMIZATION OF MODULAR DATA CENTERS

COST OPTIMIZATION OF MODULAR DATA CENTERS

By SUCHITRA NAYAK, B. TECH, MECHANICAL ENGINEERING

A Thesis Submitted to the School of Graduate Studies in Partial Fulfilment of the Requirements for the Degree Master of Applies Science

McMaster University MASTER OF APPLIED SCIENCE (2018) Hamilton

Ontario

(Mechanical Engineering)

TITLE: Cost Optimization of Modular Data Center

AUTHOR: Suchitra Nayak, B. Tech (National Institute of Technology, Trichy, India), Mechanical Engineering

SUPERVISOR: Professor Ishwar K. Puri

NUMBER OF PAGES: 110

ABSTRACT

During the past two decades, the increasing demand for digital telecommunications, data storage and data processing coupled with simultaneous advances in computer and electronic technology have resulted in a dramatic growth rate in the data center (DC) industry. It has been estimated that almost 2% of US total energy consumption and 1.5% of worlds total power consumption is by DCs. With the fossil fuels and earth's natural energy sources depleting every day, greater efforts have to be made to save energy and improve efficiencies. As yet, most of the DCs are highly inefficient in energy usage. A significant part of this inherent inefficiency comes from poor design and rudimentary operation of current DCs. Thus, there is an urgent need to optimize the power consumption of DCs. This has led to the advent of modular DCs, newer scalable DC architectures, that reduces cost and increases efficiency by eliminating overdesign and allowing for scalable growth. This concept has been particularly appealing for small businesses who find it difficult to commit to setting up a traditional DC with huge upfront capital investment. However, their adoption and implementation is still limited because of a systematic approach of quickly identifying a module DC design. Considering many different choices for subcomponents, such as cooling systems, enclosures and power systems, this is a non-trivial exercise, especially, considering the complex multiphysics interactions among components that drive system efficiency. For designing such DCs, there is no research available. Therefore, most of the time, the engineers and designers rely on experience, to avoid lengthy elaborate engineering

analysis, particularly during the conception stages of a DC deployment project. Here, we are developing a design tool that will not only optimize the design of modular DCs but also make the design process much faster than manually done by engineers. One of the major problem in designing modular DCs is finding optimum placement of the cooling unit to keep the temperature under ASHRAE guidelines (recommended safe temperature threshold). In addition to finding the optimum selection and placement of the cooling units and its auxiliary components, the tool also gives an optimum design for the power connection to the cooling units and IT racks with redundancy. Also, a bill of materials and key performance index (KPI) for those designs are generated by the tool. Overall, this tool in the hands of the bidders or sales representatives can significantly increase their chance of winning the project.

Contents

1. Introduction..1

1.1 Problem Definition..11

1.2.2 Genetic Modular DC configuration 11

1.2 Scope of Study 12

1.2.2 Design Cost Optimization..12

1.2.2 Redundant power configuration..16

2. Methodology..17

2.1. Inefficiencies of a modular DC 17

2.2. Cooling range 18

2.3. Algorithm for minimizing excess power and cooling 19

2.3.1 Parameters defined in algorithm 22

2.3.2 Problem constraints...23

2.3.3 Objective junction...26

2.4. CFD approach for cooling curve determination.....................................27

2.5. Experimental approach for cooling curve determination........................31

2.5.1 Experimental setup...31

2.6. Understanding client requirements...35

2.7. Design algorithm based on experiments 37

2.7.1 DX (Direct Expansion) cooling units 39

2.7.2 CW (Chilled Water) cooling units 42

3. Results and Discussion 47

3.1. Validation of CFD model 47

3.2. Temperature trend vs. IT load and position 51

3.3. Parametric study of the air flow rates 52

3.4. CFD based optimization example 52

3.5. Pressure criteria for cooling range 54

3.5.1. Pressure criteria based cooling range 55

3.6. Cold aisle width effect on cooling range 59

3.7. Parameters affecting cooling range 61

3.8. Multivariable regression of dimensionless numbers 66

3.9. Redundant Systems 68

3.9.1. Power redundancy 68

3.9.2 Cooling redundancy 72

3.10. Case Studies 77

4. Conclusion and future work 79

5. References 81

6. Appendix 85

6.1 MatLab code for Optimization based on CFD 85

6.2 MatLab code for N+1 redundancy for power 92

6.3 MatLab code for optimization based on experiment – CW 96

6.4 MatLab code for optimization based on experiment – DX 103

6.2 Component data repository of cooling units used 109

6.2.1 Chilled water cooling units 109

6.2.2 Direct expansion cooling units 109

6.2.3 Chiller units 110

LIST OF FIGURES

FIGURE 1. ACTUAL IMAGE OF A MODULAR DC IN-HOUSE (INDUSTRIAL PARTNER - CINNOS). TO THE EXTREME RIGHT (BLUE LED) IS THE POWER RACK, WHICH FEEDS POWER TO THE COOLING UNIT (NEXT TO IT) AND THE IT RACKS (RED LED). THE COLD AIR IS SUPPLIED TO THE IT RACKS THROUGH THE SPACE BETWEEN THE FRONT GLASS COVER AND FRONT OF THE SERVERS................................5

FIGURE 2. SCHEMATIC REPRESENTATION OF THE ARRANGEMENT AND AIR FLOW THROUGH A MODULAR DC. THE GREY BOXES REPRESENT IT RACKS AND BLUE BOX REPRESENTS COOLING UNIT. THE COLD AIR IS SUPPLIED FROM THE FRONT OF THE COOLING UNIT (ALSO CALLED AS COLD AISLE OR COLD CHAMBER) REPRESENTED QUALITATIVELY BY THE BLACK ARROWS WHERE THE SIZE OF THE ARROW REPRESENTS AMOUNT OF COLD AIR ENTERING EACH RACK. SIMILARLY THE RED ARROWS REPRESENT THE HOT AIR COMING OUT FROM THE SERVERS AND MOVING TO THE COOLING UNIT (THROUGH THE HOT AISLE OR HOT CHAMBER). THE GAPS BETWEEN ANY RACKS IS PARTIALLY CLOSED WITH THE USE OF THE BRUSH, SO THAT IT PREVENTS COLD AIR FROM MIXING WITH HOT AIR...12

FIGURE 3. SCHEMATIC OF A TYPICAL BIDDING SCENARIO. THE MOST TIME CONSUMING PART IS THE CYCLE OF COMMUNICATION PROCESS BETWEEN THE SALES PEOPLE AND THE ENGINEERS TO FINALIZE THE PRELIMINARY DESIGN FOR CONVEYING IT TO THE CLIENTS..14

FIGURE 4. SCHEMATIC SHOWING THE INVOLVEMENT OF THE ENGINEERING TEAM WITH SALES TEAM BEFORE AND AFTER THE OPTIMIZATION TOOL IMPLEMENTATION. THIS IMPLIES THAT THE TOOL IS TAILORED TO SAVE ENGINEERING TIME ALLOWING THEM TO USE IT ON ALREADY APPROVED PROJECTS……………………………………………………………………….15

FIGURE 5. SCHEMATIC SHOWING ARRANGEMENT OF THE POWER, COOLING AND IT RACKS BY THE OPTIMIZATION TOOL. WITH THE CLIENT INPUTS AND WITH THE HELP OF THE BUILT-IN INVENTORY OF COMMERCIALLY AVAILABLE COMPONENTS THE ALGORITHM CAN GENERATE MULTIPLE DESIGNS RANKED IN THE ORDER OF THE COST FOR THE DESIGNERS TO CHOOSE FROM. FROM THE AMPLE TYPES AND CONFIGURATION OF POWER, SERVER AND COOLING MODULES, SELECTING AND ARRANGING THEM MANUALLY IN AN OPTIMUM CONFIGURATION CAN TAKE WEEKS. WITH THE TOOL, A RATHER TIME CONSUMING PROCESS OF FINDING THE BEST ARRANGEMENT CAN BE ACHIEVED SIMPLY…………………………………………………………………………16

FIGURE 6. GUI OF THE OPTIMIZATION ALGORITHM FOR EASY ACCESS FOR THE SALES REPRESENTATIVES. THE ALGORITHM IS PACKED IN AN EXECUTABLE PACKAGE WHICH CAN BE INSTALLED ON ANY COMPUTER WITHOUT NEEDING MATLAB. UPON GIVING IN THE REQUIRED PARAMETERS AND HITTING THE OPTIMIZE BUTTON, THE RESULT IS DISPLAYED BELOW AFTER COMPUTATION. ALSO, AN EXCEL FILE IS GENERATED WHERE ALL RESULTS ARE STORED FOR LATER USE BY THE SALES REPRESENTATIVES……………………………………………….21

FIGURE 7. SCHEMATIC SHOWING THE DIMENSIONS OF THE SIMULATED 2D MODEL. THE COOLING UNIT IS TO THE LEFT OF THE MODEL WITH THE INLET BEING THE COLD AIR FROM THE COOLING MODULE TO THE COLD AISLE AND THE OUTLET IS THE RETURN OF THE HOT AIR FROM THE HOT AISLE TO THE COOLING UNIT. THE 4 IT MODULE HAVE THE EXACT FEATURES WITH EACH OF THE IT MODULE PRODUCING THE SAME AMOUNT OF HEAT UNIFORMLY DISTRIBUTED WITH IN THE POROUS MEDIA CLOSER TO THE HOT AISLE……………………………………..28

FIGURE 8. SHOWS THE ARRANGEMENT SEQUENCE OF THE SERVER, COOLING AND POWER UNITS AS AN OUTPUT FROM THE OPTIMIZATION ALGORITHM. THE THIRD ROW NAMED POWER INDEX DENOTES THE CONNECTION OF THE SERVER AND COOLING UNITS TO THEIR RESPECTIVE POWER MODULE AND SIMILARLY THE FOURTH ROW NAMED COOLING INDEX SHOWS THE COUPLING OF THE SERVERS TO THE RESPECTIVE COOLING UNITS…………………………………………29

FIGURE 9. SCHEMATIC REPRESENTATION OF THE POWER AND COOLING COUPLING BETWEEN SERVER, COOLING AND POWER UNITS, ALSO GENERATED AS AN OUTPUT OF THE OPTIMIZATION ALGORITHM. THE POWER UNITS' COLOUR CODED CONNECTION IS SHOWN BELOW THE SEQUENCE AND THE COUPLING BETWEEN THE COOLING UNITS AND SERVERS ARE SHOWN ABOVE THE SEQUENCE. FURTHERMORE, THE TOP GRAPH OF C_{TOTAL} VS. *I* (POSITION) SHOWS THE NET COOLING AVAILABLE AT ANY POSITION (*I*)……………………………………..30

FIGURE 10. (A) SHOWS THE TOP VIEW OF THE IN-ROW DC, WHERE THE COLD AISLE IS AT THE BOTTOM AND HOT AISLE IS AT THE TOP. THE CIRCULATION OF THE AIR IS

ALSO MARKED IN THE FIGURE. THE SIZE OF THE ARROW GIVES A QUALITATIVE VALUE OF THE MAGNITUDE OF THE AIR FLOW RATE. (B) SHOWS AN ISOMETRIC VIEW OF THE IN-ROW DC WITH THE POSITION OF THE TEMPERATURE SENSORS MARKED BY WHITE AND BLACK DOTS. SIMILAR ARRANGEMENT OF SENSORS IS ALSO PRESENT AT THE BACK SIDE OF THE RACKS. (C) SHOWS THE SETUP CONFIGURATION (FRONT VIEW) AND THE POSITION OF TEMPERATURE PROBES THAT WERE INSTALLED WITHIN THE TEST SETUP ALONG THE COLD CHAMBER AND THE HOT CHAMBER. IN THE FIGURE S# REPRESENTS THE IT RACK AND G# REPRESENTS THE GAPS BETWEEN THE IT RACKS……………………………….32

FIGURE 11. ALGORITHMIC FLOW SCHEMATIC OF SELECTING A DX COOLING UNIT FOR A GIVEN REQUIREMENT…………………………………………………………42

FIGURE 12. ALGORITHMIC FLOW SCHEMATIC OF SELECTING A CW COOLING UNIT FOR A GIVEN REQUIREMENT……………………………………………………46

FIGURE 13. SCHEMATIC SHOWING THE IT MODULE NUMBERED IN THE DIRECTION OF X MARKED……………………………………………………………………...49

FIGURE 14. (A) COMPARISON BETWEEN THE EXPERIMENTAL AND SIMULATION RESULTS FOR AIR FLOW ACROSS THE IT MODULES. (B) COMPARISON OF MASS FLOW RATE OF THE AIR FROM THE SIMULATION INTO THE IT MODULES (RACKS) AND ALONG THE IT MODULES FOR ALL IT MODULES STARTING FROM THE COOLING MODULES (WHICH IS ALONG X AXIS)…………………………………50

FIGURE 15. VELOCITY VECTOR WITH COLOURS DEFINED BY PRESSURE (GAUGE PRESSURE) MAGNITUDE……………………………………………………...50

FIGURE 16. (A) MASS FLOW RATE INTO THE IT MODULE AT DIFFERENT INLET MASS FLOW RATE (B) MASS FLOW RATE ACROSS THE IT MODULE AT DIFFERENT INLET MASS FLOW RATE……………………………………………………………51

FIGURE 17. SHOWS THE NON-LINEAR INCREASING TREND FOR TEMPERATURE WITH RESPECT TO THE IT RACK POSITION AT DIFFERENT AIR FLOW RATES FROM THE COOLING UNIT DECIDED BY THE RPM OF THE FANS IN THE COOLING UNIT FOR TWO DIFFERENT IT LOADS…………………………………………………..51

FIGURE 18. GEOMETRY FOR THE 2D SIMULATION OF IT RACKS AND COOLING UNIT. THE IT RACKS ARE SEPARATED FROM EACH OTHER BY POROUS GAPS REPRESENTED IN DEEP BLUE COLOUR AND THE COOLING UNIT IS TO THE EXTREME RIGHT. THE OVERALL AIR FLOW DIRECTION IS SHOWN BY THE ARROWS. THE BOTTOM AISLE IS THE COLD AISLE FROM WHERE THE COLD SUPPLY AIR IS SUCKED IN BY THE SERVERS AND THE TOP AISLE IS THE HOT AISLE FROM WHERE THE HOT AIR IS SUCKED BY THE COOLING UNIT………………………56

FIGURE 19. SHOWS THE PRESSURE CONTOUR ALONG WITH THE VELOCITY VECTORS OF THE AIR FLOW IN THE MODEL. IT IS SEEN THAT A HIGH-PRESSURE ZONE IS CREATED IN FRONT OF THE COOLING UNIT FANS AND PRESSURE SLOWLY DROPS ALONG THE COLD AISLE WHEREAS THE PRESSURE SLOWLY INCREASES ALONG THE HOT AISLE FROM RIGHT TO LEFT……………………………………..57

FIGURE 20. SHOWS THE LOCAL DIFFERENTIAL PRESSURE (COLD AISLE PRESSURE – HOT AISLE PRESSURE) AT THE RESPECTIVE POSITION OF THE GAPS FROM THE COOLING UNIT FOR A MAXIMUM VELOCITY DEFINED FOR THE FANS IN THE

cooling curve as (a) 3 m/s, (b) 5 m/s and (c) 10 m/s. From the plots, it can be observed that the limiting position of the gap for the differential pressure to be positive is 3. After the third gap, the differential pressure is negative which means that the hot air will leak into the cold aisle which is undesirable as mentioned earlier.58

Figure 21. (a) Shows the temperature in the cold aisle in front of every rack for 3 different cold aisle width 1200 mm, 1400 mm and 1500 mm. (b) Shows a schematic of the geometry with the position of the cooling unit (CU) and IT racks (R1, R2, R3 and R4)…………………..60

Figure 22. Plot showing the regression fitting between the dimensionless temperature and the dimensionless parameters for different system size of the modular DC. It can be seen that the curves for different system sizes namely 2 IT rack, 4 IT rack and 5 IT racks the relationship curve increases in slow in the order of the system size, with 2 IT rack system having the minimum slope and 5 IT rack system having the maximum……………………………………………………...66

Figure 23. Shows the accuracy of fitting a second order polynomial function to the dimensionless numbers. The R-squared value for the fit is 0.92……………………………………………………………………67

Figure 24.The redundant power connection generated from the optimization program. P_1, P_2, P_3 are the 3 different power modules.

EACH LOAD UNIT IS CONNECTED TO TWO POWER SOURCES INDICATED BY THE RESPECTIVE COLOURS OF THE POWER MODULES……………………………72

FIGURE 25. ARRANGEMENT OF 2 COOLING AND IT SERVERS WHERE THE AIR FROM THE MIDDLE COOLING UNIT CAN REACH BOTH DIRECTIONS (LEFT AND RIGHT). THE SITUATION SIMULATES A REDUNDANT COOLING SYSTEM WHERE ONE OF THE COOLING UNIT (TO THE EXTREME RIGHT) HAS FAILED. SO, WITH JUST 2 COOLING UNITS, UNDERSTANDING THE DISTRIBUTION OF THE AIR FLOW FROM THE MIDDLE COOLING UNIT TO THE LEFT AND RIGHT IS ESSENTIAL IN DESIGNING REDUNDANT SYSTEM…………………………………………………………………...72

FIGURE 26. REPRESENTATION OF THE ABOVE CONFIGURATION IN AN EQUIVALENT CIRCUIT FORM. HERE, R REPRESENTS RESISTANCE, WITH SUBSCRIPTS REPRESENT THE FOLLOWING: S= IT RACK, L= LEAKAGE, F=FRONT, B=BACK AND C=COOLING…………………………………………………………………73

FIGURE 27. VARIATION OF NORMALIZED AIR FLOW RATE (NORMALIZED TO TOTAL AIRFLOW RATE FROM THE COOLING UNIT) WITH THE RATIO BETWEEN DIFFERENT RESISTANCES. SHOWS THE VARIATION OF AIR FLOW RATE TO THE LEFT AND RIGHT SIDE OF THE COOLING UNIT AND THE TOTAL AIR FLOW IT WITH THE (A) RATIO OF RESISTANCE OF THE COLD AISLE OVER RESISTANCE OF THE IT RACKS, (B) RATIO OF RESISTANCE OF LEAKAGE OVER RESISTANCE OF IT RACKS, (C) RATIO OF RESISTANCE OF COLD AISLE OVER RESISTANCE OF LEAKAGE………75

LIST OF TABLES

TABLE 1. ONE OF THE SEVERAL EXPERIMENTS ON THE IN-ROW IN-HOUSE DC. THE EXPERIMENTS WERE CARRIED OUT TO UNDERSTAND THE EFFECT OF IT LOAD AND THE AIR FLOW RATE FROM THE COOLING UNIT. HENCE, THE TWO SETS OF EXPERIMENTS WITH CONSTANT IT LOAD BUT VARYING AIR FLOW RATE WAS CARRIED OUT AND THE TEMPERATURES AT SEVERAL POINTS WERE RECORDED IN ORDER TO UNDERSTAND THE RELATION. SET POINT TEMPERATURE: 19 °C.....33

TABLE 2. SHOWS THE TEMPERATURE DISTRIBUTION AT DIFFERENT IT RACK POSITION INSIDE THE TEST SETUP FOR (A) 11.99 KW, (B) 14.16 KW. IT CAN BE SEEN THAT WITH LOWER AIR FLOW RATE (50% RPM) FROM THE COOLING UNIT THE TEMPERATURE OF THE IT RACKS 3 AND 4 INCREASE SIGNIFICANTLY. HOWEVER THE TEMPERATURE OF THE IT RACKS 1 DECREASES WITH LOWER AIR FLOW RATE. INCREASING TREND IN THE TEMPERATURE AT THE GAPS CAN ALSO BE OBSERVED. 35

TABLE 3. OPTIMIZED PLACEMENT OF THE POWER, COOLING AND IT MODULES 54

TABLE 4. THREE DIFFERENT DESIGN SUGGESTIONS FROM THE DESIGN OPTIMIZATION TOOL FOR CW TYPE COOLING UNITS ARRANGED ACCORDING TO INCREASING COST...78

ACKNOWLEDGEMENTS:

As I recall the time from the day of arrival to McMaster University till today, I remember a quote from a famous man, "You cannot connect the dots looking forwards, you can always connect the dots looking backwards". All the doubts about the decisions that I had in the beginning about joining McMaster University have disappeared and I am most certain that it's one of the best thing that has happened in my life. I would not have been able to ground myself and carry on without the mentorship of Dr. Ishwar K. Puri, who was there for me not only when technical problems surfaced but equally for the often hidden emotional ones. I was already familiar with most of the members in the lab from my internship earlier. This helped me settle in soon. Dr. Suvojit Ghosh, who started as a daily mentor and ended up as a friend. As Dr. Ghosh soon became the director of CIRC I was mentored by Dr. Abdel Rehman and followed by Dr. Souvik Pal who is one of the most hard working and knowledgeable person in the lab. I have learned a lot under their mentorship. I am also grateful for the advice and support of Dr. D. Down, Dr. G. Badawy, Dr. T. Davidson. Hosein, Jenifer and Sahar were in my core friendship group. Their support was essential for me success. I learned that small discussions around the lab, when in a perfect environment, will incubate and grow into great projects. Last but certainly not least, thanks to my family who had to tend to me on all the nights when my experiments failed. Without them being there, it would have been impossible to carry on the journey.

