Report for **AdeptDC's SBIR Phase I**

Real-time Software-based Cooling Optimization for Data Centers

This report describes the research accomplishments of AdeptDC's SBIR Phase I project. The objective of the project was to verify a hypothesis that data center cooling costs would be significantly reduced if cooling were dynamically regulated based on chip level thermal data (such as motherboard temperature or CPU temperature). AdeptDC developed a prototype software appliance (SA) and conducted proof-of-concept (POC) studies at the Data Center Laboratory at Georgia Tech (GT DCL) in Atlanta, GA and Arrow Data Center in Alpharetta, GA (beta customer data center).

Summary of Phase I Research Activities

This summary is organized based on the R & D objectives and technical milestones in the SBIR Phase I proposal, as identified in Table 1.

Table 1: R & D Objectives and Technical Milestones Proposed in SBIR Phase I Project

R & D Objectives	Technical Milestones
1. Minimal viable SA Development	(a) IT component temperature data extraction system (b) Training database (c) Filter (d) Real-time analytics (e) Dashboard (f) Cooling automation system
2. Component Testing	(a) Testing (b) Test result database compilation and adjustments
3. Integration	(a) Developing Bridging protocols
4. System-level Testing	(a) Laboratory-scale deployment (b) Exhaustive testing (c) Test result compilation and refinement
5. Performance Benchmarking	(a) Live deployment (b) Continuous performance monitoring (c) Evaluation and refinement

1. Minimal Viable SA Development

As shown in Figure 1, the operational sequence for the SA includes the following steps:

Step 1: IT component temperatures and cooling equipment set points are read by Collector application.

Step 2: Collector application interacts with Configuration Service.

Step 3: Collector application stores extracted data in Database Service.

Step 4: Configuration and Database Services transfer data to the Control Algorithm.

Step 5: The Control Algorithm determines optimal cooling set points.

Steps 6-8: The optimal cooling set points are sent to the corresponding cooling equipment

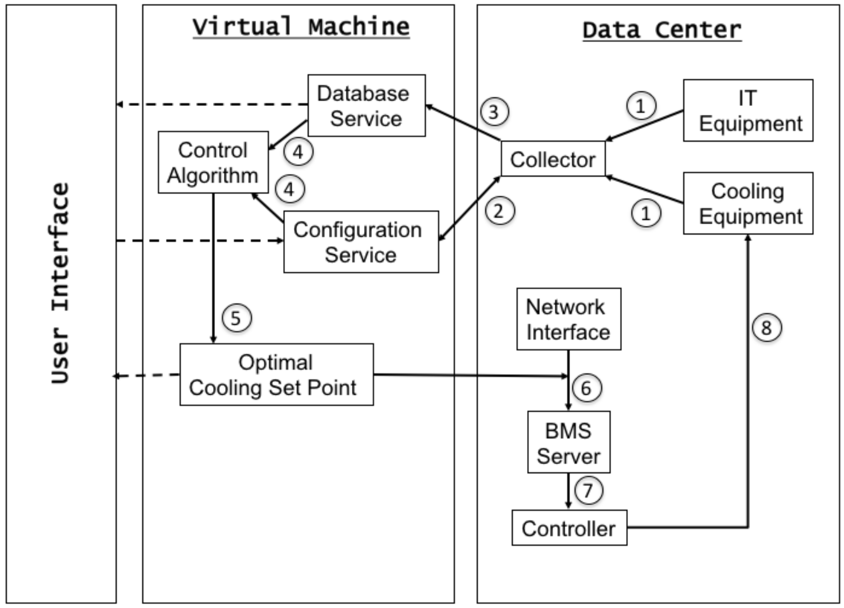


Figure 1: Sequence Diagram for AdeptDC's SA

(a) IT component temperature data extraction system:

IT component temperature data extraction system is an integral part of Collector application. It uses intelligent platform management interface (IPMI) and simple network management protocol (SNMP). The generic IPMI command to pull out server component (motherboard, CPU, memory) temperatures uses lanplus interface and sensor data record (sdr):

- ipmitool -I lanplus -H < IP Address> -U <username> –P <password> sdr list full

For networking switches and storage disks, 'snmpwalk' command was used to collect the component temperature data:

- snmpwalk -v 2c -c <community string> <IP Address> <OID>

The "OID" for a particular temperature sensor was determined from the manufacturer's SNMP management information base (MIB).

The IP addresses, login credentials, SNMP community strings, and SNMP object identifiers (OID-s) were fed into a Node.js based application which periodically queries IP addresses and populates a SQL database. The collected data could be visualized real-time either in a custom-made user interface or in a prevalent big data platform such as Splunk.

Figure 2 shows the screenshot of a custom-made visualization windows for the dynamic temperature profiles of the motherboard, CPU, and memory for a HP ProLiant server at GT DCL. Figure 3 shows the screenshots of Splunk dashboards for visualizing dynamic temperatures of different IT components in IBM System x servers at Arrow data center. Figure 4 shows the screenshot of Splunk dashboard for visualizing dynamic temperature profiles of field programmable gateway arrays (FGPAs) inside IBM Flashsystems 840 and 900 located in Arrow Data Center. Figure 5 shows the screenshots of Splunk dashboards for visualizing panel temperatures of Brocade Fibre DCX and Cisco Nexus switches in Arrow Data Center.

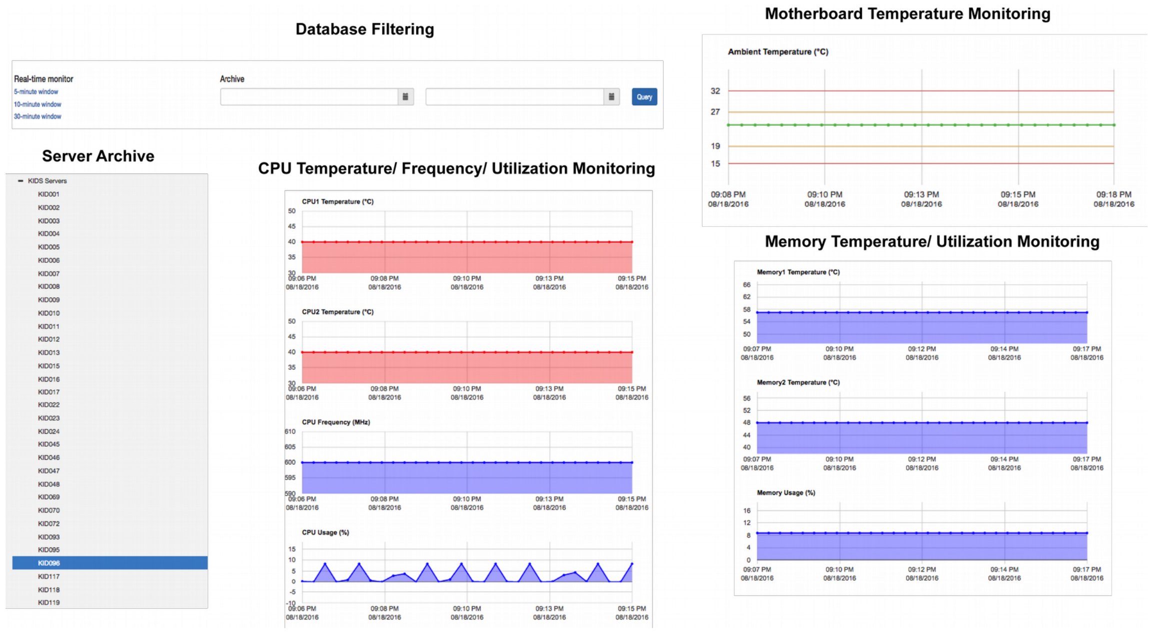


Figure 2: Visualization of HP ProLiant Server Component Temperatures (^{0}C) in GT DCL

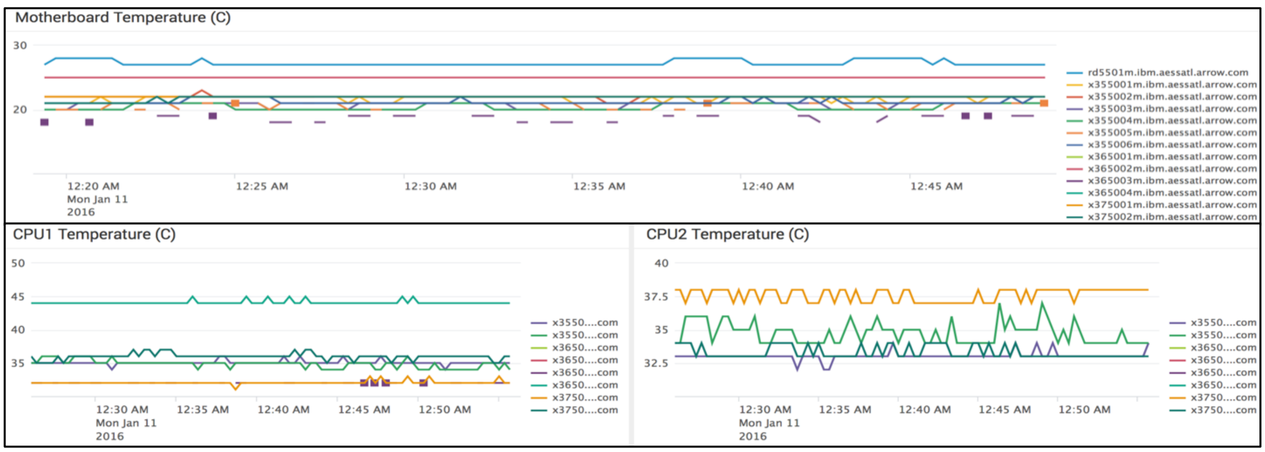


Figure **3**: Splunk Dashboard for IBM Server Component Temperatures (^{0}C) in Arrow Data Center