LIST OF ABREVIATIONS

DC	Data Center
MDC	Modular Data Center
CW	Chilled Water
DX	Direct Expansion
CRAH	Computer Room Air Handler
TWh	Terawatt hour
CC	Cooling Curve
CU	Cooling Unit
IT	Information Technology
CFD	Computational Fluid Dynamics
HVAC	Heating, Ventilation and Air Conditioning
ANN	Artificial Neural Network
PUE	Power Usage Effectiveness
CPU	Central Processing Unit
ILP	Integer Linear Problem
GUI	Graphics User Interphase
ASHR AE	American Society of Heating, Refrigerating, and Air-Conditioning Engineers
RPM	Revolutions Per Minute
COP	Coefficient of Performance

IPLV	Integrated Part Load Value
KPI	Key Performance Index
CDR	Component Data Repository
RMSE	Root Mean Square Error
TCO	Total Cost of Ownership

1. Introduction

The physical infrastructure that drive the internet relies on the operation and management of hundreds of DCs scattered around the world. The actual growth of the DCs started during the dot-com bubble where companies needed fast and non-stop internet connection to deploy systems and carry transactions. Essentially, a DC is a system capable of transmitting, receiving, processing and storing huge amounts of data simultaneously to anything that is connected to the internet, such as Google DC and Facebook DC. These are spread across the world for faster speed and to have a decentralized system to protect against failures and localized damage. The building blocks of such data DCs are server units, which run 24/7 to provide continuous service also known as the IT (Information Technology) equipment and is the most important component in a DC. The safety and security of these IT equipment being critical, the DCs include redundant and backup power and environmental control systems and several other security devices. One of the concerns associated with DCs is that, they have been consuming increasingly enormous amounts of power from the last decade. A typical DC can consume as much power as a small city which is as much as 25,000 households or 100-200 times the power of a standard office. The exponential increase of cloud services in recent years has triggered a like growth in DC facilities increasing the power consumption of data centres to about 2% of the world's total power [1][2][3]. A recent study has found that the DCs worldwide consumed 270 TWh of energy in 2012 and this consumption had a Compound Annual Growth Rate (CAGR) of 4.4% from 2007 to 2012. Hence the energy efficiency of a DC has attained high importance in the recent years. Another major problem associated with DCs is that they tend to get hot. The servers and processing hardware generate heat as a result of operating and processing data. Since, these servers and hardware are densely packed in a room, the heat generated is not able to dissipate easily, leading to increase in the temperature of the room. It has been well studied and understood that running the servers at high temperature negatively affects its processing power and also ages the servers faster, as a result the servers show frequent failures at early stages of its operation. The high temperatures can also lead to data corruption or complete failure of the hard drive as mentioned in this white paper [4]. Hence, to avoid all these troubles cooling systems are installed in a DC to keep the temperature under control. A significant portion of the total energy consumption in a DC goes towards cooling systems. Almost half of the power required to run IT equipment is consumed for cooling them. In other words, for every 1 kW of energy consumed by servers, 0.5 kW of additional power is required for cooling, fueling the high energy consumption by DCs [5][6][7] . To that end, in an effort to optimize energy consumption in a data centre, more research is being undertaken.

Different DCs implement different types of cooling techniques depending on what is suitable for them. One of the most common form of cooling systems implemented in typical DCs is raised floor cooling. Here, the cold air is supplied to the servers from under the floor. Hence, the floor is raised to allow air flow through it and parts of the floor is perforated to allow the cold air flow to come out at specific locations. The cooling units themselves, referred as CRAH (Computer Room Air Handler) unis are at the corners and edges of the DC room. This type of cooling arrangement gives space to increase the capacity of the DC in the future since more CRAH units can be installed and appropriately more perforated tiles can be placed on the floor for providing cold air to the additional IT equipment. But raised floor cooling systems require high initial investment to have provisions for future expansion. Also, these cooling systems are inefficient, which is why the cooling systems are excessively over designed to overcome the inefficiencies which in turn leads to increased operational costs in addition to the initial costs. With the constantly improving cooling systems, there is an observable trend of DCs shifting from air based cooling to liquid based cooling. Not only it is highly efficient, but also it gives a greater controlling capability which can be as granular as to individual server. Where as in traditional raised floor cooling system, due to minimal controllability, to neutralize a hotspot in a specific region, the whole cooling system has to be ramped up causing energy wastage.

The liquid based cooling systems for DC has its own set of limitations. It is a well-known fact that liquid (generally water) and powered electronics are dangerous together. In an event of damage to the liquid (water) cooling systems inside a DC, the liquid could damage the servers by shorting its internal circuits. This can lead to huge loss of financial resources and long DC downtime. A DC downtime is when the services provided by the DC have to be stopped in order to carryout maintenance works or in the event of an unplanned failure. On an average, 1 minute of DC downtime in US costs about US$7,900. And in liquid damage type of failures, the downtime can range from days to weeks.

Another type of cooling system that has been gaining popularity recently is in-row type cooling system. In these type of cooling system, the CRAH units (similar dimensions of that of an IT rack) are placed beside the IT racks. The cold air flow from the cooling unit is directly thrown into the room, rather than coming from under the floor as in the raised floor cooling system. One of the obvious advantage of these type of cooling units is that it reduces construction cost, by not having to construct a raised floor.

With better proximity to the source of heat generation the efficiency of cooling systems increases. Hence in terms of efficiency, the in-row cooling unit is more efficient than the raised floor cooling system, but less efficient than the liquid cooling system. Also, being an air cooled system, there are no risks of water damage to the servers since the cooling system is outside the IT racks.

With the constant efforts towards increasing efficiency and reducing costs, new type of DC designs are continuously being developed. A modular DC, is a DC which is completely confined within a box, so that it can be easily deployed with little construction. Our research is focused on these modular DC. There are several different designs for modular DC, one of which is using the IT racks as the boundary or the box for the DC. The IT racks are arranged in a row stacked side by side and the cold air flow is supplied into the box by modular cooling units (of similar dimensions to the IT racks) as shown in **Figure 1**. These type of modular DCs are suitable for small businesses and companies who cannot afford to set up bigger raised floor DCs. Also these modular DCs are much faster to deploy and can be expanded as the business grows because it is modular similar to Lego blocks.

Figure 1. Actual image of a modular DC in-house (Industrial partner - Cinnos). To the extreme right (blue led) is the power rack, which feeds power to the cooling unit (next to it) and the IT racks (red led). The cold air is supplied to the IT racks through the space between the front glass cover and front of the servers.

For different types of data centres, there has been research to understand the interaction between DC's different components, for example the balance between cooling and IT capacities and their respective power consumption [8][9]. The authors in [8] developed steady state energy and exergy destruction models for the modular DCs and have reported that augmenting DX (Direct Expansion) cooling with evaporative and free air cooling can save up to 38% of the cooling energy. With the help of self-developed software called EnergyPlus, they are able to model modular DCs and optimize the energy consumption of the cooling system based on the climatic condition of the region. The tool uses a heat balance method for heat transfer calculations which is more accurate compared to other methods such as weighting factor approach. They have also mentioned that, there is a trade-off between IT equipment and HVAC power and a balance temperature for the facility has to be maintained in order to achieve optimal results. This in turn aids in the understanding of energy consumption and losses within the DC space (e.g. overheating of equipment, air recirculation, and downtime), eventually leading to energy optimization. Similarly, computational fluid dynamics (CFD) simulations have been implemented to understand the temperature and velocity profiles associated with raised floor data centres [10][11][12]. In [10] they are trying to solve the inefficiencies inside a raised floor DC which is the non-uniform distribution of cold air flow in front of an IT rack. Recirculation (the mixing of hot air and cold air) causes non-uniformity and increase in the intake air flow temperatures in front of some of the servers. Some servers will receive cold air (at the bottom of the rack) where as others will receive hot recirculated (at the top and edge of the rack) air in front of the server inlet, which makes the temperature go above the ASHRAE recommended limit (27 °C). Simply increasing the amount of cold air flow rate pumped and reducing the set point temperature of the CRAH units would be very inefficient because it will make excess cold air to pass through the bottom servers. Hence, the temperature distribution, air flow characteristics and thermal management of DC racks array are predicted and evaluated for the different arrangements and different configurations of the CRAC units to derive useful performance indices from the study. Their conclusion was that the hot air recirculation, cold air bypass and the measurable performance indices of the racks strongly depend on the rack's location in the racks array. They also pointed that using cold aisle containments improves the thermal performance of a DC. Most of the CFD simulations on DCs study the thermal behaviour of conventional raised floor DCs [13][14][15][16][17]. In [16] they quantitatively analyzed the air side economizers for their energy saving potential for these modular DCs. A detailed cooling load estimation process was established and annual cooling rate simulations were carried out using various air side economizers which resulted in 76-99% cooling coil load savings and 47.2% to 67.2% of energy savings. In another study [18] which compared the efficiency of air side economizer to water side economizer for the DCs and concluded that the air side economizer is more efficient. However, in terms of installation costs, the water side economizers were preferred. Lee and Chen [19] analyzed the energy-saving potential of air side economizers in data centers located in various climate zones and found that the best climates for the air side economizer is mild climate with moderate humidity. The humidification energy in the cold and dry climates offsets the energy savings by the air side economizers. Also reducing the cold air temperature by 2 °C reduced the energy savings by 2.8%-8.5% was also mentioned. However efficient and suitable design methods for air side economizers are crucial for energy savings and are still being developed. Even though CFD results give better accuracy and more details, the runtime of the simulation can be weeks depending on the complexity stretching the time for deployment of a data centre facility.

The rapid simulations implement reduced order governing equations [17], providing shorter run times and reasonable accuracy compared to CFD equivalent. Hence, rapid simulations are often beneficial for modelling overall DCs to gain preliminary design and behaviour information. CoolEmAll, [13] a project developed by a collaborative effort of researchers and industrial partners in Europe, focuses on complete CFD simulation as well as rapid simulation tool kit to better understand DCs. Their software is based on OpenFoam and is freely available [20] to use and modify. The CoolEmAll project also allows for real time mapping of temperature within a data centre. This provides a deeper understanding of data centre performance and energy optimization at various IT loads. Jim Gao, at Google used Artificial Neural Network (ANN) for DC energy monitoring [21]. Here, an algorithm is trained by previously collected power, cooling, IT readings and so on summing up to a total of 19 different parameters, to predict the Power Usage Effectiveness (PUE) of the DC within a range of 0.4% error for a PUE of 1.1. A possible drawback with Neural Networks is the need of large amounts of data before accurate predictions are achieved. The time required for predicting temperatures can be reduced by using actual measurements to supplement the calculations by a model. Hendrik F. Hamann et al. in their paper discussed about using more direct measurements from sensors to accelerate energy predictions [22].

Power management is becoming an important issue with more DCs being set up each year. A typical 10 MW DC would cost around 8 million dollars annually as electricity bills just to run the servers, and an additional 8 million dollars for the energy costs of the cooling system and other auxiliary units [23]. This high power consumption indirectly leads to huge amounts of carbon emission annually. Hence it is essential to understand the power consumption behaviours of servers. Therefore the author in [23] has developed a software 'mantis' which can provide power estimates with high accuracy. In a non-intrusive way 'mantis' can provide fast and accurate predictions of power consumptions in power systems. This can identify the critical components of energy consumption and help in developing future energy aware solutions.

Another aspect of data centres energy optimization is centred on workload management [7]. Workload is defined as the total requests made by users and applications of a system. In a typical DC, most of the time the servers are only lightly loaded, except the very rare times, where the workload reaches close to the full capacity. This means that most of the time, all the servers have very low utilization, but consume almost 70%-90% of as much energy as a fully loaded server would consume. But with intelligent workload management system, the workload can be reassigned or redistributed to conserve energy. Most of the DCs don't implement workload management system of a third party because of the fear of data security. But very simple and effective solutions can be developed in-house for workload management. For example, certain number of tasks on some servers can be reassigned to other servers and made to work at higher utilization so that the previous set of servers can be switched off which would lead to energy savings. Even more cost saving can be achieved by using prediction algorithms to predict the workload ahead of time, so that the maximum required energy cost can be cut down. This in turn, will lead to better server life and decreased rate of data corruption or server failure. Experiments have highlighted the relation between server performance and power consumption [24][25][26]. They have shown that server computation to power consumption curve is highly disproportional. The authors in [25] identified models that can be applied in the design of customized energy-aware controllers that dynamically adjust CPU frequency suitable for the application-specific workload patterns. According to their experimental results, customized controllers may outperform general ones both in terms of reachable server performance and power saving capabilities. Therefore strategic workload distribution among servers can conserve energy. Some of the research are concerned with dynamically controlling the number of active servers to get optimized energy profiles [26][27]. Others show that re-distributing the work load among several servers or nodes conserves energy [28][29].

1.1 Problem Definition

1.2.2 Generic Modular DC configuration

Modular DCs have attracted considerable attention because of their excellent stability, scalability, and economic feasibility. A schematic of generic configuration of a modular DC system is described in **Figure 2**. The different components and the direction of air flow are mentioned in the figure. The grey boxes are IT racks and the blue box is the cooling unit. The black arrows show the cold air flow direction along the racks, which after being drawn in by the servers, become hot and are sent back to the cooling unit as shown by the red arrows. Before making a preliminary design for a modular DC, the type of servers, number of racks and total number of servers in each rack are specified by the client. The type of cooling (chilled water or DX system) is also specified by the client. There is flexibility in choosing any of different commercially available cooling units within that selected category of cooling units.

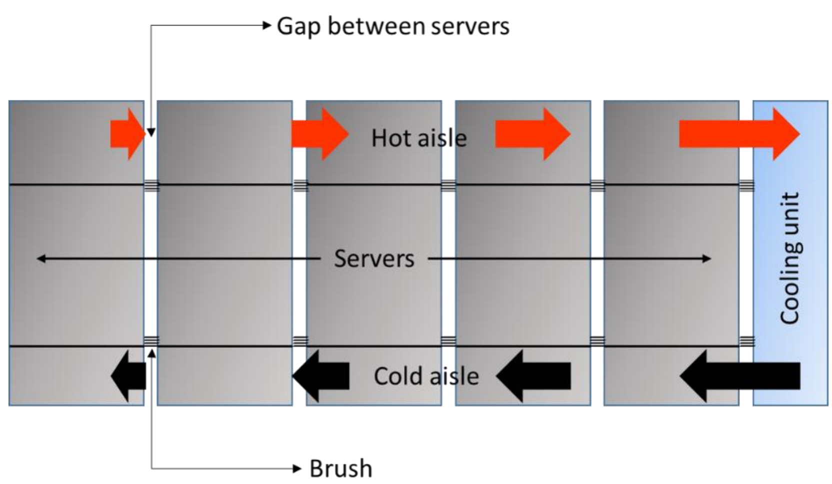


Figure 2. Schematic representation of the arrangement and air flow through a modular DC. The grey boxes represent IT racks and blue box represents cooling unit. The cold air is supplied from the front of the cooling unit (also called as cold aisle or cold chamber) represented qualitatively by the black arrows where the size of the arrow represents amount of cold air entering each rack. Similarly the red arrows represent the hot air coming out from the servers and moving to the cooling unit (through the hot aisle or hot chamber). The gaps between any racks is partially closed with the use of the brush, so that it prevents cold air from mixing with hot air.

1.2 Scope of Study

1.2.2 Design Cost Optimization

The type of DC that is discussed here is modular DC. Since, these type of DCs are newer than other traditional DCs like raised floor DCs and in-row DCs, it lacks systematic study and understanding of the temperature distribution, power consumption and so on. Therefore, it provides a greater opportunity to improve several aspects of the modular DC like more compact module, power and thermal optimization. Here, the focus is mostly on the design aspect of these DCs. Generally, the design of a DC starts with the bidding process where companies perform cost estimation and preliminary designs to provide to clients in an effort to be awarded the projects. In a typical bidding scenario as shown in **Figure 3**, the bidders, generally sales representatives, strive to provide an accurate bid, which require preliminary analysis and the involvement of engineers and designers, finally communicating it back to the clients. Because of the limited study available on modular DCs, the engineers and the designers have to go discuss through several iterations of the design to have a cost efficient design based on their experiences and not concrete study. This chain of communication between the sales representative, engineers and clients, consumes valuable engineering time, leading to cost expenditures on the part of the bidding company. There has not been any organized study about the design of modular DCs. Hence the engineers, to keep the DC failures and downtime at bay usually over design the system leading to increased design cost. This part lacks in terms of methods and research which could help make better decisions in lesser time, saving valuable engineering time and cost expenditures for the company.

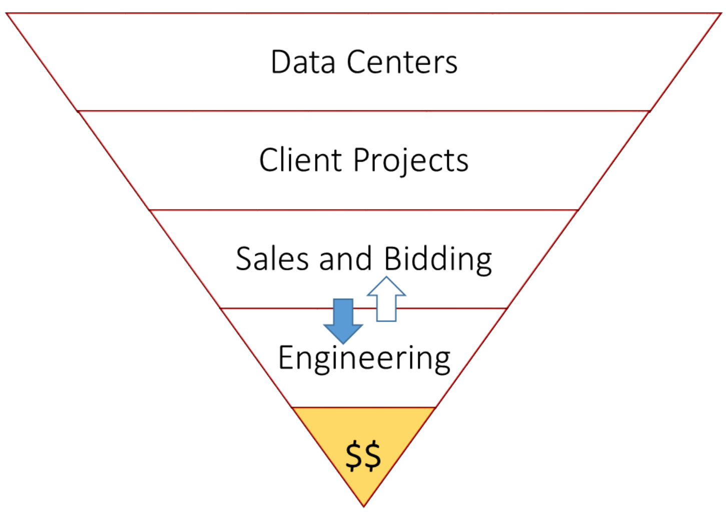


Figure 3. Schematic of a typical bidding scenario. The most time consuming part is the cycle of communication process between the sales people and the engineers to finalize the preliminary design for conveying it to the clients.

There are several companies which cater to the needs of DCs. And each company has several models of cooling units, power units, racks, etc. In addition to that often some models are stopped from production and new models of equipment are added. This causes a problem for the sales representatives and the engineers in choosing the best components for the modular DC design. Not only they have to constantly track the equipment in several companies but also have to keep updating the designs to match the current available equipment. This consumes a lot of time for both the sales representatives and the engineers. Hence, having a software which has a built in data base of the different models of components and generate design for the preliminary analysis would be highly appreciated by both the bidders and the engineers.

Herein we propose a new type of modular DC design optimization tool, which focuses on the sales representative as the end user. In the hand of the bidders, this will help optimize the project estimation based on the client requirements without going through the long chain of communication process with engineers and designers. This in turn will help save useful engineering time as schematically represented in **Figure 4**. As shown in **Figure 5** with the help of input parameters which are designed based on the need of the client, the tool can generate multiple designs with rankings involving minimum cost. Also, using this information a bill of materials can be simply developed. Specifically, the tool will generate a design composed of commercially available units along with the bill of materials.

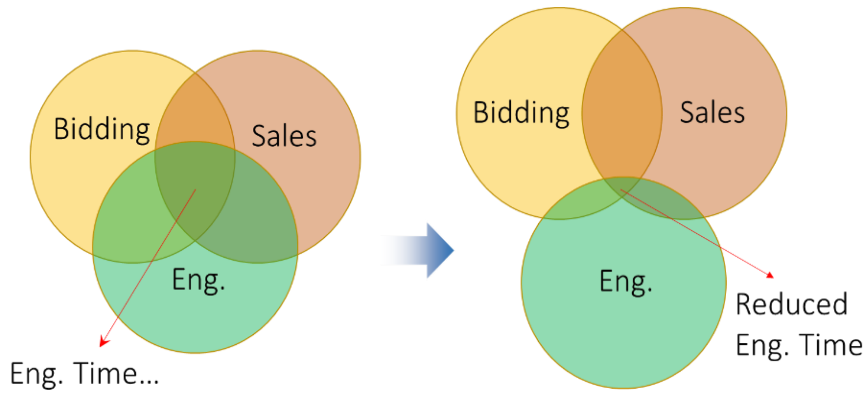


Figure 4. Schematic showing the involvement of the Engineering team with sales team before and after the optimization tool implementation. This implies that the tool is tailored to save engineering time allowing them to use it on already approved projects.

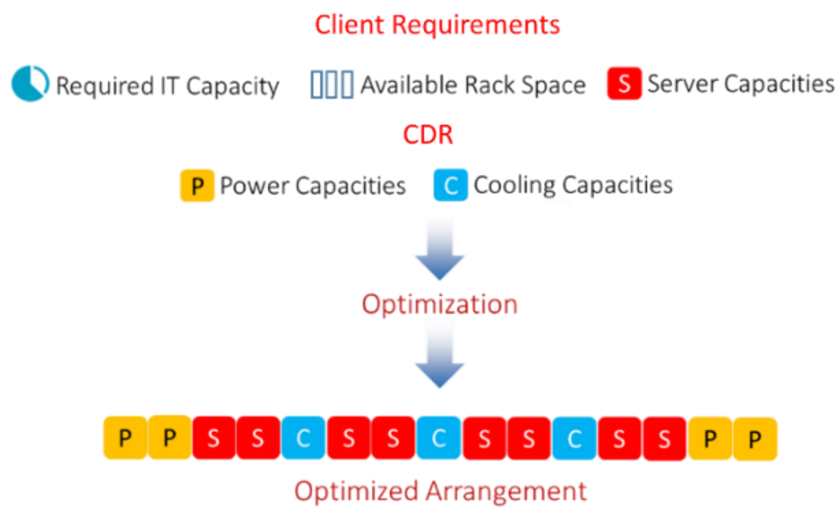


Figure 5. Schematic showing arrangement of the power, cooling and IT racks by the optimization tool. With the client inputs and with the help of the built-in inventory of commercially available components the algorithm can generate multiple designs ranked in the order of the cost for the designers to choose from. From the ample types and configuration of power, server and cooling modules, selecting and arranging them manually in an optimum configuration can take weeks. With the tool, a rather time consuming process of finding the best arrangement can be achieved simply.

1.2.2 Redundant power configuration

Redundant system in DCs are very important to respond to unplanned failures and maintenance activities. Since, the cost of downtime of a DC is so high, usually the DCs are excessively overdesigned to account for these failures and maintenance activities. But, because designing these redundant systems manually is a time consuming and difficult task for the engineers, often times they are not able to find the optimum design they simply end up overdesigning the system. There are several different levels of redundancy that are implemented in DCs depending on the tolerance of the facility to downtime. The most common form of redundancy implemented in the small MDCs is the N+1 redundancy. N+1 redundancy means, there are enough spare cooling and power systems (which are online or ready to be online) in case any one of the cooling or power system fails.

Here we have developed an algorithm which can generate an optimum N+1 redundant configuration for the power system in a very short time, hence saving engineering time and limiting over designing leading to cost savings. Developing redundant configurations designs for cooling system is much more complicated than power systems because of the limitations in experimental setup and complexities in understanding fluid and thermal behaviour which has been discussed in details in the results section.