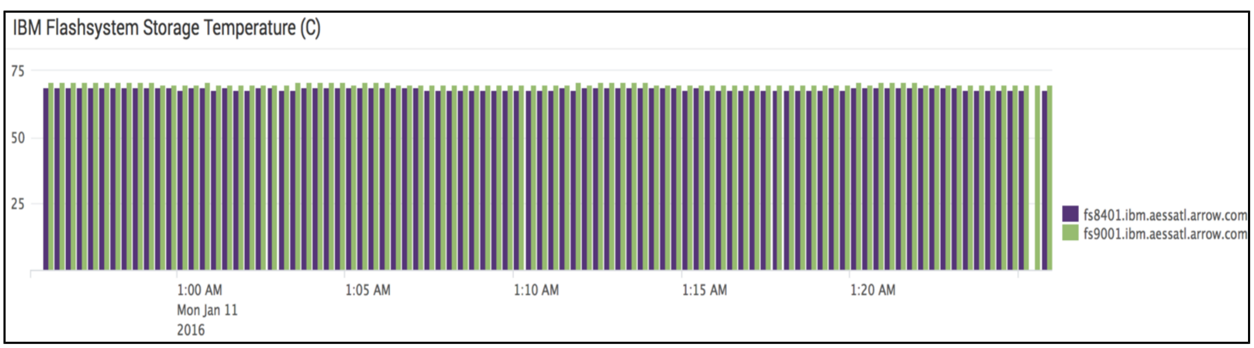


Figure 4: Splunk Dashboard for IBM Flash Systems 840 and 900 FPGA Temperatures (^{0}C) at Arrow Data Center

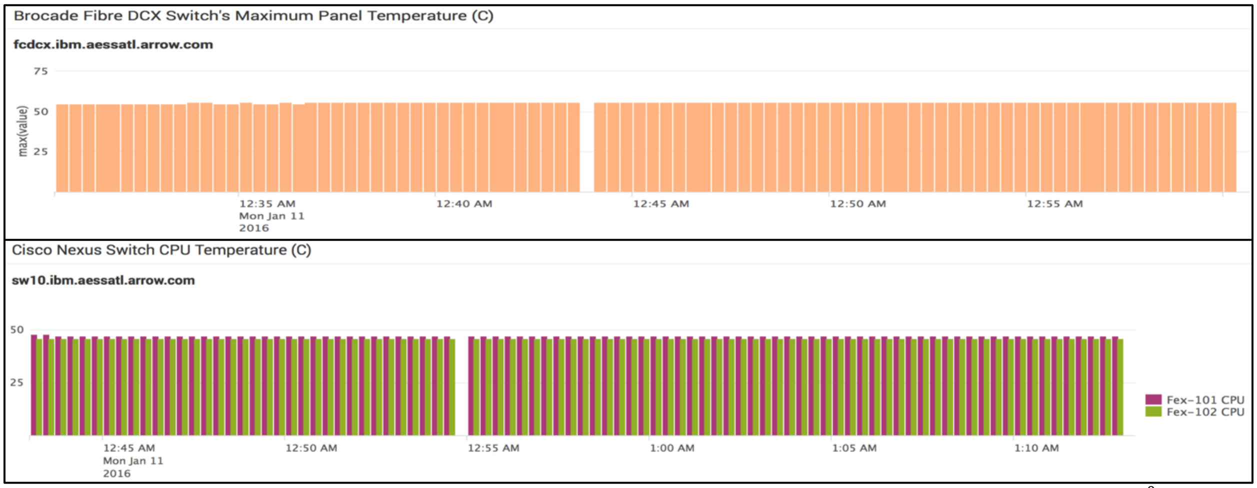


Figure 5: Splunk Dashboard for Brocade Fibre DCX and Cisco Nexus Switch Temperatures (^{0}C) at Arrow Data Center

(b) Training Database

The Collector application and Database service record a SQL-type database with the following fields: manipulated variables such as cooling set points, process variable such as maximum motherboard temperature, operator choices on process variable (maximum motherboard/CPU/memory temperature), testing parameters such as CPU frequency dip percentage and power usage effective (PUE), disturbance variables such as PDU power, chiller power, computer room air conditioning (CRAC) unit power, and predictive computation results. A sub-database of the compiled database, chosen based on user-defined filter criteria, is used as the training data for predictive control sequence and related post-processing.

(c) Filter

Filter—a part of the configuration service—enables users to set the criteria based on which a sub-database is populated matching user specifications. The sub-database is used as the training data for predictive control algorithm. Figure 6 shows the screenshot of software-enabled database filtering tool deployed at GT DCL, with process variable, safety limit, starting CRAC temperature set point, and starting rear door heat-exchanger (RDHx) pressure set point as the search criteria. [RDHx is a water-cooled device, mounted on the rear of a rack, to cool hot exhaust air coming out from IT equipment inside the rack.]

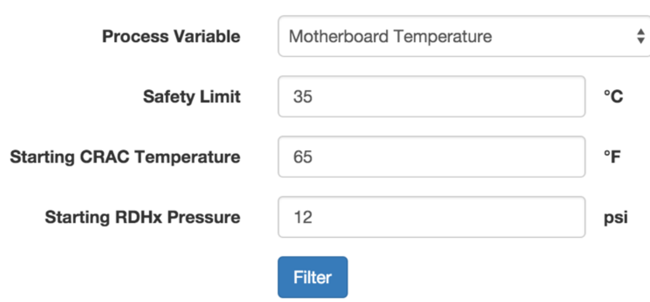


Figure 6: Dashboard for Software-enabled Database Filter Tool

(d) Real-time Analytics Design

Figure 7(a) shows the real-time analytics framework that is implemented by AdeptDC's optimal control mechanism. The objective function of the optimal control system is to reduce cooling energy consumption and the associated constraint is to keep IT equipment safe from overheating. Two major considerations of the controller design were fidelity and efficiency of the data-driven control algorithm. The real-time analytics is implemented in two different modes: reactive and predictive. In the reactive mode, the cooling set points are adjusted based on the real-time feedback of the process variable. The real-time processing enhances fidelity at the expense of processing speed. While operating in the reactive mode, a database with process variable, manipulated variables, and disturbance variables is compiled. The database feeds the necessary training data to the machine learning-based control algorithm to implement the predictive control sequence. It gradually reduces the dependency on real-time data for the process variable computation. The superior efficiency of the predictive control sequence comes at the cost of degraded fidelity, necessitating additional measures. The deviations between run-time and predictive values of process variable are measured at a pre-assigned check interval. Should the deviation be more than an operator-specified limit, the control action gears back to the reactive mode, as shown in Figure 7(b). The overall control system operates like a self-learning artificial intelligence (AI) system.

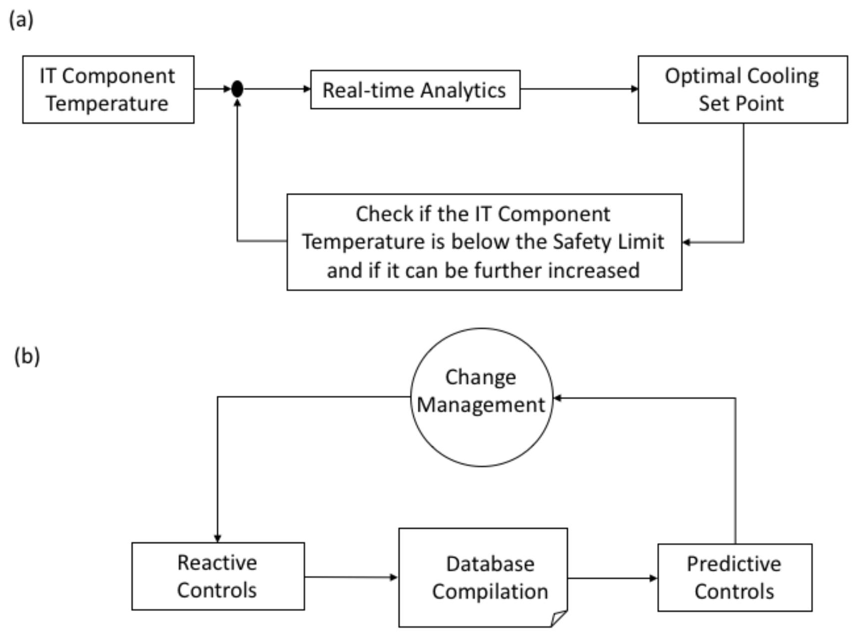


Figure 7: (a) Real-time Analytics-based Control Framework (b) Transition between Reactive and Predictive Control Sequences

Control Sequence in the Reactive Mode

The reactive control sequence involves the following steps:

Step 1: Determine Delta, defined as the difference between the Threshold and the run-time values of the process variable (maximum temperature of a chosen IT component):

$$\Delta = T_{Thresold} - T_{Runtime} \qquad \textbf{Eq. 1}$$

Step 2: Change cooling equipment set points such that overall cooling energy consumption is reduced. It means increasing temperature set points and reducing pressure set points. A proportional-integral-derivative (PID) controller is implemented for modulating set points of high-power cooling equipment, such as CRAC unit. With Δ as the process variable, the governing equation for PID control algorithm is:

$$\Omega = K_p \Delta + K_i \int_0^t \Delta(\tau)\, d\tau + K_d \frac{d\Delta}{dt} \qquad \textbf{Eq. 2}$$

A few factors make this controller operation difficult are:

- The absolute zero tolerance for overshooting (process variable breaching the Threshold)
- The non-linear nature of forced convective air-cooling in data centers with long response time
- Disconnect between IT load and cooling

To reduce the risk of overshooting, the integral and derivative gains are set to be equal to zero: $K_i = 0$ and $K_d = 0$. The choice of proportional gain, K_p depends on the thermal characteristics of a data center. Moreover, the reactive control sequence includes a user-defined cut-off set point for each cooling device, C after which the controller action ceases to change cooling set point.

Step 3: Populate a database as cooling set points are being regulated. The database will record process variable, manipulated variables such as cooling equipment set points, disturbance variables such as power distribution unit (PDU) power, CRAC power, chiller power, outdoor air temperature, and performance metric such as power usage effectiveness (PUE).

Step 4: Monitor the designed PID controller response, which is expected to be:

- Increase in IT component temperature in a low-cooling environment.
- Depending on cooling equipment design, there could be a transient spike in reheat coil power.
- If the cooling equipment has a variable-frequency fan, the fan should eventually slow down.
- The chiller starts to consume less power in low-cooling environment.