2. Methodology

2.1. Inefficiencies of a modular DC

A DC consists of IT, cooling and power systems. The heat generated by the IT and power equipment are isolated and cooled by the cooling units and the power requirements of the IT and the cooling units are fulfilled by the power systems.

Hence, the system is highly coupled. A good understanding of this coupling is not known. Therefore, due to lack of understanding we have to compensate with higher capacity and number of cooling units. In this study we will explore the coupling between the cooling units, IT and the heat to better design and improve the energy consumption which will be translated into cost savings.

The several inefficiencies within the MDC leads to the temperature gradient inside the MDC. The inefficiencies inside an MDC is due to air flow mismatch between the IT racks and the cooling unit. Typically, the total airflow rate from commercially available cooling units is less than that drawn in by the servers, when the capacity of the cooling unit and power consumption of the servers are matched. Consequently, the confined back chamber is always at a higher pressure than the front. Through the several gaps present between the IT racks and in the IT racks, the pressure difference pushes the air flow in the hot aisle to leak into the cold aisle. This hot air mixing increases the temperature of the cold air flow from the cooling unit, leading to development of temperature gradient.

The first method that we used to solve the problem is using an optimization algorithm. In short, for the design this algorithm minimises the excess cooling and excess power in an in-row modular DC.

2.2. Cooling range

The IT rack, where the temperature in front of it is above the desired threshold (ASHRAE – 27 °C) is considered as high temperature and appropriate measured must be taken to bring the temperature under control. But if we limit the number of IT racks after a cooling unit to a number where the temperature is within the desired threshold, the problem can be solved. This is called as the range of the cooling unit. The increase of the temperature in front of the servers from the set point to a higher value as a function of distance from the cooling unit is referred as cooling curve here. This can be found out by experiments or CFD simulations and subsequently from that derive relation between parameters of cooling unit and IT racks. The characteristics of this relation is not linear and is a function of several different parameters as discussed in further sections.

2.3. Algorithm for minimizing excess power and cooling

The aim of the optimization algorithm is to obtain an arrangement sequence of IT racks, cooling units and power units in such a way that it uses the least capacity of cooling and power units. For every optimization problem there are (1) Parameters, (2) Decision variables, (3) Problem constraints and (4) Objective function. The decision variables for modular data centre design algorithm are binary values and hence fall under the category of ILP (Integer Linear Problem) optimization.

Having formulated the constraints and the objective function for the optimization problem at hand, MatLab 2014b was used to define it in code. The inbuilt optimizer function of MatLab intlinprog (Integer Linear Programming) was used to carry out the optimization. The algorithm is solved in blocks. For the first position it determines which unit (IT, cooling or power) and of what type (capacity) should be places in that block. The problem is defined as a binary linear optimization, and thus the range of the integral values of the variable are limited to 0 and 1 where 0 means that an IT rack, or cooling unit or power unit is not selected and 1 would mean that it is selected. Finally, the optimization code in developed into an executable application, where the sales representatives can enter the required details and upon running the application the result is displayed as well as an excel file containing the detailed information of sequence configuration is generated for easier interpretation as shown in **Figure 6**.

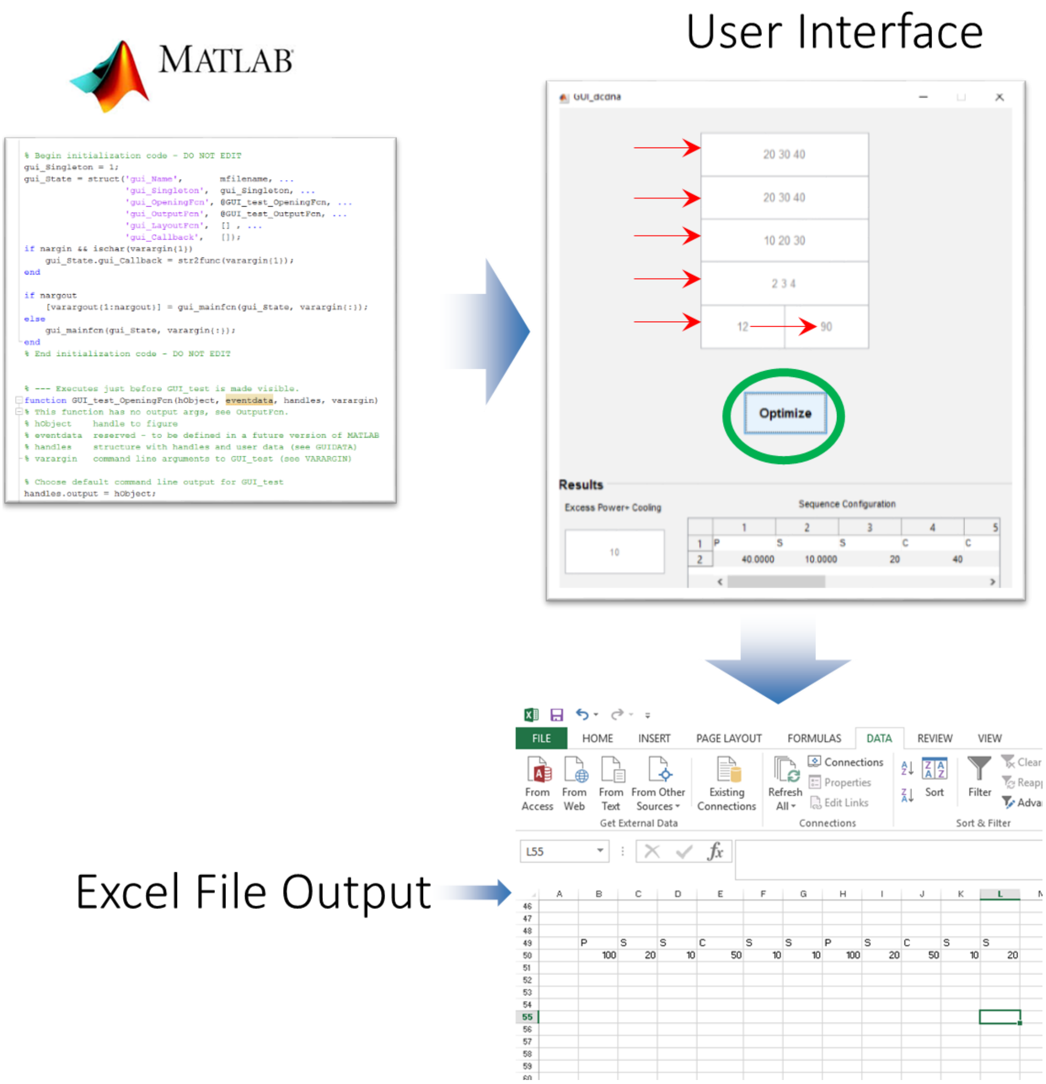


Figure 6. GUI of the optimization algorithm for easy access for the sales representatives. The algorithm is packed in an executable package which can be installed on any computer without needing MatLab. Upon giving in the required parameters and hitting the optimize button, the result is displayed below after computation. Also, an excel file is generated where all results are stored for later use by the sales representatives.

Currently, no research has been done to study the cooling profile or pattern of modular cooling units. Most of the experiments or simulations cater to the raised floor type cooling system in DC. For modular cooling systems its capacity can be described as a continuous function which will depend on the position and capacity of the cooling unit and the IT racks. To understand this better CFD simulations were performed and later the results were validated with the experimental results from an actual in-house modular DC. The purpose of the simulations was to obtain a function which relates the cooling capacity at any block which is at a particular distance and the IT module it is coupled with.

2.3.1 Parameters defined in algorithm

The following parameters are used for optimization

Decision Variable: $x_{ijk} \in \{0,1\}$ provides full configuration information with 0 and 1 representing no selection (i.e. x = 0) and selection (i.e. x = 1). This is also referred as the decision variable, where $i =$ The position in the arrangement, $i = \{1, 2, 3, ...\}$, $j =$ The component at that position, $j = \{1, 2, 3\}$ where 1, 2 and 3 represent the power, cooling, and server modules respectively. $k =$ The capacity type of that component, $k = \{1, 2, 3, ...\}$.

So, for example, x_{123} represents (Boolean) the value of a cooling unit (j =2) which is of 3rd type (k = 3) capacity (for instance, 60 kW cooling capacity) at the first position in the arrangement (i = 1) and the value of x_{123} can be either 0 or 1 (1, if it's true, else 0).

Capacity: P_k, C_k, S_k represent the capacity of the respective modules (Power, Cooling, IT respectively) at i^{th} position and of k^{th} type, where $k \in K$.

For example, $P_{23} = 40$ implies that there is a power unit placed at the second position, which is of third type and has a capacity of 40 kW.

Length: Total length of the arrangement or sequence (in terms of the number of units) is defined by $\max(i) = L$ or $\sum_{i,j,k \in I} x_{ijk} = L$. (1)

2.3.2 Problem constraints

Constraints used in the optimization algorithm are as follows:

- The sum of all the server capacity should be equal to or greater than the total IT power capacity (IT_{total}) defined by the client.

  Mathematically, $\sum_{i,k \in I} x_{i3k} * S_k \geq IT_{total}$ (2)

- The sum of the total power unit capacity should be at least equal to the sum of the total IT power and power consumed by the cooling unit, which is defined as

$$\sum_{i,k \in I} x_{i1k} * P_k \geq \left(\sum_{i,k \in I} x_{i3k} * S_k + \sum_{i,k \in I} x_{i2k} * C_k / cop_k\right). \quad (3)$$

where cop_k is defined as the coefficient of performance =

$$\frac{C_k}{\text{maximum power consumption}}$$

The amount of heat that can be removed from the IT modules depends on the cooling module. But to have the temperature in front of the servers within ASHRAE guidelines, other parameters like distance of the IT module from the cooling module, air flow rate from the cooling module, pressure drop across servers, etc. have to be regulated. Due to the complex relations between the above mentioned parameters, it's very difficult to find an optimum location for the cooling module with a wide variety of choices of cooling unit in hand. Hence, the cooling modules are generally over designed to cover for the complexity of finding the optimum location. Here some of the above mentioned parameters and their interconnectedness have been explored to cut out the overdesigning of the cooling modules. The distance of the IT module from cooling module affects its temperature significantly. Therefore, a relation between the distance from the cooling module and the cooling effectiveness is developed using CFD simulations and experiments. This relation is expressed in terms of a continuous function and is referred to as 'Cooling Curve' here.

A bell curve or standardized function is described mathematically as: $\frac{1}{\sqrt{2\pi}} e^{-\frac{(x-\mu)^2}{2\cdot\sigma^2}}$

where x represents the distance from the mean (cooling unit), μ represents the mean (position of the cooling unit), σ represents the standard deviation of the distribution.

The amount of cooling that is made available by a cooling unit at any position is equal to the area under the "bell curve" at that position for that cooling unit, which can be described by the error function (erf), where the total area under the bell curve represents the total capacity of that cooling unit. Considering there will be multiple cooling units in a data centre configuration sequence, the cooling available at any position will be equal to the sum of areas' (area under the cooling curve) of all cooling units at that position which is represented in **Figure 9**. Thus, any IT rack is allowed access only to the amount of cooling present at its location (e.g. for i = 12, cooling available is sum of all cooling curves areas between 11 and 12).

The area under the curve between two positions a and b can be expressed as follows:

$\int_a^b \frac{1}{\sqrt{2\pi}} e^{-\frac{(x-\mu)^2}{2\cdot\sigma^2}}$, where a and b are the minimum and maximum distance from the mean (cooling unit).

Which can also be expressed as (in terms of erf) follows:

$$\frac{erf\left(\frac{b-\mu}{\sqrt{2\cdot\sigma^2}}\right) - erf\left(\frac{a-\mu}{\sqrt{2\cdot\sigma^2}}\right)}{2}.$$

$\boldsymbol{C_{ilk}}$ aggregates the cooling capacity information for each cooling unit at position $\boldsymbol{i}$ at every possible other position in the array. These other positions are represented by $\boldsymbol{l}$. For each type of cooling unit or type $\boldsymbol{k}$, $\boldsymbol{C_{ilk}}$ is mathematically represented as

$$C_{ilk} = \frac{C_k}{2} \cdot \left(erf\left(\frac{|l-i|}{\sqrt{2\cdot\sigma^2}}\right) - erf\left(\frac{||l-i|-1|}{\sqrt{2\cdot\sigma^2}}\right) \right) \tag{4}$$

With this knowledge it is possible to describe a constraint which ensures that the server capacity at any position is less than the aggregate cooling capacity found at that position. This can be formulated as

$$S_k \cdot x_{l3k} \leq \sum_{i,k \in I,K} C_{ilk} \cdot x_{i2k}, l = \{1 ... L\} \quad (5)$$

2.3.3 Objective function

Modularity is important for small scale data centres because it gives the freedom to scale the system as per necessity, which cuts down on initial costs. In addition, it is much simpler and cost effective to just replace smaller faulty parts rather than larger ones, which negatively impact downtime. Hence, in modular systems, often times, systems are over designed to preserve the modularity. Limiting the overdesign through optimization can further cut down on costs.

Therefore, the optimization objective parameters are 1) over-designed cooling (ΔC) and 2) over-designed power (ΔP). The over-designed power is defined as the difference between the total power installed and total IT installed. Similar method is used for over-designed cooling calculation.

The cost function for this algorithm is minimizing ($\Delta P + \Delta C$)

where $\Delta P = \sum_{i,k \in I} (x_{i1k} \cdot P_k) - \sum_{i,k \in I} (x_{i3k} \cdot S_k)$, (6)

$$\Delta C = \sum_{i,k \in I} (x_{i2k} \cdot C_{ik}) - \sum_{i,k \in I} (x_{i3k} \cdot S_k) \quad (7)$$

C_{ik} is the sum of the cooling made available by a cooling unit at position *i* over the entire domain represented mathematically as

$$C_{ik} = \sum_{l \in I} C_{ilk} \tag{8}$$

This is mathematically represented as

$$\text{minimize} \sum_{i,k \in I} (x_{i1k} P_k) + \sum_{i,k \in I} (x_{i2k} C_k) - 2 \sum_{i,k \in I} (x_{i3k} S_k) \tag{9}$$

The optimization is achieved with the help of integer programming to solve for the minimization cost function (Objective function) coupled with the above set of constraints.

2.4. CFD approach for cooling curve determination

The CFD model configuration and dimensions are shown in the schematic **Figure 7**. In the simulation 5 IT rack space were simulated with the cold and hot aisles. Since this is a rapid simulation, it was assumed no parameter changed along the z direction. To get an overall idea of the cooling profile and to avoid computational complexity a 2D model was preferred over a 3D model. Some of the important features that have been expressed in the simulation are the recirculation of air between the cold and hot aisle through the porous gap as shown in **Figure 7** and the ridges which represent the doors and connection between the IT modules. The Reynolds number for the flow in this regime is between 30,000 and 65,000. Hence the flow is simulated using k-epsilon turbulent model. The servers in the IT racks are modelled as a fan sandwiched between two porous media and this configuration has been adapted from other publications [30]. The recirculation gap and the ridge feature which are not addressed in any research papers significantly affects the air flow as discussed in subsequent sections. Based on the control parameters of the cooling module which are the temperature set point and the air flow rate, the input condition was set as a constant mass flow at a set temperature. The outlet was set as a pressure outlet for the simulation. Inflation condition was applied to the boundaries to enforce the boundary layer effects. The fans were treated as momentum source terms and its value was equal to the equivalent average pressure drop created across the fans. Also the recirculation gaps were treated as porous materials to create some resistance for the recirculation flow.

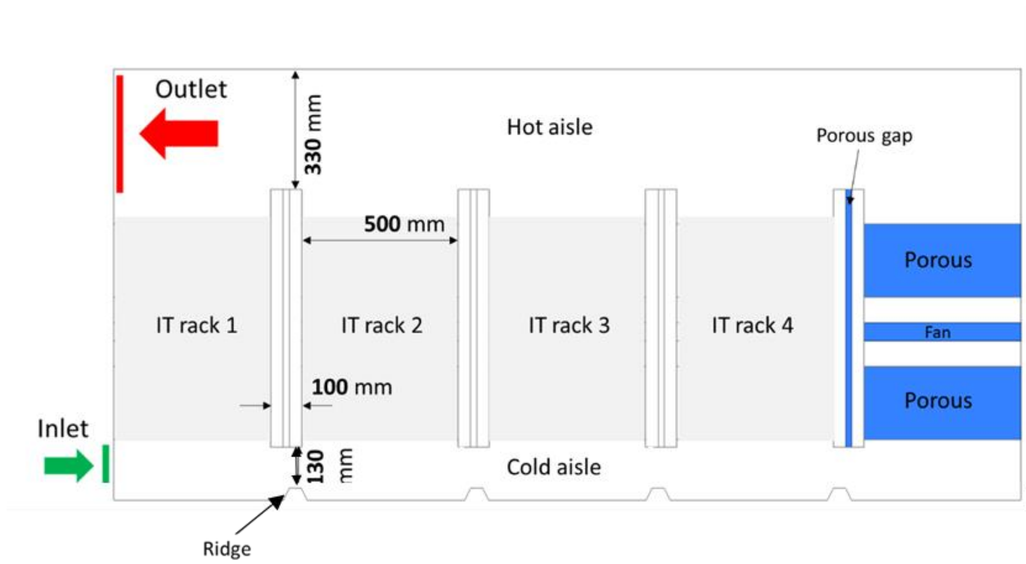


Figure 7. Schematic showing the dimensions of the simulated 2D model. The cooling unit is to the left of the model with the inlet being the cold air from the

cooling module to the cold aisle and the Outlet is the return of the hot air from the hot aisle to the cooling unit. The 4 IT module have the exact features with each of the IT module producing the same amount of heat uniformly distributed with in the porous media closer to the hot aisle.

Module	P	S	S	C	S	S	C	S	S	P	P
Density (kW)	40	15	15	50	15	15	50	15	15	40	40
Power index	1	1	1	1	2	2	2	3	3	2	3
Cooling index		1	1	1	1	2	2	2	2		

Power index = Power source to server and cooling units

Cooling index = Cooling source to the server units

Power and Cooling source

Figure 8. Shows the arrangement sequence of the server, cooling and power units as an output from the optimization algorithm. The third row named power index denotes the connection of the server and cooling units to their respective power module and similarly the fourth row named Cooling index shows the coupling of the servers to the respective cooling units.

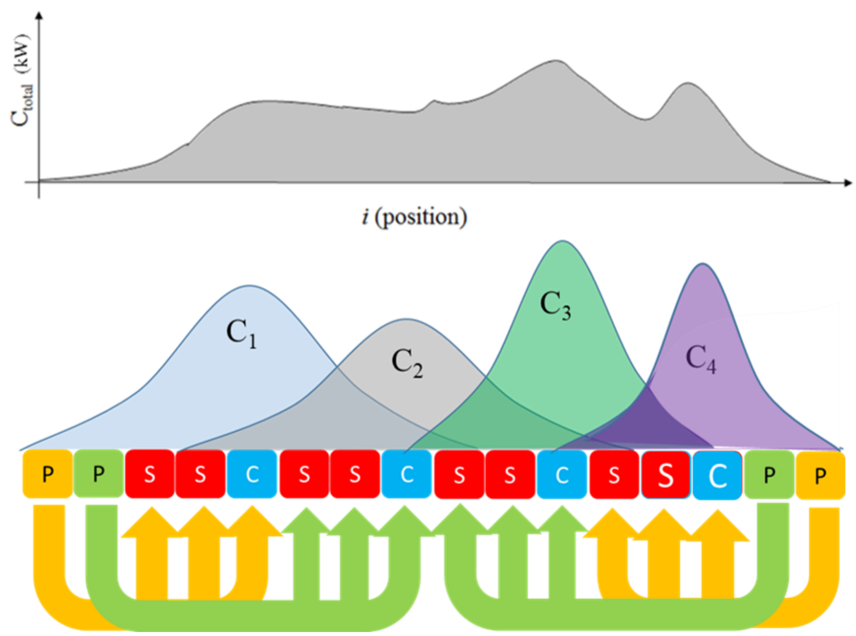


Figure 9. Schematic representation of the power (P) and cooling (C) coupling between IT racks (S), cooling and power units, also generated as an output of the optimization algorithm. The power units' colour coded connection is shown below the sequence and the coupling between the cooling units and servers are shown above the sequence. Furthermore, the top graph of C_{total} vs. *i* (position) shows the net cooling available at any position (*i*).

Because of the many limitations of the first method (bell curve assumption for the cooling curve) that is discussed in the Results and Discussion section, we had to use a different method for optimizing the design process which is described as follows:

2.5. Experimental approach for cooling curve determination

In modular type DC, it is often observed even if the total power of the IT matches the total cooling capacity, the IT equipment runs into problems due to overheating because of the presence of temperature gradient. It is because there are other factors affecting the temperature within the DC, for example the air flow rate from the cooling unit, the temperature of the cold air, recirculation, geometry of the racks, etc. Hence, a systematic series of experiments were performed to understand and identify the major parameters which affect the temperature distribution inside a DC. To begin with, the experiment was performed on an in-house in-row DC with 5 racks and 1 cooling unit.

2.5.1 Experimental setup

The IT racks in total produced from 5 kW to 14.5 kW. The distribution of the IT load was changed based on the experiment. The cooling unit was a Rittal chilled water cooling unit with side way (along the cold aisle) blowing fans as shown in **Figure 10** (a). The total capacity of the cooling unit was 30 kW. A total of 27 temperature sensors were placed in front of the racks and in between the racks at different positions as shown in **Figure 10** (b) and similar arrangement was made behind the racks. The temperature probes in between the racks were to understand the effect of recirculation on the temperature distribution. The speed of the fans and the temperature of the cold air flow from the cooling unit were controlled.

Thereafter, a systematic set of steady state experiments were performed on the system by varying the total IT load, its distribution and airflow rate.