Step 5: When the value of Δ reaches a pre-assigned value $\Delta_1 (\neq 0)$, the controller action for modulating set points of high-power cooling equipment ceases and the low-power cooling equipment modulation begins. This changeover ensures that the process variable always remains below the Threshold. The low-power cooling set point modulation could be based on a PID-like controller (Eq. 2) or a discrete control scheme based on the manufacturer ratings and operators' preferences.

Step 6: When the value of Δ reaches a pre-assigned value $\Delta_2 (< \Delta_1)$, the low-power cooling equipment set-point rolls back to its initial state to provide maximum cooling rapidly and ensure equipment protection against overheating.

[*Comment on choices of* Δ_1, Δ_2: The values of operator-specified constants, Δ_1, Δ_2 dictate cooling cost savings and protection from overheating. Their values should be as low as possible to provide sufficient cooling cost savings, but not too low to risk overheating.]

Machine Learning-based Predictive Algorithm and Control Sequence in the Predictive Mode

The control sequence in the predictive mode reduces SA's dependency on live data feed. An historical database is used to predict optimal cooling set points. A proper orthogonal decomposition (POD)-based machine learning algorithm is used for the real-time analytics in the predictive mode. Table 2 shows the predictive control algorithm, and Table 3 shows the overall predictive control sequence.

Table 2: POD-based predictive control algorithm

1. Filter the database based on the operator-specified criteria.

2. Rearrange the database based on the choices of independent and dependent variables to form a training data matrix.
3. Calculate the parametric average of the training data matrix.
4. Subtract the parametric average from each column of the training data matrix.
5. Apply POD on the residual matrix to compute POD modes and POD coefficients.
6. Recognize the output point and its relationship with the parametric input space.
7. The POD coefficients corresponding to the output point are a part of the column space spanned by the POD coefficients of the residual matrix. Apply suitable statistical techniques such as weighted interpolation/ extrapolation or Kriging to determine the POD coefficients for the output point.
8. Multiply POD modes and POD coefficient for the output point and add the product to the parametric average to get the prediction.

Table 3: Predictive control sequence

1. Measure runtime values of the disturbance variables such as PDU power, chiller power, outdoor air temperature, and the manipulated variables such as cooling set points.
2. Filter the training database based on the runtime condition at the onset of the sequence.
3. Use the machine learning algorithm (Table 2) to predict the trend of process variable.
4. Collect real-time data at a pre-assigned check interval (<< data interval).
5. Determine Deviation: the difference between runtime process variable value and corresponding predictive value.
6. Determine Delta-s: the differences between the Threshold and predictive values of the process variable, generated in Step 3.
7. If Deviation < *D* AND Delta > Δ_1 : Change the high-power cooling set points by (0.5 x Delta) and keep the low-power cooling set points constant.
8. If Deviation < *D* AND Δ_2 < Delta < Δ_1 : Change the low-power equipment to lower-cost set point following manufacturer's guideline and keep the high-power equipment set point constant.
9. If Deviation < *D* AND Delta < Δ_2 : Change the low-power equipment to a set point that provides maximum cooling and keep the high-power equipment set point constant
10. If Deviation >= *D* AND Delta > Δ_1 +1: Step change high-power cooling equipment set point to provide rapid cooling (e.g.: In GT DCL, the high-power CRAC unit temperature set point is reduced by 1 ^{0}F.)
11. If Deviation >= *D* AND Delta < Δ_1 +1: Step change high-power cooling equipment set point to provide emergency cooling (e.g.: In GT DCL, the high-power CRAC unit temperature set point is reduced by 2 ^{0}F.)

Comments:

- The numerical values of *D*, Δ_1 , Δ_2 used in Steps 7-11 are set based on data center cooling and load characteristics.
- Steps 10-11 are designed for handling low-fidelity nature of the predictive control mode.

(e) Dashboard

For GT DCL, AdeptDC developed a dashboard using a Highcharts-based (www.highcharts.com) JavaScript library. The cross-browser compatibility of Highcharts makes it particularly suitable for data visualization. For Arrow data center, AdeptDC developed a Splunk (www.splunk.com) dashboard, complying with the user demand.

(f) Cooling Automation System

A typical data center has a building management system (BMS)-integrated web-based supervisory control and data acquisition (SCADA) system for their cooling infrastructure. The incumbents' intellectual property (IP) rights do not allow a third-party software provider like AdeptDC to connect with the BMS server and programmatically control cooling set points. AdeptDC developed a RESTful API-based web application for controlling cooling set points. For developing this API, the data packet sent by the BMS server to a cooling equipment needs to be intercepted. The key steps for cooling automation framework development for a commercial-grade data center are, as follows:

1. Open cooling network management interface (PrismII in Arrow and Tridium Niagara in GT DCL)
2. Navigate to the control panel for each cooling equipment.
3. Open Wireshark or an equivalent packet analyzer. Listen on 'Local Area Connection' and filter the traffic from and to the BMS server. Find the command packet that is sent from PrismII/ Tridium Niagara to BMS server to get all set point related data and also the packets containing the response data.
4. Update one cooling set point at a time and determine the corresponding packets via Wireshark.