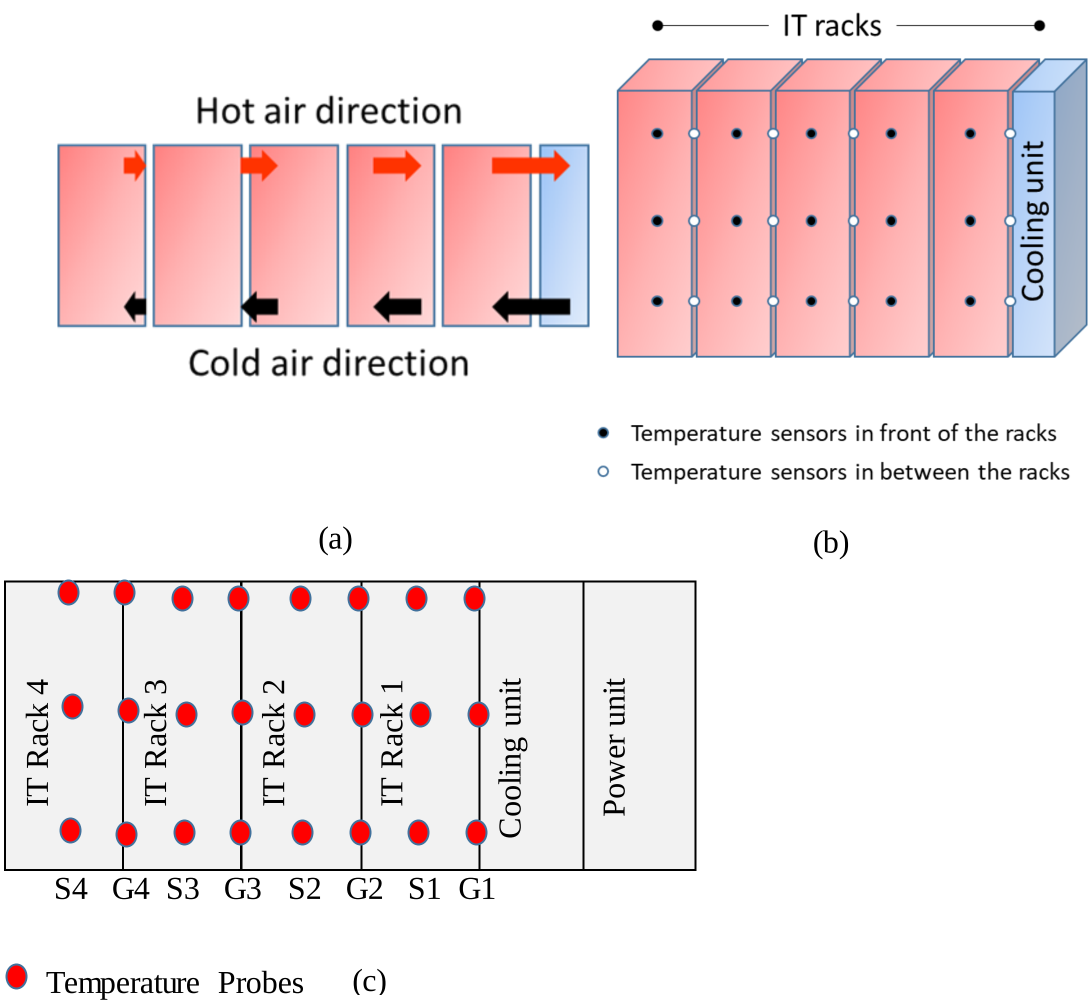


Figure 10. (a) Shows the top view of the in-row DC, where the cold aisle is at the bottom and hot aisle is at the top. The circulation of the air is also marked in the figure. The size of the arrow gives a qualitative value of the magnitude of the air flow rate. (b) Shows an isometric view of the in-row DC with the position of the

temperature sensors marked by white and black dots. Similar arrangement of sensors is also present at the back side of the racks. (c) Shows the setup configuration (front view) and the position of temperature probes that were installed within the test setup along the cold chamber and the hot chamber. In the figure S# represents the IT rack and G# represents the gaps between the IT racks.

Following the experiments multiple CFD simulations were done of the experimental setup to understand the physics of the behaviour. An example of one the several experiment conducted is shown below:

		IT load [kW]					Air flow rate (CFM)
		Rack 1	Rack 2	Rack 3	Rack 4	Total IT	Cooling unit air flow
Experiment 1	a	2.44	3.55	2.7	3.3	11.99	1178
	b	2.44	3.55	2.7	3.3		1766
	c	2.44	3.55	2.7	3.3		2354
Experiment 2	a	2.5	4.26	3.3	4.1	14.16	1178
	b	2.5	4.26	3.3	4.1		1766
	c	2.5	4.26	3.3	4.1		2354

Table 1. One of the several experiments on the in-row in-house DC. The experiments were carried out to understand the effect of IT load and the air flow rate from the cooling unit. Hence, the two sets of experiments with constant IT load but varying air flow rate was carried out and the temperatures at several points were recorded in order to understand the relation. Set Point temperature: 19 °C

The test setup in-house (MC-X) consisted of 4 IT racks with a Rittal CW cooling unit (on the right) with an installed capacity of 30 kW, i.e., 3 fans (maximum capacity is 60 kW with 6 fans). The maximum average air flow rate from the cooling unit at its installed capacity is 2354 cubic feet per minute (CFM). The configuration from right to left is as shown in the **Figure 10** (c).

The first set of experiments were carried out with 4 racks with similar loads for 2 different total IT load as shown in the **Table 1** for different air flow rates from the cooling unit. The temperatures inside the unit were recorded at different points as shown in **Table 2**.

Experiment #1 Load 11.99 kW		Temperature [°C]							
RPM (%) of Cooling unit [2825 CFM at 100% RPM]		G1	S1	G2	S2	G3	S3	G4	S4
100	Cold Aisle	19.1	19.3	19.7	20.6	30.0	26.9	32.3	24.6
		19.2	19.1	19.2	19.1	21.8	20.8	22.3	20.6
		19.0	18.7	18.4	20.3	19.3	19.5	20.3	20.7
	Average	**19.1**	**19.0**	**19.1**	**20.0**	**23.7**	**22.4**	**24.9**	**22.0**
	Hot Aisle		29.8		30.6		34.2		37.8
75	Cold Aisle	17.3	17.1	16.9	21.8	18.8	19.0	25.5	27.5
		17.6	17.4	17.8	17.8	25.2	21.0	26.9	20.8
		17.4	17.8	18.5	20.8	33.9	27.6	34.9	26.6
	Average	**17.4**	**17.4**	**17.7**	**20.1**	**26.0**	**22.6**	**29.1**	**25.0**
	Hot Aisle		28.8		30.0		34.6		39.4
50	Cold Aisle	16.1	16.6	16.1	24.4	32.7	27.6	38.4	38.2
		17.1	16.2	21.8	17.4	33.4	22.3	38.1	29.2
		17.0	17.0	25.9	21.5	36.8	30.9	38.5	37.1
	Average	**16.7**	**16.6**	**21.3**	**21.1**	**34.3**	**26.9**	**38.3**	**34.9**
	Hot Aisle		29.7		31.8		37.9		42.8

(a)

Experiment #2 Load 14.16 kW		Temperature [°C]							
RPM (%) of Cooling unit [2825 CFM at 100% RPM]		G1	S1	G2	S2	G3	S3	G4	S4
100	Cold Aisle	19.3	19.0	18.7	21.4	20.0	20.0	21.4	22.4
		19.6	19.6	19.7	19.6	23.9	21.6	24.4	21.6
		19.6	19.8	20.4	21.7	34.0	29.1	36.1	26.1
	Average	**19.5**	**19.5**	**19.6**	**20.9**	**26.0**	**23.6**	**27.3**	**23.4**
	Hot Aisle		32.8		34.7		39.3		41.1
75	Cold Aisle	18.7	18.4	18.2	24.8	21.1	20.7	31.4	33.1
		18.9	18.8	19.2	19.1	28.9	23.0	32.6	23.1
		18.8	19.2	20.3	23.5	38.0	27.4	38.1	30.6
	Average	**18.8**	**18.8**	**19.2**	**22.5**	**29.3**	**23.7**	**34.1**	**28.9**
	Hot Aisle		32.4		34.6		40.2		41.9
50	Cold Aisle	17.3	18.0	17.4	25.4	37.2	34.7	41.2	43.0
		18.6	17.5	24.2	18.8	38.8	24.7	42.3	35.5
		18.9	18.5	29.8	24.1	40.3	34.6	43.4	41.8
	Average	**18.3**	**18.0**	**23.8**	**22.7**	**38.8**	**31.3**	**42.3**	**40.1**
	Hot Aisle		34.0		37.0		42.6		48.4

(b)

Table 2. Shows the temperature distribution at different IT rack position inside the test setup for (a) 11.99 kW, (b) 14.16 kW. It can be seen that with lower air flow rate (50% RPM) from the cooling unit the temperature of the IT racks 3 and 4 increase significantly. However the temperature of the IT racks 1 decreases with lower air flow rate. Increasing trend in the temperature at the gaps can also be observed.

2.6. Understanding client requirements

The in-row modular DC are fairly new technology and hence there is a lot of scope for improvements. For example the initial design to set up a DC is done manually with the help of engineers for estimating the components needed such as, the number of cooling units needed, selecting commercially available cooling units and so on. Because these estimations are mostly driven by experience, they could be inaccurate as there are no well-studied criteria for accurately estimating these critical factors. This also leads to a huge amount of engineering time being allocated towards the design process which results in excess cost expenditures on part of the company. Our goal is to automate this initial design process, which will not only make more accurate estimations but also do it much faster than the manual process saving much valuable engineering time.

Multiple approaches were taken to find a suitable method to understand the coupling between the IT, cooling and power. The second approach was to conduct a series of experiments and using that to derive an empirical relationship between different IT and cooling units. One of the sets of those series of experiments is what is tabulated above. From these experiments we can derive an experiment driven regression function which will be a relationship between few dimensionless numbers that we have introduced in subsequent sections. Now, once the regression function is formulated, it is then used in an algorithm, to find the maximum number of IT racks (of same IT power) that can be safely put after a selected cooling unit.

Clients who want to set up a small scale DC which they want to grow as the company grows, generally prefer the modular DC, because it is very cost efficient and gives them the opportunity to make it bigger with incremental investments. Because our research is aimed towards these small companies, it’s essential for us to know their requirements in order to setup a design for the DC. As a standard, the requirements to setup a small in-row modular DC are as follows:

- Total IT power
- IT power per rack
- Total number of racks
- Maximum available space
- Type of cooling unit preferred (CW or DX)

CW – Chilled Water cooling units

DX – Direct Expansion cooling unit

2.7. Design algorithm based on experiments

Based on the requirements of the clients, the regression function is then extended to create a selection of cooling units which will satisfy the requirements of the client. Also, we have designed it to produce multiple sets of solutions, calculate their TCO for 5 years, and arrange them in rank of the best solution.

The logical flow of the algorithm is as follows:

The design requirements such as the total IT load, total number of racks and the type of cooling system needed are inputs to the system. Thereafter with the help of a data base of different models of cooling systems from different vendors along with features and with the experimental results, it obtains the effectiveness of each of the cooling system in terms of how many racks it can cool effectively. The different features of the cooling units that are used are as follows:

- Total air flow rate
- The total cooling capacity
- Part load efficiency
- Full load efficiency
- Cost
- Auxiliary units and its costs

Then the operating cost of the cooling unit for 5 years is calculated based on its COP vs. load ratio in case of a DX unit and the fan power consumption for CW unit based on the air flow rate from it. Following that the cost (both initial cost and operating cost for 5 years) of the auxiliary units for each of the DX units are added to the cost of the of the DX units. Then the list of DX units are rearranged in an ascending order based on minimum cost of cooling per rack. For the CW units, chillers are necessary to remove the heat extracted by the cold water in the CW systems. Similarly here as well, the list of CW units are rearranged in an ascending order based on minimum cost of cooling per rack. The chillers are not dependent on the CW units and hence, can be independently selected after the CW units are selected. Depending on how much heat is being rejected form the DC, the total capacity of the chiller systems are calculated. Generally the capacity of the chillers should be more than the amount of heat being rejected from the DC. For each chiller unit available in the inventory its total cost for 5 years is calculated. The cost calculation is sum of its initial cost and operating cost based on its load ratio (full load or part load). A similar method is assigned for arranging the list of chiller units in ascending order based on the minimum total cost per chiller unit normalized over the total IT load that is being handled by that chiller unit.

2.7.1 DX (Direct Expansion) cooling units

The selection criteria for the DX cooling units are as follows:

1. Even if the criteria for measuring the effectiveness of a cooling unit is based on its air flow rate, the first step in selection of the DX unit is to ensure that the total cooling capacity of the unit is more than the sum of IT powers for the racks that it is designated to cool. If the cooling capacity of the DX unit is less than the sum of IT powers for the racks, its effectiveness is decreased by one rack and re-evaluated for satisfying the total cooling capacity criteria. If it passes the evaluation, the effectiveness is accordingly modified.
2. In order to calculate the operating cost, it is necessary to know the part load efficiency (since power consumption is a function of cooling load). The efficiency of a DX cooling unit which is represented as COP (Coefficient of performance), is a ratio of total cooling capacity and total power consumed. The COP is a function of several different factors, but cooling load is the major factor. The data in the inventory has two load values, which

are part load COP (IPLV) and full load COP. Based on a specific standard function for COP vs. cooling load and with the help of an interpolating function the values of the COP are generated at different load as per the requirement.

3. If the total IT power remaining to be installed is more than the capacity of the cooling unit, it picks the first DX unit in the list which is also the minimum cost (since it has been arranged in ascending order based on cost). Following that the corresponding outdoor unit for the selected DX unit is also linked to the selection.
4. If the total IT power remaining to be installed is less than the capacity of the first DX cooling unit in the list, the list is rearranged again based on minimum total cost of each cooling unit. This is useful because the objective is to minimize the total cost of the design. And since the remaining IT power is less than the capacity of the most effective cooling unit, selecting it may or may not give us the least cost. Instead, arranging in ascending order of least cost of cooling unit will ensure that the cost of the last unit is minimum which is capable of efficiently cooling the remaining IT power.
5. From the top each of the item in the cooling unit list is then checked if it satisfies condition 1 (that is the total IT power should be less than the capacity of the DX cooling unit) and for the effectiveness that is the number of IT racks remaining is less than or equal to the number of racks the DX unit can cool effectively.

6. The item in the list which satisfies the criteria first is selected and added to the design list along with its corresponding outdoor unit. This completes the first set of solution for the design of the DC. As a KPI (Key Performance Index) marker the total IT power over the total cooling capacity is calculated. The KPI will always be less than 1, but a KPI close to 1 would mean the total IT power and cooling capacity are better matched.
7. Our objective is to find multiple design solutions for the client. This is done by removing the DX cooling unit that is most used in the first design, and then creating a temporary inventory data list which has all the DX cooling unit information but excluding the one most used.
8. Following that the steps from 1 till 6 is repeated 2 more times to obtain a set of 3 design solutions for the client. These solutions can then be compared and discussed as to which one would be best for the client.

An algorithm flow diagram for the above process is as shown below:

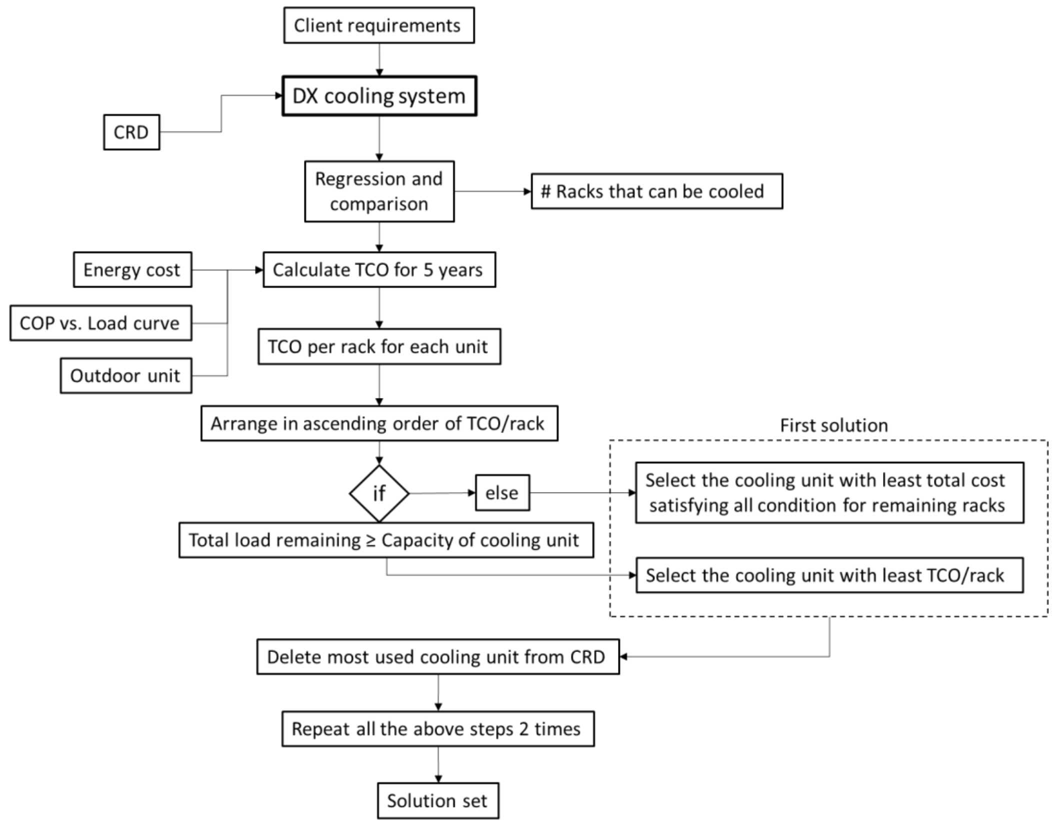


Figure 11. Algorithmic flow schematic of selecting a DX cooling unit for a given requirement.

2.7.2 CW (Chilled Water) cooling units

Unlike DX cooling units CW cooling units do not have compressors in them, rather they use liquid (water) to extract the heat from IT equipment. This water is then pumped to a chiller where the heat is extracted with the help of chillers. Chillers are bigger units of heat removal machines which works on a vapour compression or absorption refrigeration cycle.

1. Similar to DX cooling systems (with the inputs and the inventory already in place), the first step is to find the effectiveness of the cooling unit based on its air flow rate, that is how many rack it can cool with the temperature in the cold aisle within a threshold. This is done with the help of the regression function as mentioned earlier. Then it is ensured that the total cooling capacity of the CW cooling unit is less than the total IT power. If the condition is not satisfied then the number of racks is reduced by 1 and the condition is re-evaluated till it satisfies. The effectiveness is then changed accordingly to take the modifications into account.
2. The power consumption of the CW cooling units is mainly from the fans and the pump. Most of the fans and pumps used in the commercial CW cooling units have a linear relation of load to power consumption. With the minimum and maximum power consumption already known, an interpolation function is used to calculate the power consumed at any given load, which helps us to find out the operating costs for the CW cooling units.
3. Then the CW cooling units in the inventory is rearranged in ascending order based on cost per rack it can handle. Following this an algorithm compares the total IT power remaining and the first option in the cooling unit inventory. If the total IT power is more than the capacity of the CW cooling unit, then the first unit in the list is picked (which is also the least cost).
4. If the total IT power remaining to be installed is less than the capacity of the first DX cooling unit in the list, the list is rearranged again based on

minimum total cost of each cooling unit. As mentioned earlier the objective being minimum cost, and since the demand is limited, the least cost per rack may not be the minimum cost of the design. By choosing a cooling unit which meets the needs with the least cost will be cheaper.

5. Again the rearranged list is checked one by one if it meets all the criteria mentioned in step 1. If it satisfies, the cooling unit is selected, else it moves on to the next one. This completes the selection procedure of the indoor CW cooling unit.
6. To have a complete design, the selection of the chiller systems is necessary. Similar to the inventory of the DX and CW cooling units an inventory of the chiller units available is also made. The selection of the chiller units will be independent from the selection of the CW cooling units because the chiller units have a direct relation with the water that is being cooled and this water serves as a connection between the CW cooling units and the chiller. So there is no direct connection between the chiller and CW cooling units. The criteria to be satisfied by the chiller is that the total IT power should be less than the total capacity of the chiller units.
7. Again the objective here would be to minimize the total cost of the chiller units which satisfies the criteria. To calculate the total cost, which is the sum of the initial cost and the operating cost, a relation between the power consumption and working load is required. Most of the chiller manufactures provide data of full load COP and the part load IPLV for the chiller and

based on that we will calculate the operating cost. Firstly, for each type of chiller unit, the total number of chiller is calculated which would be sufficient for the IT power. Then full load COP will be used for the units which will be working in full load and maximum only one of them will be working in part load for which IPLV will be used to calculate the operating cost. This cost is then divided by the total number of units used to calculate the cost per chiller unit. The same is done for each of the chiller unit in the list and then arranged in ascending order based on the cost per unit.

8. Similar procedure is carried out to select the chiller units as the CW cooling units. The difference is that, the condition to be satisfied here is that the total capacity of the chiller units must be greater than the total IT power.
9. Thereafter, the CW cooling unit and chiller unit which is used most is deleted from the temporary inventory list and the steps from 1 to 8 are repeated 2 more times to find 3 sets of solution arranged which are arranged in increasing order of cost. Later on these design solutions can be analysed by the sales representatives and one that best suits can be selected.
10. As mentioned in the DX cooling unit selection process, a KPI is generated for all the CW cooling units solution as well as for the chiller systems.