The developed data packets are sent to the BMS server, which in turn orchestrates the optimal set points to corresponding cooling devices. Figure 8 shows the Wireshark-based cooling control sequence and cooling control dashboard deployed AdeptDC at Arrow Data Center.

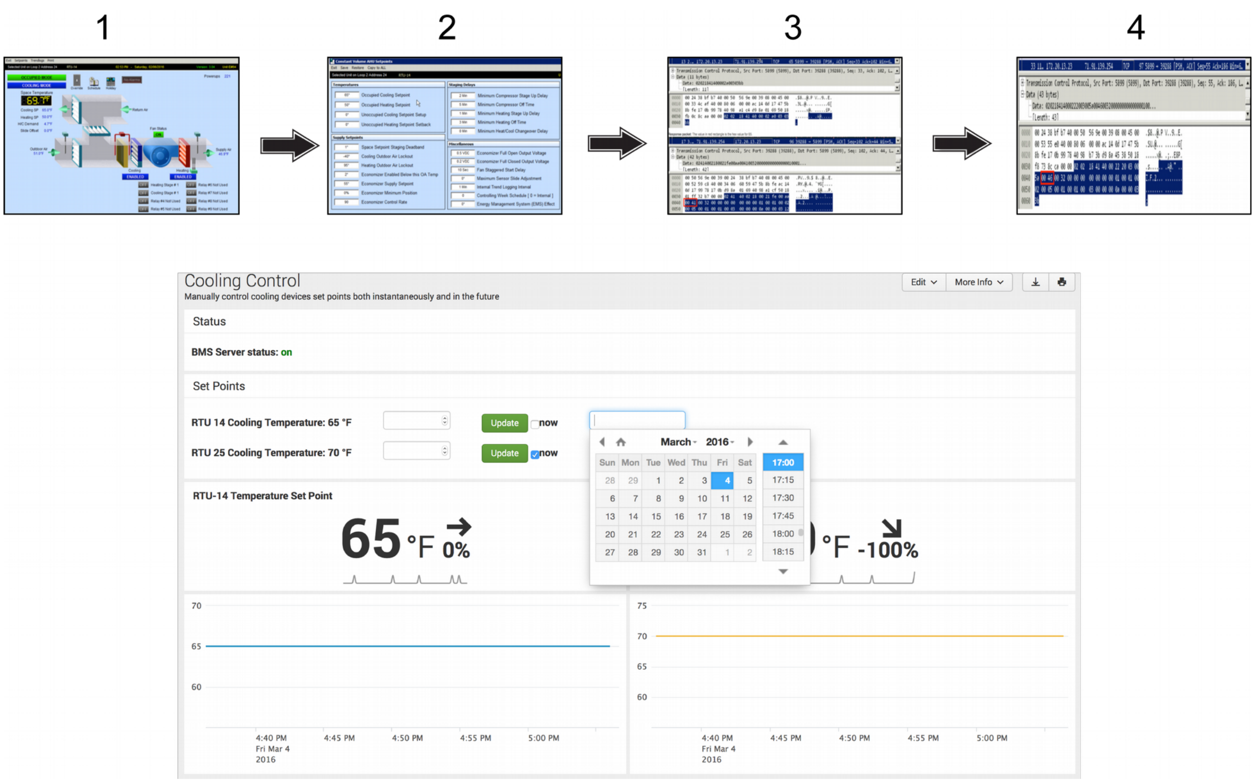


Figure 8: Cooling Control Sequence and Dashboard for Arrow Data Center. The Users Can Change Cooling Set Points Instantly or Schedule a Set Point Change in the Future.

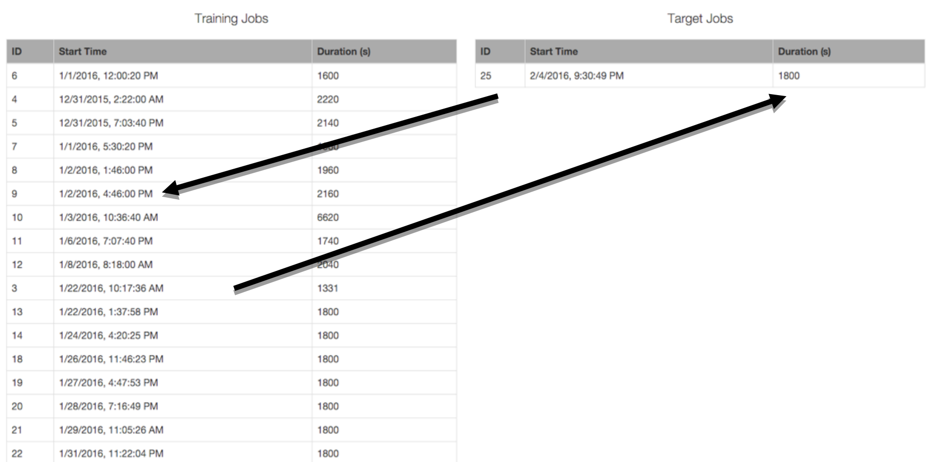
2. Component Testing

(a) Testing

Three major functional components of AdeptDC's SA—IT temperature data collection application, real-time analytics-based control module, and cooling automation application, as shown in Figure 1—have been evaluated using Markov analysis-based cleanroom software development methodology. Special care has been taken to protect the appliance from web vulnerabilities such as SQL injection.