An algorithmic flow diagram for the above selection process is as shown below:

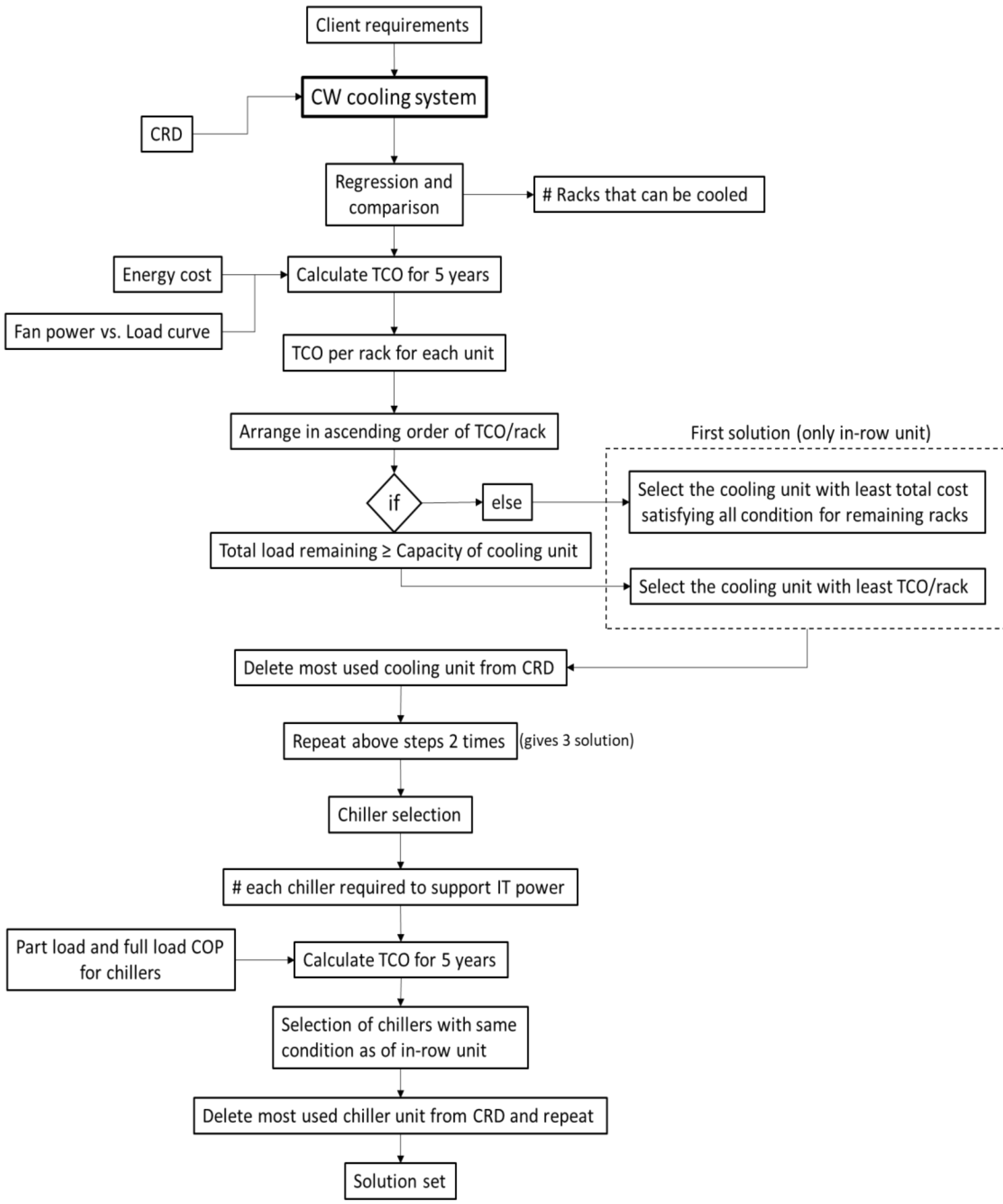


Figure 12. Algorithmic flow schematic of selecting a CW cooling unit for a given requirement.

A Graphics User Interface is made for the program to make it easier for the sales representatives to use the program.

3. Results and Discussion

The coupling relationship between the heat generated in the IT rack and heat rejected by the cooling unit is complex, because fluid behaviour is based on mostly empirical formulas. And, when the conditions change, the behaviour can only be estimated using those empirical formulas. Inside a DC, understanding the air flow pattern and its properties are crucial for understanding and improving the systems. Hence, our experimental techniques will also use empirical relationship to estimate the flow conditions inside the in-row modular DC which we have already defined as 'Cooling Curves' in the methodology section.

3.1. Validation of CFD model

The 'Cooling Curves' are obtained from CFD simulations. To validate the simulation, an experiment was done with the same configuration of the modular DC (**Figure 13**) as in the simulation and the comparison is plotted in **Figure 14**. The plot **Figure 14** (a) compares the air flow across the IT modules starting from the unit closest to the cooling module to the end. It is clear from the simulation that the trend predicted by the simulation matches the experiment. Next, in order to understand the flow distribution a plot comparing the air flow into the IT modules and across the IT modules is shown in **Figure 14** (b). Also a velocity vector map which is coloured by the pressure magnitude is shown in **Figure 15**. From the plot in **Figure 14** (b) we infer that the air flow into the IT modules is non-uniformly distributed. The first and the last IT modules receive maximum air flow and the middle IT modules receive minimum air flow. This flow pattern is shaped by two main features namely the recirculation air and the momentum of the air flow. Firstly, the air flow will slow down as it moves away from the cooling unit due to friction and suction by the servers. Secondly, the recirculation air flow which emerges perpendicular to the air flow at the cold aisle decreases the ability of the servers to suck in cold air flow and increases the chances of receiving the recirculated hot air flow. This recirculated air flow increases in magnitude as we move away from the cooling unit because the pressure difference between the hot aisle and the cold aisle increases (hot aisle is at higher pressure) which creates higher force leading to higher recirculation. But due to minimal recirculation before the first IT module, more air is sucked in by the servers and hence we see a higher mass flow rate of air into the first IT module. At the fifth IT module, the remaining air has only one way to flow that is into the fifth IT module and it receives the maximum recirculated air flow rate. Hence, this results in higher flow rate into the last IT module. It was also observed that the differential pressure between the hot aisle and the cold aisle increases as we move away from the cooling unit, with the 5^{th} rack having the maximum differential pressure and maximum air flow rate of the recirculation air.

From **Figure 15** it can be seen that the ridges are disturbing the boundary layer along the boundary (outer edge) of the cold aisle. This created better air flow distribution into the IT modules. On comparing the flow distribution in simulation with and without the ridges, the simulation with the ridges gives more realistic air flow distribution. So it is inferred that the flow distribution also depends on factors like the number of IT modules, recirculation and cabinet features and other cooling units.

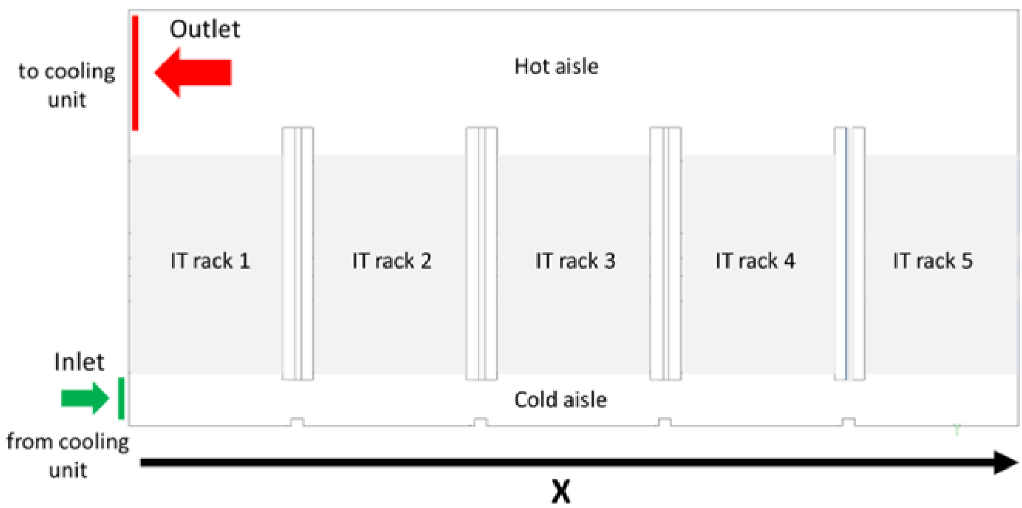
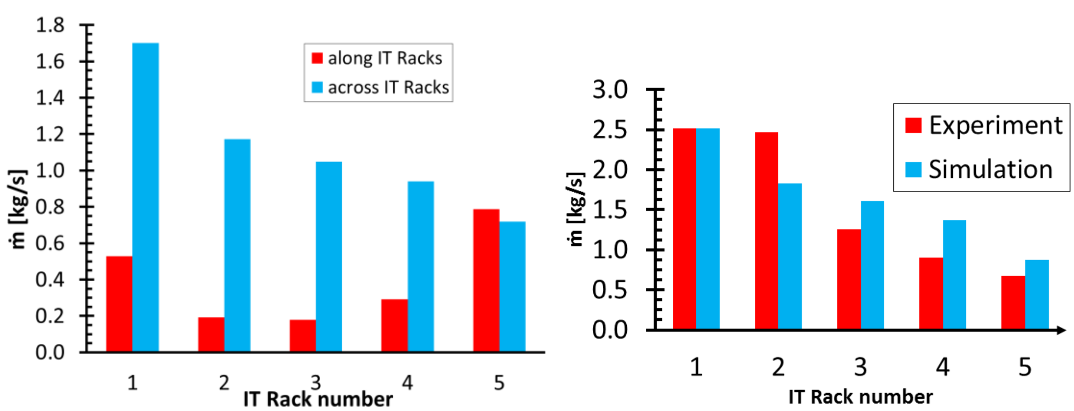


Figure 13. Schematic showing the IT module numbered in the direction of x marked.



(a) (b)

Figure 14. (a) Comparison of simulated air flow rates between across the IT modules and along (into) the IT modules. (b) Comparison of mass flow rate of the air from the simulation into the IT modules (Racks) and along the IT modules for all IT modules starting from the cooling modules (which is along x axis)

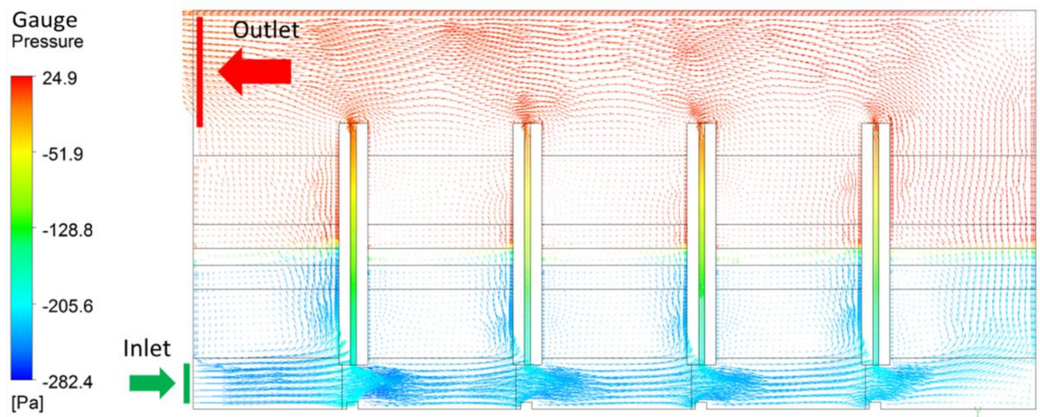
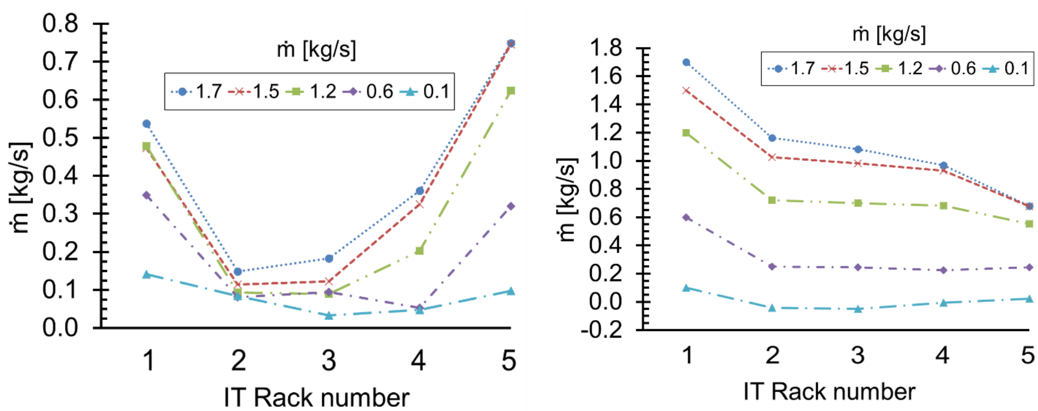


Figure 15. Velocity vector with colours defined by pressure (gauge pressure) magnitude.



(a) (b)

Figure 16. (a) Mass flow rate into the IT module at different inlet mass flow rate (b) mass flow rate across the IT module at different inlet mass flow rate

3.2. Temperature trend vs. IT load and position

An important observation from the experiment can be made from **Figure 17** that is, with decreasing air flow from the cooling unit, the IT racks closer to the cooling unit get colder and the IT racks far from the cooling unit get hotter than at higher air flow from the cooling unit. Alternatively, higher air flow creates more uniform air flow distribution to the IT racks. The effect of hot air flow recirculation from the hot side to the cold side is only amplified from the third IT rack.

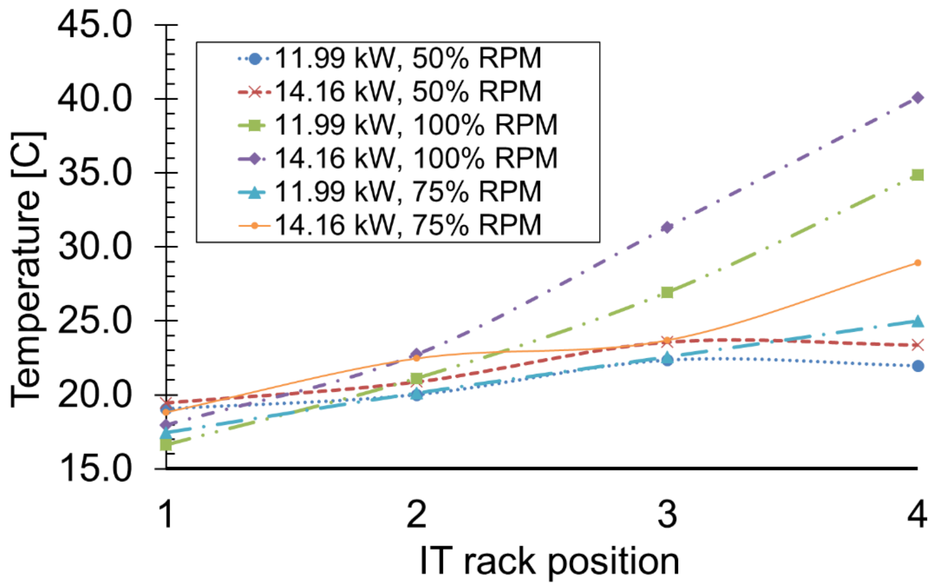


Figure 17. Shows the non-linear increasing trend for temperature with respect to the IT rack position at different air flow rates from the cooling unit decided by the RPM of the fans in the cooling unit for two different IT loads.

From the **Figure 17**, it can be observed that the temperature profile is similar for same air flow from the cooling unit. The temperature change depends non-linearly on factors like air flow from the cooling unit and the distance from the cooling unit. Hence it can be assumed that the temperature in front of any IT rack is a function of the air flow from the cooling unit, the distance from the cooling unit, the total IT load and its distribution.

3.3. Parametric study of the air flow rates

Thereafter a parametric study of the simulation was done over different mass flow rate input from the cooling unit to study the effect of air flow rate on the trend. As shown in

Figure 16 (a) and (b), the flow distribution into the IT module and along the IT module show similar trend over the different mass flow rates at the inlet.

3.4. CFD based optimization example

After we had the 'Cooling Curve' and experiments that proved the air flow rates generated from the 'Cooling Curve' is close to the experimental values, the next step was to implement that into the optimization algorithm. The constraint for the cooling requirement was developed which has been mentioned in the previous methodology section. Following the completion of the algorithm, a GUI for the algorithm was developed to make it easier to access for the sales representatives.

Following is an example of the optimization from the optimization program:

Optimization Example:

Component Data Repository (CDR): This is the built in inventory of the algorithm which stores several different types of commercially available cooling units and power units and their features and parameters.

The following are the types of capacity (kW) of power (P), cooling (C) and IT rack power (S) modules available in the repository of an example case.

P: 25, 30, 35

C: 15, 20, 25

COP: 5, 6, 7

S: 5, 7, 10

Requirements by client:

The client requirement for the DCs are as follows:

Floor space available: 9×2 m^2

Required IT: 60 kW

Optimization Output:

The following sequence as shown in **Table 3**, is the output sequence from the algorithm which arranges the modules based on the minimum over head of installed power and cooling capacity.

Module Type	P	P	S	C	S	S	C	S	S	C	S	S	C	S	P
Capacity [kW]	25	25	5	15	7	7	15	7	10	25	10	10	20	5	25

Table 3. Optimized placement of the Power, cooling and IT modules

Area needed: 5×2 m^2 which is within the available floor space by the client.

3.5. Pressure criteria for cooling range

We have already discussed that keeping the temperature of the air flow in the cold aisle within ASHRAE guidelines is absolutely crucial for the modular DCs. This is to keep the servers healthy, prevent failures and increase their life. The most important factor which leads to an increase in temperature in front of the servers is due to leakage. It is difficult to completely seal all the gaps between the racks and prevent recirculation because of the frequent maintenance activities that goes on with in the data centres.

Due to recirculation when the hot air flow is mixed with the cold supply air flow before it is inhaled by the servers, it increases the temperature in front of the servers and leads to inefficient cooling. The goal is to determine how the recirculation depends on the cooling unit, IT load and the distance from the cooling unit. In order keep the supply air temperature constant we can create perfect confinement, which is very difficult to do as mentioned earlier or the other way is to reverse the direction of the recirculation air. This can be simply achieved by increasing the pressure of the air from the supply which is the cooling unit. This will force the cold air to bleed into the hot aisle, reversing the direction of recirculation and hence keeping the cold aisle at a constant temperature.

3.5.1. Pressure criteria based cooling range

An exhaustive CFD simulation study was done to understand the effect of local pressure around the gap between the racks on recirculation. A simulation was preferred than an experiment because it is difficult to measure the local pressure differences between the hot aisle and cold aisle near the gaps. The basic skeleton of the geometry consists of similar arrangements of IT racks and cooling units as done earlier. The IT racks are mimicked by porous medium sandwiching the fans inside the servers as shown in **Figure 18**. Similar method was followed for the cooling unit where the heat exchangers are mimicked by the porous medium sandwiching the fans. **Figure 19** shows the pressure contour and the velocity vector of one of the cases. The direction of the recirculated air can be seen from the inserts **Figure 19** (a) and (b) where it can be seen that the second gap from the cooling unit has recirculation air from the cold aisle to the hot aisle whereas the fourth gap has recirculation air from the hot aisle to the cold aisle. This shows that the local pressure difference (cold aisle pressure – hot aisle pressure) around the gaps has a trend which decreases and becomes negative.

Several configurations of IT racks and cooling units were simulated along with the number of IT racks with each cooling unit as can be seen from the plots in **Figure 20**(a), (b) and (c). The configurations that were simulated are 2, 3, 4, 5, 7 and 9 IT racks with one cooling unit. It was seen that there was recirculation happening in both directions. The IT rack gaps closer to the cooling unit had recirculation form cold aisle to the hot aisle and the ones far from the cooling unit had recirculation from hot aisle to cold aisle. And interestingly, irrespective of the cooling unit's capacity, in none of the simulation the recirculation in the gaps from the cold aisle to hot aisle (favourable recirculation) was seen after the third rack.

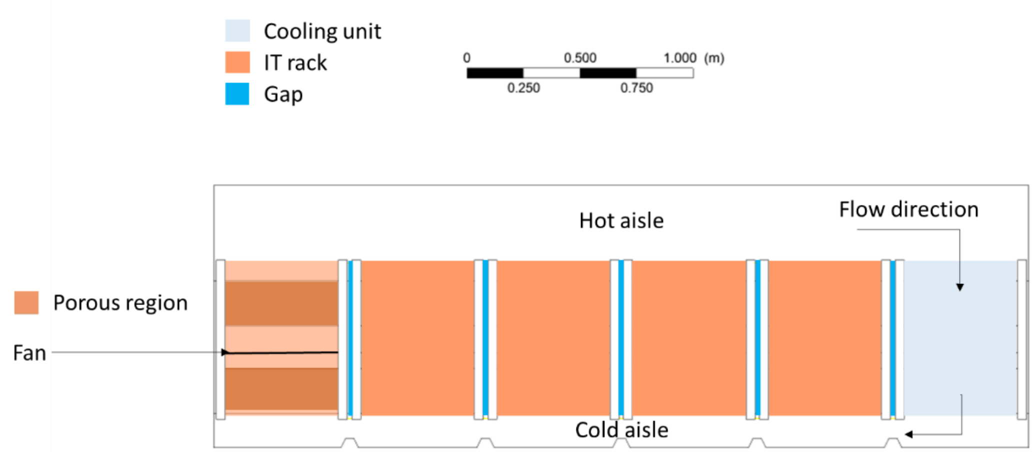


Figure 18. Geometry for the 2D simulation of IT racks and cooling unit. The IT racks are separated from each other by porous gaps represented in deep blue colour and the cooling unit is to the extreme right. The overall air flow direction is shown by the arrows. The bottom aisle is the cold aisle from where the cold supply air is sucked in by the servers and the top aisle is the hot aisle from where the hot air is sucked by the cooling unit.