(b) Test Result Database Compilation and Adjustment

The results from various test cases are compiled in a database. Several updates have been made based on the test results. The most notable update is related to easy shuffling of training and target data in the predictive control algorithm interface (Figure 9). It eases the fidelity testing for the data-driven predictive control algorithm (Table 2) with different combinations of training and target data. Figure 9 shows the interface of the sortable table for the POD-based predictive algorithm in GT DCL. The control algorithm is tested for the different combinations of training and target jobs. Table 4 shows the root mean square (RMS) error for the different combinations. As evident from Table 4, the maximum RMS value of the error (defined as the difference between the run-time and predictive values) is equal to 0.79 ^{0}C, which is less than 5% of the minimum motherboard temperature (~ 20 ^{0}C). This establishes the fidelity of the predictive control algorithm. Two parameters used for the fidelity testing are IT power and outdoor air temperature. While IT power rapidly fluctuates in a data center and influences cooling load, the outdoor air temperature change impacts the electricity consumed by cooling devices to maintain data center temperature and humidity within acceptable limits.



Training Jobs

ID	Start Time	Duration (s)
6	1/1/2016, 12:00:20 PM	1600
4	12/31/2015, 2:22:00 AM	2220
5	12/31/2015, 7:03:40 PM	2140
7	1/1/2016, 5:30:20 PM	[illegible]
8	1/2/2016, 1:46:00 PM	1960
9	1/2/2016, 4:46:00 PM	2160
10	1/3/2016, 10:36:40 AM	6620
11	1/6/2016, 7:07:40 PM	1740
12	1/8/2016, 8:18:00 AM	2040
3	1/22/2016, 10:17:36 AM	1331
13	1/22/2016, 1:37:58 PM	1800
14	1/24/2016, 4:20:25 PM	1800
18	1/26/2016, 11:46:23 PM	1800
19	1/27/2016, 4:47:53 PM	1800
20	1/28/2016, 7:16:49 PM	1800
21	1/29/2016, 11:05:26 AM	1800
22	1/31/2016, 11:22:04 PM	1800

Target Jobs

ID	Start Time	Duration (s)
25	2/4/2016, 9:30:49 PM	1800

Figure 9: Sortable Table for Rapid Shuffling between Training and Target Jobs

Table 4: Fidelity of POD-based Predictive Control Algorithm. The process variable was the maximum motherboard temperature.

Target Database ID	Start Time	End Time	RMS Error (^{0}C)	Heat Load Average (kW)	Heat Load STDEV (kW)	Outside Air Temperature Average (^{0}F)	Outside Air Temperature STDEV (^{0}F)
1	12/31/2015, 2:22:00 AM	12/31/2015, 2:59:00 AM	0.57	81.72	0.48	57.10	0.04
2	12/31/2015, 7:03:40 PM	12/31/2015, 7:39:20 PM	0.52	80.62	0.41	52.38	0.00
3	1/1/2016, 12:00:20 PM	1/1/2016, 12:27:00 PM	0.79	83.98	1.19	44.19	0.00
4	1/1/2016, 5:30:20 PM	1/1/2016, 6:03:00 PM	0.63	81.41	0.65	44.24	0.47
5	1/2/2016, 1:46:00 PM	1/2/2016, 2:18:40 PM	0.76	75.24	1.23	44.68	1.51
6	1/2/2016, 4:46:00 PM	1/2/2016, 5:22:00 PM	0.57	71.86	0.72	47.02	0.05
7	1/3/2016, 10:36:40 AM	1/3/2016, 12:27:00 PM	0.43	72.05	3.80	40.73	2.60
8	1/6/2016, 7:07:40 PM	1/6/2016, 7:36:40 PM	0.4	71.82	0.81	42.12	0.82
9	1/8/2016, 8:18:00 AM	1/8/2016, 8:52:00 AM	0.59	71.77	0.65	47.61	0.00

3. Integration

(a) Developing Bridging Protocols

Three major pieces—IT temperature data collection application, real-time analytics module, and cooling automation application—are connected via a node.js based software framework, developed in-house.

4.System-level Testing

(a)Laboratory-scale Deployment

The SA is deployed in GT DCL which is a 600 sq. ft. facility with 100 kW design IT load, 14,000 CFM rack flow rate, 7x2 alternating cold/hot aisle architecture, 6,000 CFM tile flow rate, under-floor plenum supply and overhead plenum return air-flow design. It has FH 740 Liebert down-flow CRAC unit connected to 140 ton Trans R series air cooled chiller and 18 kW Coolcentric rack-mounted RDHx-s. There are 32 HP ProLiant servers under the purview of the deployed SA. Figure 10 shows the results of reactive control sequence implemented at GT DCL. For the reactive control sequence, the process variable is the maximum motherboard temperature. While the manipulated variables are the set points of cooling equipment, the disturbance variables are PDU power, chiller power, CRAC power, and outdoor air temperature. The measurement time window (*T*) was 2000 s and sampling period was 20 s. For this particular sequence, the operators set following parameters:

$$T_{Threshold} = 35\ {}^0C,\ k_p = 0.5,\ C = 75\ {}^0F, \Delta_1 = 3\ {}^0C,\ \Delta_2 = 2\ {}^0C.$$

[The temperature units are based on the convention: ^{0}C for IT temperatures, ^{0}F for facilities temperatures.]