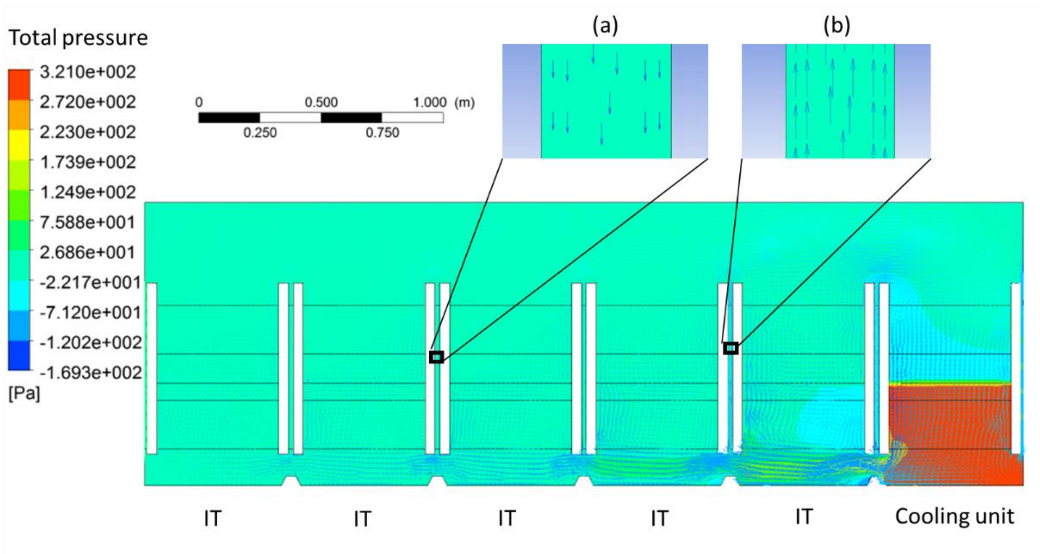


Figure 19. Shows the pressure contour along with the velocity vectors of the air flow in the model. It is seen that a high-pressure zone is created in front of the cooling unit fans and pressure slowly drops along the cold aisle whereas the pressure slowly increases along the hot aisle from right to left.

This study shows us a novel method to keep the temperature in the cold aisle from increasing. By maintaining higher pressure in front of the servers, i.e., the cold aisle, the recirculation can be made to happen in the opposite direction that is from the cold aisle to the hot aisle. It is also inferred from the simulation study that, maximum 3 racks following the cooling unit will have cold air recirculated from front to back through the gaps because of higher pressure at front than the back of the servers irrespective of the power of the fans(within available commercial limits) in the cooling unit. From the next gap, the direction is going to be reversed and the hot air will start leaking into the cold side (front of the servers) increasing the supply temperature of the air into the servers (because of higher pressure at back of servers).

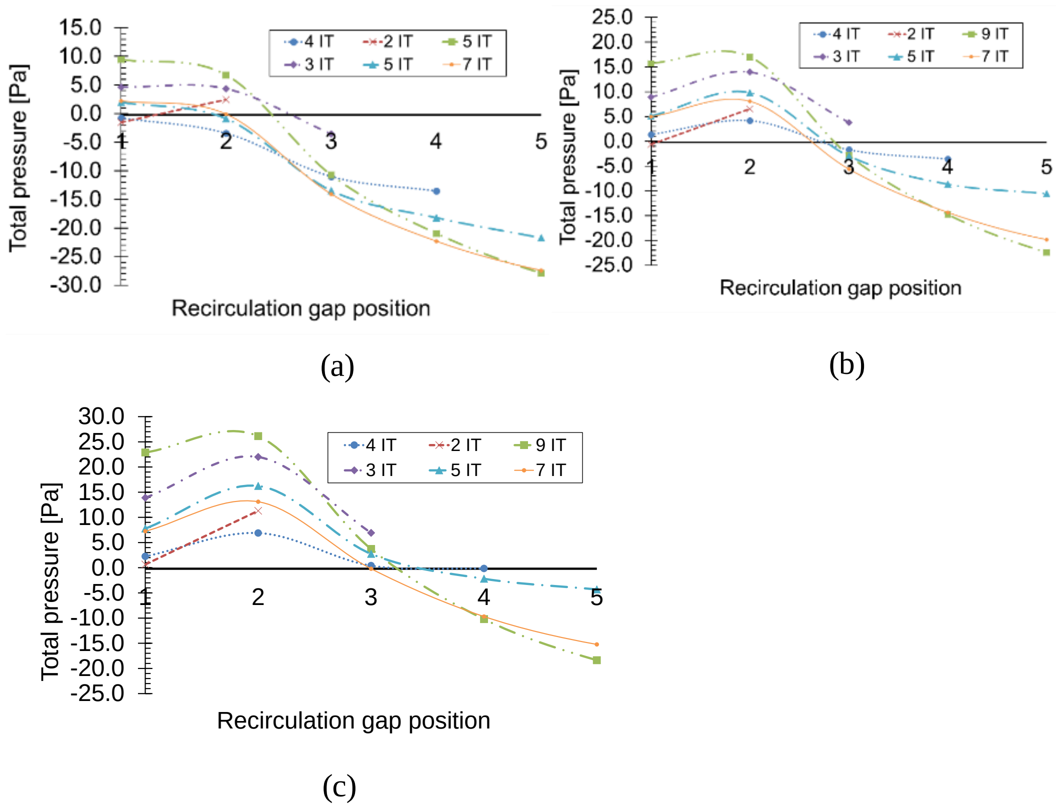
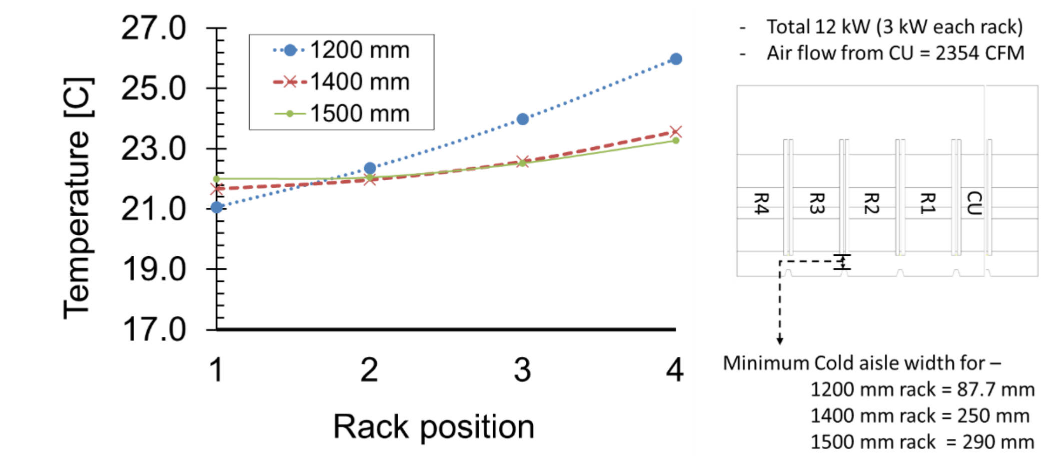


Figure 20. Shows the local differential pressure (cold aisle pressure – hot aisle pressure) at the respective position of the gaps from the cooling unit for a maximum velocity defined for the fans in the cooling curve as (a) 3 m/s, (b) 5 m/s and (c) 10 m/s. From the plots, it can be observed that the limiting position of the gap for the differential pressure to be positive is 3. After the third gap, the differential pressure is negative which means that the hot air will leak into the cold aisle which is undesirable as mentioned earlier.

With the differential pressure method even though we can prevent hot air recirculation, it comes at a higher cost. Current commercially available in-row cooling units don't have the ability to produce an air flow rate as shown by simulations. But that air flow rate can be achieved with the help of auxiliary fans. This adds to the initial cost of the cooling unit as well as to the operational and maintenance cost. This would result in a higher design cost for the initial design which is not optimal and doesn't help reduce design cost. Hence we moved on to explore another method to design the modular DC.

3.6. Cold aisle width effect on cooling range

Another important parameter which affects the temperature gradient in the cold aisle is the width of the cold aisle. Multiple CFD simulations were done with same configurations as the one mentioned above but with different cold aisle width. Rack size of 1200 mm, 1400 mm and 1500 mm with cold aisle width of 88 mm, 250 mm and 290 mm respectively were studied and the temperature along the racks were plotted as shown in **Figure 21** (a). The configuration consisted of 4 IT racks namely R1, R2, R3 and R4 in **Figure 21** (b) of 3 kW each with 1 cooling unit with a set air flow rate of 2354 cubic feet per minute (CFM). From the plot it can be seen that with decreasing cold aisle width the temperature gradient increases. Also, there seems to be a saturation point where increasing the width of the cold aisle doesn't significantly affect the temperature gradient.



(a) (b)

Figure 21. (a) Shows the temperature in the cold aisle in front of every rack for 3 different cold aisle width 1200 mm, 1400 mm and 1500 mm. (b) Shows a schematic of the geometry with the position of the cooling unit (CU) and IT racks (R1, R2, R3 and R4).

3.6.1. Limitations of bell curve assumption of cooling curve

Here, let's take a step back to look at the limitations of assuming the cooling units capacity to be distributed as a bell curve on both sides as described in the first method. Often times, cooling units are places at one end of the sequence, allowing air to pass to only one side. It is difficult to account for such changes in the cooling unit's configuration and hence its 'Cooling Curve' will be incorrect. This scenario is going to be very common since this design tool is being made for small size modular DCs for small businesses. Another reason is that, because of the way some cooling units are designed, it restricts to some extent any air to pass through it other than its own air flow. This makes the overlapping assumption of the cooling curve to be incorrect in some situations resulting in non-optimized arrangement.

Furthermore, from the experiments on the in-house modular DC, it was found that the cooling capacity distribution in form of a bell curve is not very accurate. Rather, it depends on several other factors, like IT rack power, recirculation, geometry, the type of servers, type of fans in the cooling unit, etc. Because of all these variables, significant deviations of the cooling range from the bell curve assumption was found.

3.7. Parameters affecting cooling range

After analyzing the advantages and shortcomings of the CFD based optimization algorithm method, we decided to base our study on experiments rather than simulations, since there are several factors which affect the air flow significantly and all of them cannot be implemented in the simulation. The results from the second method that is using actual experiments to get the relation between position of the IT rack and temperature in front of it is explained in the following section.

It was observed that there are several different parameters which affect the temperature in cold chamber of the modular DC. One of the major parameter affecting temperature distribution is recirculation. Recirculation is defined as the reverse flow of fluid in a region due to pressure differences reducing the efficiency of the operation. This pressure difference is an effect of the fans in the servers and the fans in the cooling unit. In general, the pressure created by the fans of the servers in the IT racks is more than the pressure difference created by the fans of any commercially available modular cooling unit. This means that the hot chamber of the modular DC will be at a higher pressure than the cold chamber, causing the hot air to seep through the gaps and mix with the cold air in the cold chamber. This mixing increases the temperature of the cold air flow making it harder to reject heat from the servers. Also, this effect is magnified if there are more empty spaces between the racks which gives an easy path to the air flow, facilitating more mixing and higher temperature gradient in the cold aisle. Calculating a value for the recirculated air is a difficult problem because of the uncertainties in the spaces between the racks and complexity within the racks due to cables, holes for fittings and server types. A good analysis on recirculation air and its effect on the heat rejection is done by Evans et al. [31].

Another parameter which influences the temperature gradient in front of the racks is the distance from the cooling unit. As the air travels further from the cooling unit more and more heat is added to the cold air from outside the modular box due to leakage, conduction etc. Also, flow velocity decreases due to the complex geometry of the cold chamber making it harder for the cold air from the cooling unit to reach a rack far from the cooling unit.

The amount of air sucked in by the servers in the rack also largely influences the temperature gradient in the cold chamber. It is the mismatch between the airflow rate supplied by the cooling unit and the rate of air sucked in by the server fan is what gives rise to majority of the recirculation within the modular DC. Hence, during the design of a DC higher cooling capacity units are installed and the temperature set point is lowered to keep the temperature under the required limit.

Other factors like set point temperature in the cooling unit, total number of racks, and width of the cold aisle also affect the temperature distribution but deriving a formula to define all these effects is extremely difficult because of the complex and uncertain geometries of the DC and also because these parameters are highly coupled. Instead, we can capture all these effects without worrying about the actual relation using an empirical formula. With the help of a dimensionless number, the parameters, their coupling and effects can be captured on a global level. Having identified the parameters we describe the dimensionless number as follows.

$\frac{Ts - Tc}{T_{ref} - Tc} =$ Dimensionless ratio capturing the temperature in the cold chamber. It is also referred as Dimensionless T (temperature).

where, $Ts =$ temperature in the cold chamber in front of any rack

$Tc =$ set point temperature of the cooling unit

$T_{ref} =$ the reference temperature

$\frac{m_s}{m_c} =$ Dimensionless number capturing the effect of the airflow rate supplied by the cooling unit and the air flow rate captured by the cooling unit. This also captures the effect of recirculation since it is a result of the mismatch in airflow rates between the servers and the cooling unit.

where,

$m_s =$ air flow rate sucked in by the servers per rack

$m_c =$ air flow rate produced by the cooling unit

$\frac{L}{w} =$ Dimensionless number capturing the effect of distance from the cooling unit and the width of the cold chamber

where,

$L =$ the distance of the rack from the cooling unit

$w =$ width of the cold chamber in front of the racks

After identifying the parameters, experiments were performed on the in-house modular DC to obtain the empirical relationship between the temperature and the defined parameters. Experiments were performed on a 4 IT rack system with one cooling unit, following that the extracted experimental results were fitted to a second order curve (between dimensionless T and the parameters) using a regression function. Later, during an attempt to verify the cooling curve algorithm for different size of modular DC, the experiments were repeated on a 2 IT rack system. Upon analysis, the regression curve used to fit the data for the 4 IT rack system did not fit the 2 IT rack system, rather it resulted in a curve which followed similar trend of the curve but along a different curve as shown in **Figure 22**. It was clear from the plot that another parameter which clearly affected the temperature distribution in front of the IT racks was the total number of racks in the system. Hence, appropriate modifications were made to the current fitting to account for another parameter creating a multiple independent variables and one dependent variable that is the dimensionless temperature as shown below. Multivariable regression function was used to generate the relation between these variables.

$N =$ total number of racks in the system

And these dimensionless numbers are related by a polynomial function of some order represented as:

$$\frac{Ts - Tc}{T_{ref} - Tc} = f\left(\frac{m_s}{m_c}, \frac{L}{w}, N\right) \tag{10}$$

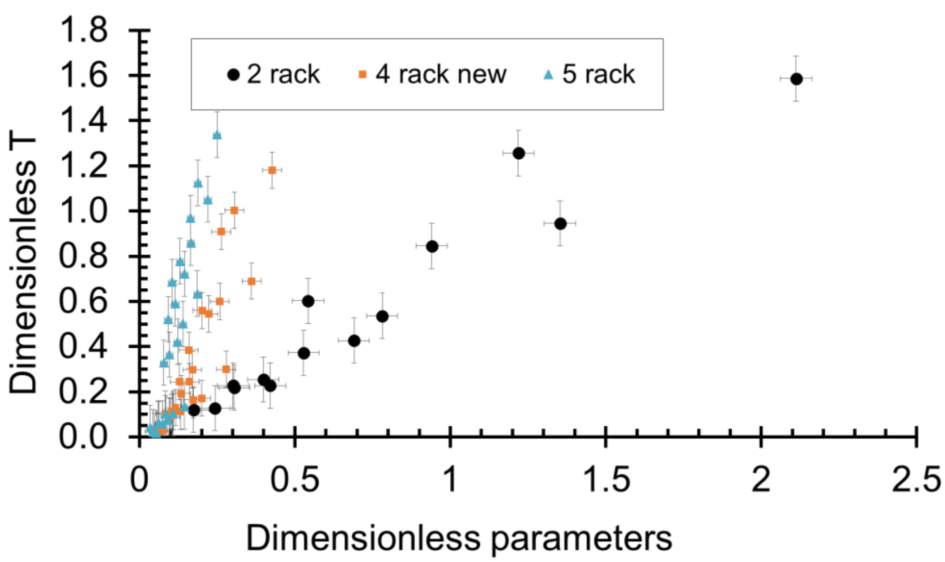


Figure 22. Plot showing the regression fitting between the dimensionless temperature and the dimensionless parameters for different system size of the modular DC. It can be seen that the curves for different system sizes namely 2 IT rack, 4 IT rack and 5 IT racks the relationship curve increases in slow in the order of the system size, with 2 IT rack system having the minimum slope and 5 IT rack system having the maximum.

3.8. Multivariable regression of dimensionless numbers

Since the problem at hand is a multivariable polynomial regression, MatLab was used to generate regression function of several orders. The RMSE of the experimental value from the calculated value was found for all the cases. Upon comparing them and it was found that there was not much difference between the second order and higher order RMSE values. Also to avoid over fitting the data by the regression function, a low order polynomial regression was preferred. Hence, after thorough analysis second order polynomial regression function was selected for defining the relation between the dimensionless numbers and the relation is as shown in the equation below.

$$\begin{aligned}\frac{Ts-Tc}{T_{ref}-Tc} &= -0.27365\cdot N -0.075009\cdot\left(\frac{L}{w}\right) +0.0075848\cdot\left(\frac{L}{w}\right)\cdot N \\ &+1.1805\cdot\left(\frac{m_s}{m_c}\right) -0.25454\cdot\left(\frac{m_s}{m_c}\right)\cdot N +0.061221\cdot\left(\frac{m_s}{m_c}\right)\cdot\left(\frac{L}{w}\right) +0.0084666 \\ &+0.0091754\cdot\left(\frac{m_s}{m_c}\right)^2 +0.00146\cdot\left(\frac{L}{w}\right)^2 +0.056652\cdot N^2\end{aligned} \quad (11)$$

With reasonable accuracy and higher certainty that the curve is not over fitted the equation was implemented into the algorithm to find out the number of racks that can be placed after a cooling unit with the temperature remaining within the threshold. A figure showing the accuracy of fitting is shown below in **Figure 23**. The x axis represents the dimensionless temperature values calculated by the regression and the y axis represents the dimensionless temperature values obtained from the experiments. The closer the data point is to the x=y line, the better is the accuracy. It can be seen that most of the points fall around the x=y region.

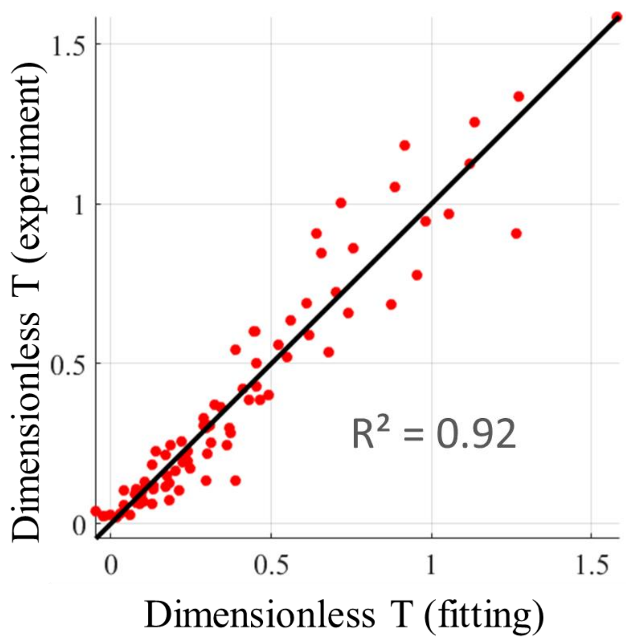


Figure 23. Shows the accuracy of fitting a second order polynomial function to the dimensionless numbers. The R-squared value for the fit is 0.92.

3.9. Redundant Systems

Server downtime leads to suspension of services, loss of valuable information, customer dissatisfaction and hence costs a lot of money for the clients. To make the servers run 24×7, the system has to be fault tolerant. The +1 in N+1 represents the number of fails the system can tolerate that is, it can tolerate one failed system. Since there are extra module running all the time (or ready to be online), failure of any one module wouldn't affect the servers. This gives sufficient time to find the fault and take appropriate measures. Designing redundant systems for cooling units is much more complex than designing it for power units and is discussed in the next section.

3.9.1. Power redundancy

Now, for fault tolerant systems in power, because of the limitation that each server (and cooling modules) can only take finite number of input power connections (that is generally 2), figuring out the connections for N+1 configuration with minimum installed power module manually is difficult. Hence, here an algorithm is developed which would automatically figure out the connections from the power systems to the respective loads much faster and with minimum number of installed power modules.

3.9.1.1. Mathematical Model

Terminologies:

Loading Fraction: This is defined as the ratio of the total load to the available power supply module (redundancy units included). Mathematically defined as:

$$lf = \frac{total\ load}{Maximum\ available\ power}$$

Conditions

1. All power modules are of same capacity

Sequence generation using Optimization:

Decision variable:

$x_{ij} =$ Represents the decision if ith power will power jth load ($\in \{0,1\}$)

Other parameters:

$P_i =$ Capacity of ith power unit (e.g. $\in \{30,30,30\}$) (kW)

$L_j =$ jth load unit (e.g. $\in \{5, 10, 7, 15\}$) (kW)

Objective function:

Minimize $\sum_{i=1}^{I}(P_i \cdot lf - \sum_{j=1}^{J} x_{ij} \cdot L_j)$

The objective is to minimize the difference between nominal distribution of load and actual distribution of load.

3.9.1.2. Constraints

Linear Constraints:

1. $$\sum_{i=1}^{I} x_{ij} = 2, \forall j \in \{1,2,..J\} \tag{12}$$

   This represents that each load has to have two power connections, and from separate power units. The expression represents J equations.

2. $$\sum_{j=1}^{J} x_{ij} \cdot L_j \le P_i, \forall i \in \{1,2,..I\} \tag{13}$$

   This represents that the sum of all the loads connected to a power unit should be less than that power unit. The expression represents I equations.

Non-linear Constraints:

$$\sum_{j=1}^{J} x_{ij} \cdot x_{nj} \cdot L_j + x_{nj} \cdot L_j \le P_n, n \in \{I - i\}, \forall i \in \{1,2,...\} \tag{14}$$

This represents if one of the power units fail, the compensating load on the other power units should be less than or equal to the maximum capacity of that power unit. The expression represents $n*I$ equations.

It was realized that there was no actual need for optimization in finding out the redundant connections for the power systems, and it was only used as a dummy to run the optimization algorithm on MatLab. Following that, by studying the pattern of redundant power connections a better algorithm was developed that doesn't require any optimization and finds out the redundant connections faster. The code for the algorithm has been described in the appendix section.

Example:

Given cooling and IT loads

Index	A	B	C	D	E	F	G	H
Load [kW]	5	5	5	7	7	7	10	10

Capacity of each power module: 30 kW

Optimization result

Power modules used: 3

Connection output:

Figure 24 shows the output generated from the algorithm which finds a possible power connections for redundant power supply where,

P_1 *connects* A, B, D, G and H

P_2 *connects* A, B, C, E, F and G

P_3 *connects* C, D, E, F and H

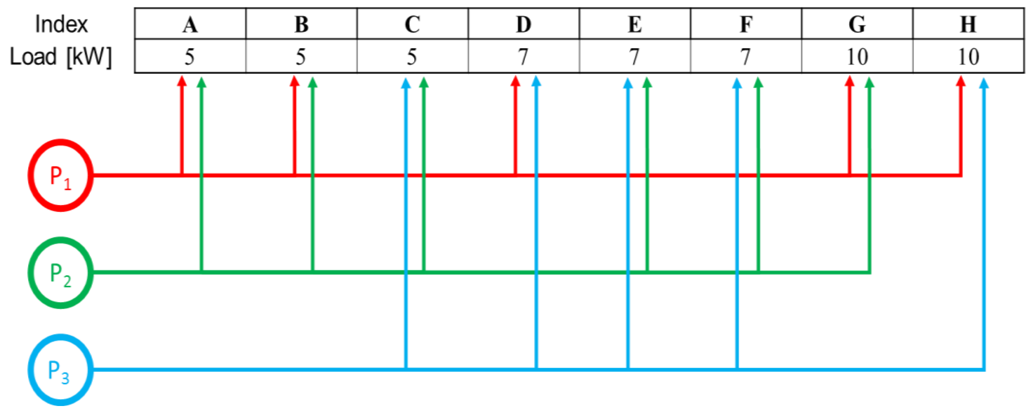


Figure 24.The redundant power connection generated from the optimization program. P_1, P_2, P_3 are the 3 different power modules. Each load unit is connected to two power sources indicated by the respective colours of the power modules.

3.9.2 Cooling redundancy

Unlike power redundancy, cooling unit redundancy is much more complicated. In electrical system the interaction of multiple power units are well defined and understood. But, there is no research which discusses the interaction of the air flow from multiple modular cooling units. Without that understanding, optimizing redundant cooling system is difficult.

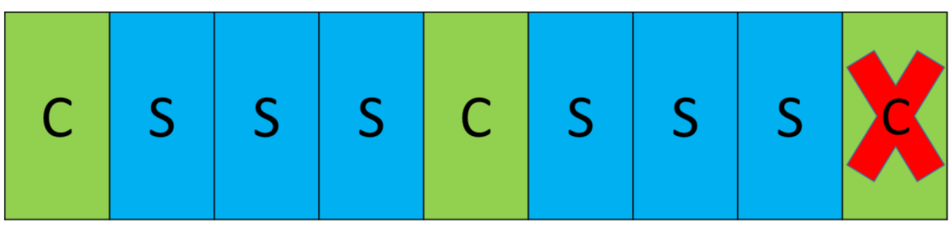


Figure 25. Arrangement of 2 cooling and IT servers where the air from the middle cooling unit can reach both directions (left and right). The situation simulates a

redundant cooling system where one of the cooling unit (to the extreme right) has failed. So, with just 2 cooling units, understanding the distribution of the air flow from the middle cooling unit to the left and right is essential in designing redundant system.

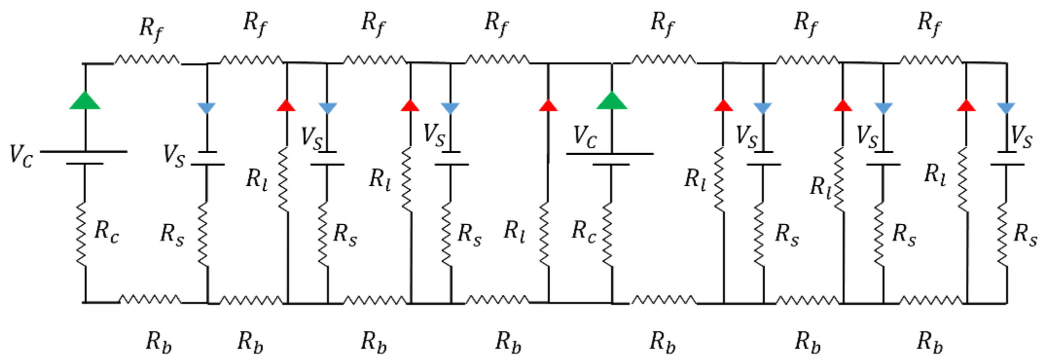


Figure 26. Representation of the above configuration in an equivalent circuit form. Here, R represents resistance, with subscripts represent the following: s= IT rack, l= leakage, f=front, b=back and c=cooling

In an effort to understand the interaction qualitatively, the combination of cooling system and the servers can be considered as a system of resistors and voltage source and the air flow is the current. Let's consider a situation where there are 3 cooling units and 6 IT racks (Redundant system with N+1 redundancy) and one of the cooling units is not operational which are arranged as shown in the **Figure 25**. We are considering a situation where one of the cooling unit has failed. Our aim is to evaluate if the central cooling unit is able to effectively supply sufficient air flow to the IT racks to its right. The equivalent circuit diagram for the case in shown in **Figure 26**.

Each cooling unit can effectively cool up to 3 IT racks if all the air is being directed to one side. For the cooling unit on the extreme left, all its air is being directed to the right. Hence, the 3 IT racks next to it are cooled effectively. The cooling unit in the middle, has opening on both sides, allowing air to freely flow in both direction (left and right). The air flow to the left will experience some opposing force because of the pressure created by the left most cooling unit. Hence, the air flow will not be equally divided, rather more air will be reaching the right than left. Let's say that the total air flow rate is 1. Then if α amount of air is reaching left, (1-α) is reaching right. If this α can be determined, optimized redundant cooling systems can be designed.

One of the major challenge is that α depends on several factors. Upon solving the circuit with KVL and KCL equations,

$$\alpha = f(R_c, R_f, R_s, R_b, V_c, V_s) \quad (15)$$

To study the effect of these parameters on α, a MatLab Simulink model was created and these parameters were systematically changed and the values of current or air flow rate were recorded.

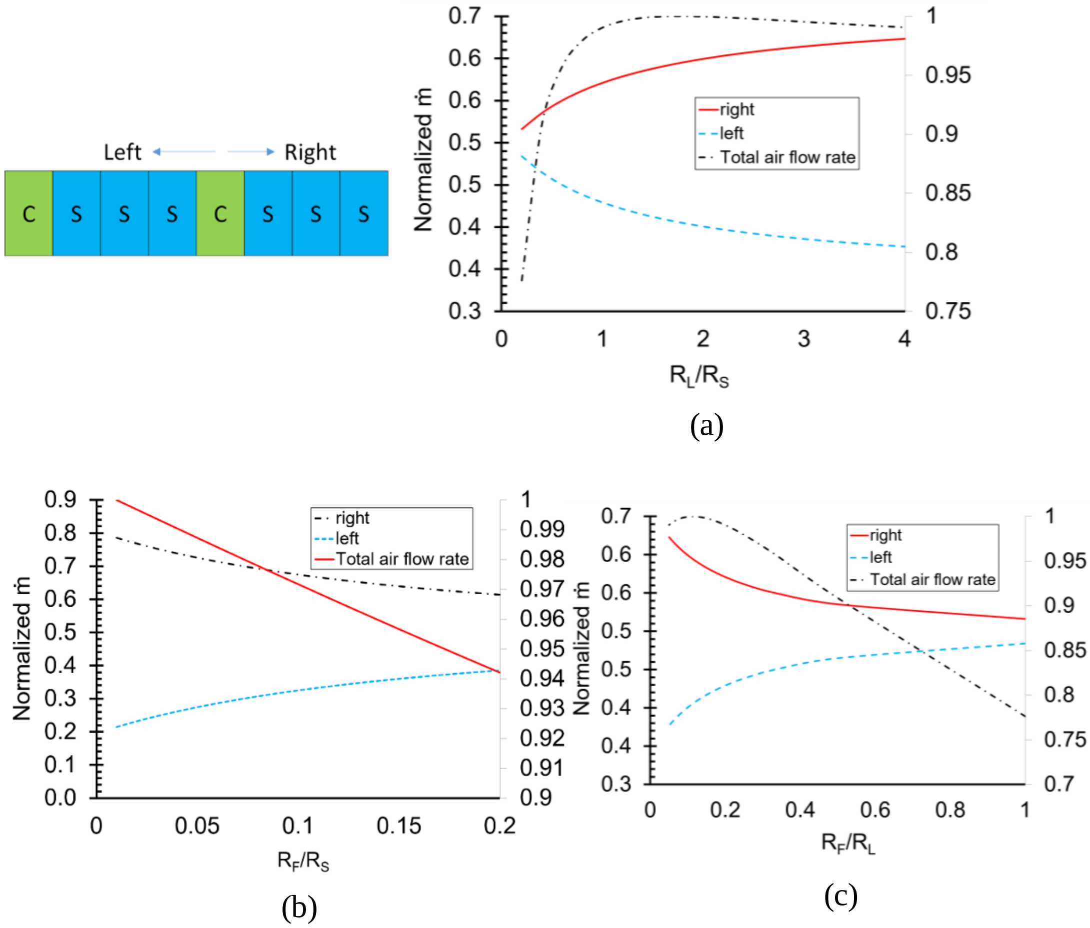


Figure 27. Variation of normalized air flow rate (normalized to total airflow rate from the cooling unit) with the ratio between different resistances. Shows the variation of air flow rate to the left and right side of the cooling unit and the total air flow it with the (a) ratio of resistance of the cold aisle over resistance of the IT racks, (b) ratio of resistance of leakage over resistance of IT racks, (c) ratio of resistance of cold aisle over resistance of leakage.

It can be inferred from the plots in **Figure 27** that the air flow rate is strongly dependent on the resistance of the cold aisle and resistance of the leakage. Also, the factor α varies non-linearly with these parameters. In **Figure 27**(a) as the resistance of the cold aisle is increased the total air flow rate decreases and the distribution of air into the left and the right side seem converging. This is affected by the more dominant pressure difference. When the resistance of the cold aisle is low, the air flow is mostly driven by the pressure created by the leftmost cooling unit, making most of the air flow to go towards low pressure that is right-side. But as the resistance of the cold aisle increases, the pressure created by the high resistance dominates, which tries to equally distribute the air flow towards bot left and right. In **Figure 27**(b), when the leakage resistance is very small, causes heavy recirculation between racks. This nullifies any pressurization created by the leftmost cooling unit and hence air is equally distributed to the left and right. Also because of this recirculation, the air flow from the cooling unit decreases, since most of the air flow is directed towards recirculation. As the resistance of the leakage increases, improvement in both total air flow rate and air distribution is seen because recirculation is minimized and the flow is mostly governed by the pressurization of the cooling unit (leftmost). In **Figure 27**(c) similar trend can be seen as the leakage resistance is decreased. At high leakage resistance that is at low R_f / R_l ratio, high air flow rate from the cooling unit is observed and most of the air is directed towards the right side. And with decreasing leakage resistance, effect of recirculation magnifies, which reduces the air flow rate in the cooling unit and nullifies the pressurization in the left side of the cooling unit leading to more uniform distribution of air flow from the observed cooling unit. For larger systems with more than 3 cooling units α will be coupled with many more parameters. There may be other factors like pressure difference created by more cooling units and IT racks which have not been studied here. Also, these simulation results require experimental validation, which cannot be performed with the existing infrastructure in the lab. Hence, proper understanding of the flow is not known which is elemental in designing efficient redundant cooling systems. Therefore, redundancy for cooling unit is not included in the design algorithm here.

3.10. Case Studies

Here we discuss one example case where for a given client information a selection of cooling unit is generated along with the arrangement configuration.

Client requirements: Total IT = 40 kW, Total number of racks = 7 (equally distributed load) with a requirement of CW type cooling units.

Sample Component Data Repository (CDR) contains a list of 6 CW cooling units and 6 DX cooling units and 4 chiller units with all necessary information and auxiliary parts.

The KPI (Total IT load over total cooling capacity) is also calculated for all the design outputs. The TCO for the individual components and total TCO is also calculated and tabulated in the design output.

After running the algorithm with the required client inputs the following are the outputs:

Arrangement sequence:

C	S	S	S	S	C	S	S	S

Where,

S	= IT racks of 5.7 kW each

C	= Cooling unit

The table below shows 3 different design options available for the cooling unit, with option 1 being most favourable according to the algorithm.

Design Output			
	Option 1	**Option 2**	**Option 3**
Cooling unit model	Stulz CW CRS-090-C	Qcooling CL80-In Row 48U x 300	DataAire GICW-03012
Capacity [kW]	26	55	24.5
Chiller unit model	Johnson Controls YCAL0019EE	Carrier 30RAP0155K-58F10	Trane CGAM-C20
Capacity [kW]	52	49	70
KPI	0.77	0.36	0.82
TCO cooling unit [CAD]	22,902.60	24,799.00	28,987.10
TCO chiller unit [CAD]	89,849.60	91,664.02	105,887.62
TCO [CAD]	112,752.20	116,463.02	134,874.72

Table 4. Three different design suggestions from the design optimization tool for CW type cooling units arranged according to increasing cost.

4. Conclusion and future work

Designing of a DC is a complex process and hence during the bidding process the sales representatives need the help of the engineers. The cycle of communication between the clients, sales representatives and the engineers occupies large working time of the engineers and the sales representatives at the company. This frequent exchange of information between engineers and sales representatives makes the bidding process to be longer, thereby reducing the chance to come up with an efficient design within the deadline. With this tool we can not only optimize the design, but also reduce the wastage of time by the engineers and the sales representatives by creating efficient MDC designs in very short time. This will allow the engineers to spend more time on already approved projects. The fast design generation will allow the sales representatives to provide the clients with an efficient design within the deadline. With several different designs to choose from, the engineers and sales people can instantly jump to choosing the most appropriate design suitable for the client. The tool also shows the key performance parameters to help them finalize the design. It also saves time and effort by designing optimum power redundancy for the modular DCs in a few minutes, which would have otherwise taken significant effort on part of the engineers. This is the first and unique tool to make the design process of MDCs easier. This preliminary design has opened up new branches of possibilities to refine and extend its use several folds by overcoming the limitations.

The foundation for obtaining N+1 redundant cooling system has been laid out in this document with the parameter α and its dependence on various factors. With systematic experiments to validate the model, the values for α could be found out for any system size. This will in turn facilitate the selection of the cooling unit with N+1 redundancy which would be able to cool the IT racks in the case of a cooling failure/maintenance. Another aspect of the design process is optimizing space. The floor space required for the modular DC to be set up is expensive. Hence, the clients have limited space for the required modular DC to be setup. A tool which could minimize or generate a design with the given map of the floor could be very useful and save a lot of time. Sometimes, an optimized design for floor space may not provide the optimized design for cooling. Therefore, a weight factor can be introduced which could be adjusted based on the best interest of the clients. Similar to the cooling units, a CDR can be added for selection of the power units and minimize the total cost of power units with redundancy. A failure of component can happen at any level with in a DC. So, redundant configuration for the chiller systems (if it is CW cooling) can be developed as an addition to the current design tool in the future. Furthermore, to add granularity, sensor (temperature, air-flow, fire detection, etc.) selection and placement in an optimized way can be incorporated into the current design tool.

5. References

[1] M. Dayarathna, Y. Wen, and R. Fan, “Data Center Energy Consumption Modeling: A Survey,” *IEEE Commun. Surv. Tutorials*, vol. 18, no. September, pp. 1–1, 2015.

[2] W. Van Heddeghem, S. Lambert, B. Lannoo, D. Colle, M. Pickavet, and P. Demeester, “Trends in worldwide ICT electricity consumption from 2007 to 2012,” *Comput. Commun.*, vol. 50, pp. 64–76, 2014.

[3] J. Koomey, “Growth in Data Center Electricity use 2005 to 2010,” *Anal. Press.*, pp. 1–24, 2011.

[4] Kim, Y., Gurumurthi, S. and Sivasubramaniam, A., 2006, February. Understanding the performance-temperature interactions in disk i/o of server workloads. In High-performance computer architecture, 2006. The twelfth international symposium on (pp. 176-186). IEEE.

[5] A. . Fallis, “Energy COst, The Key CHallenge of Today’s Data Centers: A Power Consumption Analysis of TPC-C Results,” *J. Chem. Inf. Model.*, vol. 53, no. 9, pp. 1689–1699, 2013.

[6] Y. Gao, H. Guan, Z. Qi, B. Wang, and L. Liu, “Quality of service aware power management for virtualized data centers,” *J. Syst. Archit.*, vol. 59, no. 4–5, pp. 245–259, 2013.

[7] T. Chen, X. Wang, and G. B. Giannakis, "Cooling-Aware Energy and Workload Management in Data Centers via Stochastic Optimization," *IEEE J. Sel. Top. Signal Process.*, vol. 10, no. 2, pp. 402–415, 2016.

[8] R. Khalid, Y. Joshi, and A. Wemhoff, "Rapid modeling tools for energy analysis of modular data centers," *Proc. 15th Intersoc. Conf. Therm. Thermomechanical Phenom. Electron. Syst. ITherm 2016*, pp. 1444–1452, 2016.

[9] A. Bhalerao *et al.*, "Numerical Heat Transfer , Part A : Applications Rapid prediction of exergy destruction in data centers due to airflow mixing," vol. 7782, no. May, 2016.

[10] S. A. Nada, M. A. Said, and M. A. Rady, "CFD investigations of data centers' thermal performance for different configurations of CRACs units and aisles separation," *Alexandria Eng. J.*, vol. 55, no. 2, pp. 959–971, 2016.

[11] Z. Song, "Numerical cooling performance evaluation of fan-assisted perforations in a raised-floor data center," *Int. J. Heat Mass Transf.*, vol. 95, pp. 833–842, 2016.

[12] J. W. VanGilder, Z. M. Pardey, P. Bemis, and D. W. Plamondon, "Compact modeling of data center raised-floor-plenum stanchions: Pressure drop through sparse tube bundles," *Proc. 15th Intersoc. Conf. Therm. Thermomechanical Phenom. Electron. Syst. ITherm 2016*, pp. 1148–1155, 2016.

[13] M. V. D. Berge *et al.*, “CoolEmAll - Models and tools for optimization of data center energy-efficiency,” *Sustain. Internet ICT Sustain.*, pp. 1–5, 2012.

[14] S. W. Ham, J. S. Park, and J. W. Jeong, “Optimum supply air temperature ranges of various air-side economizers in a modular data center,” *Appl. Therm. Eng.*, vol. 77, pp. 163–179, 2015.

[15] S. W. Ham, M. H. Kim, B. N. Choi, and J. W. Jeong, “Energy saving potential of various air-side economizers in a modular data center,” *Appl. Energy*, vol. 138, pp. 258–275, 2015.

[16] B. Durand-Estebe, C. Le Bot, J. N. Mancos, and E. Arquis, “Simulation of a temperature adaptive control strategy for an IWSE economizer in a data center,” *Appl. Energy*, vol. 134, pp. 45–56, 2014.

[17] K. Fouladi, A. P. Wemhoff, L. Silva-Llanca, and A. Ortega, “Optimization of Data Center Cooling Efficiency Using Reduced Order Flow Modeling Within a Flow Network Modeling Approach,” *Proc. ASME 2014 Int. Mech. Eng. Congr. Expo.*, no. November 2014, 2014.

[18] Lui, Y.Y., 2010. Waterside and Airside Economizers Design Considerations for Data Center Facilities. ASHRAE Transactions, 116(1).

[19] Ham, S.W., Kim, M.H., Choi, B.N. and Jeong, J.W., 2015. Energy saving potential of various air-side economizers in a modular data center. Applied Energy, 138, pp.258-275.

[20] “coolemall.eu.”[Online].Available: http://tricoryne.man.poznan.pl/web/guest/download/-/document_library_display/g2Kj/view/57456.

[21] J. Gao and R. Jamidar, “Machine Learning Applications for Data Center Optimization,” *Google White Pap.*, pp. 1–13, 2014.

[22] V. Lopez, H. F. Hamann, and V. López, “Measurement-based modeling for data centers,” *2010 12th IEEE Intersoc. Conf. Therm. Thermomechanical Phenom. Electron. Syst.*, pp. 1–8, 2010.

[23] Economou, D., Rivoire, S., Kozyrakis, C. and Ranganathan, P., 2006. Full-system power analysis and modeling for server environments. International Symposium on Computer Architecture-IEEE.

[24] D. Economou, S. Rivoire, C. Kozyrakis, and P. Ranganathan, “Full-System Power Analysis and Modeling for Server Environments,” *Work. Model. Benchmarking Simul.*, no. 3, pp. 807–812, 2006.

[25] M. P. Karpowicz, P. Arabas, and E. Niewiadomska-Szynkiewicz, “Design and implementation of energy-aware application-specific CPU frequency governors for the heterogeneous distributed computing systems,” *Futur. Gener. Comput. Syst.*, 2015.

[26] I. Mitrani, “Managing performance and power consumption in a server farm,” *Ann. Oper. Res.*, vol. 202, no. 1, pp. 121–134, 2013.

[27] D. Meisner, B. T. Gold, and T. F. Wenisch, "PowerNap: eliminating server idle power," *Asplos*, vol. 44, pp. 205–216, 2009.

[28] J. Meng *et al.*, "Communication and cooling aware job allocation in data centers for communication-intensive workloads," *J. Parallel Distrib. Comput.*, vol. 96, pp. 181–193, 2016.

[29] C. S. Lengsfeld and R. A. Shoureshi, "MANAGING WORKLOAD AT A DATA CENTER," vol. 1, no. 19, 2008.

[30] A. M. A. S. Almoli, "Ali M A S Almoli Submitted in accordance with the requirements for the degree of Doctor of Philosophy The University of Leeds School of Mechanical Engineering Work Formed from Jointly Authored Publication," 2013.

[31] Tatchell-Evans, M., Kapur, N., Summers, J., Thompson, H. and Oldham, D., 2017. An experimental and theoretical investigation of the extent of bypass air within data centres employing aisle containment, and its impact on power consumption. Applied Energy, 186, pp.457-469.

6. Appendix

6.1 MatLab code for Optimization based on CFD

```
%% Version 12
    % removing tail error in the sequence
```

```
%   figuring out the length problem
clear
clc
close all
% tic
%% Defined Values
%p = [20 30 40]; % Same length for p,c and s is expected
%c = [20 30 40];
%s = [8 12 16];
% p = [20]; % Same length for p,c and s is expected
% c = [15];
% s = [5];
p = [0 50 100]; % Same length for p,c and s is expected
c = [20 25 37];
s = [0 5 10];
cop = [4 4 4];
length = 17;
totalit = 80;
%%%
%cop = [2 3 4];
cpc = c./cop;    % Cooling power consumption
tp = [size(p,2) size(c,2) size(s,2)];
param = 4;
types = max(tp);  % max types in any set p,c or s
%length = 12; % # total racks to be installed
%totalit = 80;
%% Matrix Formulation
```

```
x = ones(length,3,types);
mat = zeros(length,4,types); %transforms p,c,s,cpc to 3-D
% matrix to mul with x
mat(:,1,:) = p(ones(1,length),:);
mat(:,2,:) = c(ones(1,length),:);
mat(:,3,:) = s(ones(1,length),:);
mat(:,4,:) = cpc(ones(1,length),:); %transformed
% Cooling Matrix
% Cil Matrix parameters
sigma = 1;
% Cil = zeros(length,length);
% Cilk = zeros(length,length,types);
for i = 1:length
    for l = 1:length
        csdistance = abs(l-i);
%         Cil(i,l) = ((erf(abs(l-i)/(sigma*sqrt(2))))-
(erf(abs(abs(l-i)-1))/(sigma*sqrt(2))))/2;
        if csdistance <= 3
            Cil(i,l) = 1/6;
        else
            Cil(i,l) = 0;
        end
%         Cil(i,l) = ((erf(csdistance/(sigma*sqrt(2))))-
(erf(abs(csdistance-1)/(sigma*sqrt(2)))))/2;
%           syms k; (for series calculation)
%           Cil(i,l) = (symsum(((-1)^k)*((abs(l-
i)))^(2*k+1)/((factorial(k)*(2*k+1))),k,0,5))/sqrt(pi);
```

```
%           Cil(i,l) = Cil(i,l) - (symsum(((-1)^k)*((abs(l-i)-
1))^(2*k+1)/((factorial(k)*(2*k+1))),k,0,5))/sqrt(pi);
        for k = 1:types
            if Cil(i,l) > 0
                Cilk(i,l,k) = c(k).*Cil(i,l);
            else
                Cilk(i,l,k) = 0;
            end
        end
    end
end

%% Objective Functioin
% total P + total C - 2*total S
obj_mat= mat(:,1:3,:);
obj_mat(:,2,:) = mat(:,2,:) - mat(:,4,:);
obj_mat(:,3,:) = obj_mat(:,3,:).*(-2);
f = obj_mat;
% integer variables
intcon = 1:(length*3*types);

%% Constraints
% server constraint A <=b form
    % min limit of server power capacity
server_min = zeros(length,3,types);
server_min(:,3,:) = mat(:,3,:).*(-1); % coeff of x for
% server power constr.
```

```
b = -totalit;
A = server_min(:)';
    % max limit of server power capacity
server_max = zeros(length,3,types);
server_max(:,3,:) = mat(:,3,:);
b = [b totalit*1.1]; % max limit 10% greater than defined
A = [A; server_max(:)'];
% % cooling constraint A <=b form
% cooling = zeros(length,3,types);
% cooling(:,2,:) = mat(:,2,:).*(-1); % coeff of x for cooling cap.
constr.
% cooling(:,3,:) = mat(:,3,:);
% b = [b 0];
% A = [A; cooling(:)'];
% ------------Cooling Constraints------------
for l = 1:length
    cc{l} = zeros(length,3,types);
    cc{l}(l,3,:) = mat(l,3,:);
    cc{l}(:,2,:) = -(Cilk(:,l,:));
    b = [b 0];
    A = [A; cc{l}(:)'];
end
% power constraint A <=b form
power = zeros(length,3,types);
power(:,1,:) = mat(:,1,:).*(-1); % coeff of x for cooling
% cap. constr.
power(:,3,:) = mat(:,3,:);
```

```
power(:,2,:) = mat(:,4,:);
b = [b 0];
A = [A; power(:)'];
% one unit at one position
for i=1:length
    one_item{i} = zeros(length,3,types);
    one_item{i}(i,:,:) = 1;
    b = [b 1];
    A = [A; one_item{i}(:)'];
end
% % Capacity count of the cooling unit (for recursive addition
% and subtraction of cooling and servers)
% for i=1:length
%     c_cap{i} = zeros(length,3,types);
%     c_cap{i}(1:i,2,:) = -mat(1:i,2,:);
%     c_cap{i}(1:i,3,:) = mat(1:i,3,:);
%     b = [b 0];
%     A = [A; c_cap{i}(:)'];
% end
% bounds
lb = zeros(length,3,types);
ub = ones(length,3,types);
% Equality constraint
Aeq = zeros(1,length*3*types);
beq = 0;
%% running the MIP
result = intlinprog(f,intcon,A,b,Aeq,beq,lb,ub)
```

```
result = reshape(result,length,3,types);

result = round(result);

config = mat(:,1:3,:).*result;

sumtypes = obj_mat.*result;

deltapc = sum(sumtypes(:))

%% Post Processing Data

linseq = [];

lincap = [];

for i = 1:length

    for k = 1:types

        [row,col,v] = find(config(i,:,k));

        linseq = [linseq col];

        lincap = [lincap v];

    end

end

sequence = [linseq;lincap];

sequence = num2cell(sequence);

Loads = sequence;

Ptrim = [];

for l = 1:(numel(sequence))/2

    if sequence{1,l} == 1

%         sequence{1,l} = 'P';

        Ptrim = [Ptrim l];

    elseif sequence{1,l} == 2

        sequence{1,l} = 'C';

        Loads{2,l} = Loads{2,l}/4 % divide by COP to get

% cooling power consumption
```

```
    elseif sequence{1,l} == 3
        sequence{1,l} = 'S';
    end
end
sequence(:, Ptrim) = [];
Loads(:, Ptrim) = [];
sequence
Loads = cell2mat(Loads(2,:));
[Pconnection, Pone] = Predundant(Loads, p);
Pconnection
Pone
% toc
xlswrite('sequenceConfig',sequence);
```

6.2 MatLab code for N+1 redundancy for power

```
function [xij, Pone] = Predundant(Loads, Ptypes)
% clear
% clc
% Loads = [5 5 10 10];
% Ptypes = [15 20 25];
Pone = max(Ptypes((Ptypes <= sum(Loads))));
if numel(Pone) == 0
    Pone = min(Ptypes);
end
np = ceil(sum(Loads)/Pone) + 1;
flag = true;
while flag
```

```
    flag = false;
    P = Pone*ones(1,np);
    xij = zeros(np,numel(Loads));
    Tik = zeros(np);
    Rik = diag(100*ones(1,np));
    A = zeros(2,np);
    A1 = 1;
    for j = 1:numel(Loads)
        C = 0;
        for i = 1:np
            if (Loads(j) + max(Tik(i,:))) <= P(i)
                P(i) = P(i) - Loads(j)/2;
                C = C + 1;
                xij(i,j) = 1;
                Rindex = 1:np;
                Rvalues = Rik(i,:);
                A = [Rindex; Rvalues]';
                A = sortrows(A,2);
                A = A';
                flag2 = true;
                while flag2
                    flag2 = false;
                    q = A(1,A1);
%                       [q_correspondingPa, q_correspondingPb] =
find(Tik(q,:) >= max(Tik(q,:)));

                        if Loads(j) + Tik(i,q) <= P(q)
```

```
                P(q) = P(q) - Loads(j)/2;
                C = C + 1;
                xij(q,j) = 1;
            else
                if A1 == (max(A(1,:)) - 1)
                    np = np + 1;
                    flag = true;
                    break;
                else
                    A1 = A1 + 1;
                    q = A(1,A1);
                    flag2 = true;
                end
            end
        end

        if C == 2
            Rindex = find(xij(:,j));
            Tindex = find(xij(:,j));
            Tik(Tindex(1),Tindex(2)) = Tik(Tindex(1),
Tindex(2)) + Loads(j)/2;
            Tik(Tindex(2), Tindex(1)) =
Tik(Tindex(1),Tindex(2));
            Rik(Rindex(1), Rindex(2)) = Rik(Rindex(1),
Rindex(2)) + 1;
            Rik(Rindex(2), Rindex(1)) = Rik(Rindex(1),
Rindex(2));
```

```
                break;


            else
                fprintf('Code Error\n');


            end




           else
            if i == np
                flag = true;
                np = np + 1;
                break;
            end
        end
        if flag == true
            break;
        end
    end
    if flag == true
        break;
    end
end
end
% Pone;
```

```
% xij;
```

6.3 MatLab code for optimization based on experiment – CW

```
%% Given
Tinit = 40;
T = Tinit; % total IT
Rinit = 7;
R = Rinit; % number of racks
Mspkw = 128; % CFM consumed per kW
Ms = Mspkw*(T/R); % CFM consumed by each rack
Alpr = T/R; % Average load per rack

%% inventory
Tref = 30; % celcius
Ts = 26;
Tc = 19;
W = 0.13; % width of the cold aisle in meters

%Ci = [10; 2500; 100]';
load 'matlab.mat';
C = Ci;
sizeC = size(C);

%-------Cr calculation-----------
r = zeros(sizeC(1),1);
```

```
y = (Ts-Tc)/(Tref-Tc);
for Cn = 1:sizeC(1)
    x1 = Ms/C(Cn,2);
    for ri = 1:20
        x2 = ri/W;
        x3 = ri;
%          yhat = (-0.33174.*x3+-0.10849.*x2+0.019741.*x2.*x3+-
2.1302.*x1+0.54694.*x1.*x3+0.16893.*x1.*x2+0.8446.*1+
1.9811.*x1.^2+0.0014718.*x2.^2+0.023356.*x3.^2);
        yhat = (+-0.27365.*x3+-
0.075009.*x2+0.0075848.*x2.*x3+1.1805.*x1+-
0.25454.*x1.*x3+0.061221.*x1.*x2+0.0084666.*1+
0.0091754.*x1.^2+0.00146.*x2.^2+0.056652.*x3.^2);
        if y >= yhat
            continue;
        else
            r(Cn) = ri-1;
            break;
        end
    end
end
%---------XXXXXX----------------
%Cr = (((Ts-Tc)/(Tref-Tc))*(W/(Ms^2))*(C(:,2).^2) + 0.3)/0.6;
 % Cooling range
%T1 = ((((1*0.6-0.3)/(C(:,2).^2))/(W/(Ms^2)))*(Tref-Tc)) + Tc;
```

```
Cr = round(r);
C = [C Cr];

for n = 1:sizeC(1)
    loadp = [0 0.5 1];
    fpower = [ 0 0.75 1.5];
    qfpower = interp1(loadp, fpower, (C(n,4)*Ms)/(C(n,2)));
    fanCost = min(qfpower*0.09*24*365*5, fmaxpower(n,1));
    C(n,3) = fanCost + C(n,3);
end

Cpr = C(:,3)./C(:,4); % cost per rack
C = [C Cpr];
C = sortrows(C, 5);
Cinit = C;

KPI = []; % Total IT over total cooling
%% Algorithm
S{4,1} = 0;
for rank = 1:3
    r = C(:,4); % extracting range of each cooling unit
    while R > r(1)
        S{rank} = [S{rank}; C(1,:)];
        R = R - r(1);
    end
```

```
    C = sortrows(C, 3); % sorting cooling units based on price
    r = C(:,4); % extracting range of each cooling unit
    count = 1;
    while R > 0
        if R <= r(count)
            S{rank} = [S{rank}; C(count,:)];
            R = R - r(count);
        else
            count = count +1;
        end
    end
    R = Rinit;% reset R
    C = sortrows(C, 5);% Rearrange C matrix accourding to
% increasing price/rack
    [delrow, delcol] = find(C==S{rank}(1,3),1);% delete the
% cooling unit used most
    % searching for the price of that product in master
% list and delete
    % that unit from inventory list
    C(delrow,:) = [];
    KPI(rank,1) = T/sum(S{rank}(:,1)); % Total IT/Total
% cooling


end
```

```
% Rearrange Selectionlist and KPI
Stemp = S;
S = [];
Stempcost = (1:3)';
for n = 1:3
    Stempcost(n,2) = sum(Stemp{n}(:,3));
end
Stempcost = sortrows(Stempcost, 2);

S{4,1} = 0;
for n = 1:3
    S{n} = Stemp{Stempcost(n,1)};
end

KPItemp = KPI;
for n = 1:3
    KPI(n) = KPItemp(Stempcost(n,1));
end
%% Chiller selection
Ecostpkw = 0.09*24*365*5;
% Chiller = [1 2 3 4; 1 2 3 4; 1 2 3 4]; % kW IPLV FCOP $
nC = T./Chiller(:,1);
TotalCost = ceil(nC).*Chiller(:,4) +
floor(nC).*((Chiller(:,1))./Chiller(:,3)).*Ecostpkw + (ceil(nC)
 - floor(nC)).*((min(T, T -
floor(nC).*Chiller(:,1)))./Chiller(:,2)).*Ecostpkw;
```

```
for n = 1:numel(nC)
    if nC(n) > 1
        Chiller(n,5) = TotalCost(n)./nC; % to make the
% cost per chiller
    else
        Chiller(n,5) = TotalCost(n);
    end
end

Chiller(:,6) = Chiller(:,5)./Chiller(:,1);
Chiller = sortrows(Chiller, 6); % sorting chiller based on
% cost per kw



ChillerS{4,1} = 0;
for rank = 1:3
    Chload = Chiller(:,1); % extracting range of each cooling
% unit
    while T > Chload(1)
        ChillerS{rank} = [ChillerS{rank}; Chiller(1,:)];
        T = T - Chload(1);
    end

    Chiller = sortrows(Chiller, 5); % sorting cooling units
% based on price
    Chload = Chiller(:,1); % extracting capacity of each chiller
```

```
    count = 1;
    while T > 0
        if T <= Chload(count)
            ChillerS{rank} = [ChillerS{rank}; Chiller(count,:)];
            T = T - Chload(count);
        else
            count = count +1;
        end
    end
    T = Tinit;% reset T
    Chiller = sortrows(Chiller, 6);% Rearrange Chiller matrix
% accourding to increasing price/kW
    [delrow, delcol] = find(Chiller==ChillerS{rank}(1,6),1);
% delete the cooling unit used most
    % searching for the price of that product in master list
% and delete that unit from inventory list
    Chiller(delrow,:) = [];
    ChillerKPI(rank,1) = T/sum(ChillerS{rank}(:,1));
% Total IT/Total cooling

end

% Rearrange chiller Selectionlist and KPI
ChillerStemp = ChillerS;
ChillerS = [];
ChillerStempcost = (1:3)';
for n = 1:3
```

```
    ChillerStempcost(n,2) = sum(ChillerStemp{n}(:,5));
end
ChillerStempcost = sortrows(ChillerStempcost, 2);


ChillerS{4,1} = 0;
for n = 1:3
    ChillerS{n} = ChillerStemp{ChillerStempcost(n,1)};
end


ChillerKPItemp = ChillerKPI;
for n = 1:3
    ChillerKPI(n) = ChillerKPItemp(ChillerStempcost(n,1));
end
```

6.4 MatLab code for optimization based on experiment – DX

```
%% Given
Tinit = 40;
T = Tinit; % total IT
Rinit = 7;
R = Rinit; % number of racks
Mspkw = 128; % CFM consumed per kW
Ms = Mspkw*(T/R); % CFM consumed by each rack
Alpr = T/R; % Average load per rack


%% inventory
```

```
Tref = 30; % celcius
Ts = 26;
Tc = 19;
W = 0.13; % width of the cold aisle in meters

%Ci = [10; 2500; 100]';
load 'DX.mat';
C = DX;
sizeC = size(C);

%-------Cr calculation-----------
r = zeros(sizeC(1),1);
y = (Ts-Tc)/(Tref-Tc);
for Cn = 1:sizeC(1)
    x1 = Ms/C(Cn,2);
    for ri = 1:20
        x2 = ri/W;
        x3 = ri;
%          yhat = (-0.33174.*x3+-0.10849.*x2+0.019741.*x2.*x3+-
2.1302.*x1+0.54694.*x1.*x3+0.16893.*x1.*x2+0.8446.*1+
1.9811.*x1.^2+0.0014718.*x2.^2+0.023356.*x3.^2);
        yhat = (+-0.27365.*x3+-
0.075009.*x2+0.0075848.*x2.*x3+1.1805.*x1+-
0.25454.*x1.*x3+0.061221.*x1.*x2+0.0084666.*1+0.0091754.*x1.^2+
0.00146.*x2.^2+0.056652.*x3.^2);
        if y >= yhat
            continue;
```

```
        else
            r(Cn) = ri-1;
            break;
        end
    end
end
%---------XXXXXX----------------
%Cr = (((Ts-Tc)/(Tref-Tc))*(W/(Ms^2))*(C(:,2).^2) + 0.3)/0.6;
% Cooling range
%T1 = ((((1*0.6-0.3)/(C(:,2).^2))/(W/(Ms^2)))*(Tref-Tc)) + Tc;
Cr = round(r);
C = [C Cr];
for n = 1:sizeC(1) % to make sure that the total IT load is
% less than cooling capacity
    if C(n,1) >= Alpr*C(n,6)
        continue;
    else
        C(n,6) = floor(C(n,1)/Alpr);
    end
end
[rowsC colC] = size(C);

for n = 1:rowsC
    loadp = [0 0.6 1];
    cop = [ 0 C(n, 4:5)];
    qcop = interp1(loadp, cop, (C(n,6)*Alpr)/(C(n,1)), 'spline');
    compressorCost = ((C(n,6)*Alpr)/qcop)*0.09*24*365*5;
```

```
    C(n,7) = compressorCost + C(n,3);
end

Cpr = C(:,7)./C(:,6); % cost per rack
C = [C Cpr];
C = sortrows(C, 8);
Cinit = C;

KPI = []; % Total IT over total cooling
%% Algorithm
S{4,1} = 0;
for rank = 1:3
    r = C(:,6); % extracting range of each cooling unit
    while R >= r(1)
        S{rank} = [S{rank}; C(1,:)];
        R = R - r(1);
    end

    [rowsC colC] = size(C);
    for n = 1:rowsC
        loadp = [0 0.6 1];
        cop = [ 0 C(n, 4:5)];
        if R <= C(n,6)
            qcop = interp1(loadp, cop, (R*Alpr)/(C(n,1)),
'spline');
        else
```

```
            qcop = interp1(loadp, cop, (C(n,6)*Alpr)/(C(n,1)),
'spline'); % no need to change the cost
        end
        compressorCost = ((R*Alpr)/qcop)*0.09*24*365*5;
        C(n,9) = compressorCost + C(n,3);
    end

    C = sortrows(C, 9); % sorting cooling units based on new
% price of cooling for the remaining racks
    r = C(:,6); % extracting range of each cooling unit
    count = 1;
    while R > 0
        if R <= r(count)
            S{rank} = [S{rank}; C(count,1:8)];
            R = R - r(count);
            S{rank}(end,7) = C(count,9);
        else
            count = count +1;
        end
    end
    R = Rinit;% reset R
    C(:,9) = []; % delete the revised price of cooling unit
% based on no. of racks remaining
    C = sortrows(C, 8);% Rearrange C matrix accourding to
% increasing price/rack
    [delrow, delcol] = find(C==S{rank}(1,8),1);% delete the
% cooling unit used most
```

```
    % searching for the price of that product in master list
% and delete
    % that unit from inventory list
    C(delrow,:) = [];
    KPI(rank,1) = T/sum(S{rank}(:,1)); % Total IT/Total cooling



end


% Rearrange Selectionlist and KPI
Stemp = S;
S = [];
Stempcost = (1:3)';
for n = 1:3
    Stempcost(n,2) = sum(Stemp{n}(:,7));
end
Stempcost = sortrows(Stempcost, 2);


S{4,1} = 0;
for n = 1:3
    S{n} = Stemp{Stempcost(n,1)};
end


KPItemp = KPI;
for n = 1:3
    KPI(n) = KPItemp(Stempcost(n,1));
```

```
end
```

6.2 Component data repository of cooling units used

6.2.1 Chilled water cooling units

Model and Company	Capacity (kW)	Air flow rate (CFM)	TCO 5 years (CAD)	Price (CAD)
DataAire GICW-03012	24.45	3000	18990.07	14492.25
DataAire GICW-06034	62.95	4600	34483.03	22902.75
Qcooling CL80-In Row 48U x 300	55	4237.7	22516.2	12398.4
Stulz CW CRS-090-C	26	2900	16232.96	11450
Stulz CW CRS-0180-C	52	5800	29365.92	19800

6.2.2 Direct expansion cooling units

Model	Cost	Air flow rate (CFM)	Capa city (kW)	Part Load COP	Full Load COP	Outdoor unit	Power consumption (kW)	Cost (CAD)
APC ACRD100	11535	2290	10.62	2.57	2.83	ACCD75214	0.85	4671
APC ACRD500	25348 .94	4600	33.7	2.28	2.79	ACCD75201	1.23	5500
Liebert CRV019	20811	2250	19	3.83	3.3	MCS028	0.53	5100
ClimaVeneta CRCX-I 0051	5336	882	10.6	3.73	3.43	i-HCAT for 0051	0.26	3965

ClimaVeneta CRCX-I 0071	6394	1589	16.6	3.25	3.03	i-HCAT for 0071	0.6	4914

6.2.3 Chiller units

Model	Capacity(kW)	PLCOP	COP	Price (CAD)
Carrier 30RAP0155K-58F10	49.2	3.8	2.9	50595
Trane CGAM-C20	70.3	4.2	3.0	74565
STULZ CSO 541 ASN	57	2.9	3.5	60000
Johnson Controls YCAL0019EE	52.8	4.3	3.0	51000