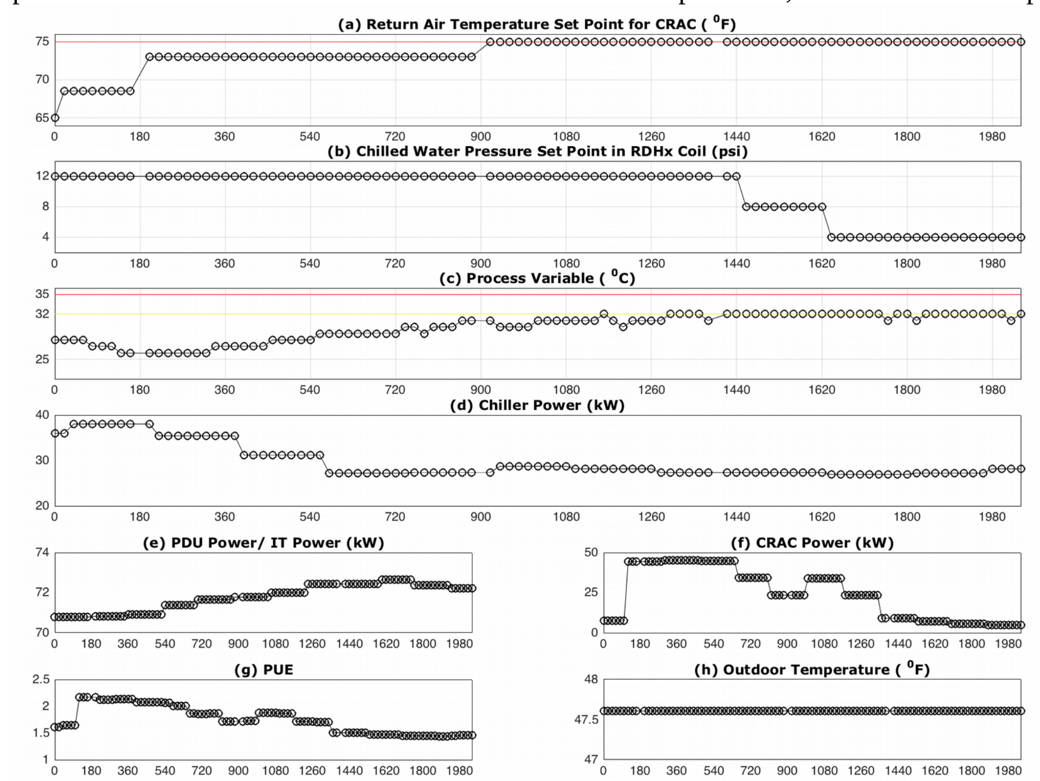


Figure 10: AdeptDC's SA-enabled reactive control sequence at GT DCL. The X-axis shows time in seconds. This controlled sequence started at 8:18:00 AM on 1/8/2016.

Figure 10(a) shows the first manipulated variable: CRAC return air temperature set point. At t=0, the process variable (Figure 10(c)) was equal to 28 ^{0}C or $\Delta = 7\ {}^0C$, so the PID controller increases CRAC set point by 3.5 ^{0}F to 68.5 ^{0}F from 65 ^{0}F. Because of the slow nature of the data center thermal processes, each cooling set point modulation is followed by a 180 s settling time. That explains the next rise in CRAC temperature set point, from 68.5 ^{0}F to 73 ^{0}F, corresponding to $\Delta = 9\ {}^0C$ at 180 s. From 0-180 s, the process variable dipped from 28 ^{0}C to 26 ^{0}C even though the CRAC temperature has increased at t=0. This counter-intuitive observation indicates IT and cooling disconnection. The CRAC return air temperature set point remained constant for next three control triggers at 360 s, 540 s, and 720 s because the CRAC temperature set point can not cross the operator-specified cut-off, C = 75 ^{0}F. However, as soon as the critical temperature reaches 31 ^{0}C or $\Delta = 4\ {}^0C$ at 900 s, the PID controller action causes the CRAC set point to rise by 2 ^{0}F to 75 ^{0}F. Eventually, the process variable rises further to 32 ^{0}C or $\Delta = 3\ {}^0C$, and the controller action switches over to the modulation of chilled water pressure set point of RDHx (low-power cooling unit). For the RDHx, the operator prefers pressure set point modulation in three discrete stages, [12, 8, 4] psi. As the process variable hovers over 32 ^{0}C, the RDHx pressure set point reduces from 12 psi to 8 psi at 1440 s, and then to 4 psi at 1620 s (Figure 10(b)). The major driver of the cooling cost savings is the lower chiller operating power. As shown in Figure 10(d), the chiller operating power reduced by 9 kW, offering nearly 25% cooling cost savings. The air-cooled chiller cost-savings value depends on the outdoor air temperature, 47.6 ^{0}F in this case (Figure 10(h)). [Higher outdoor air temperature results higher power savings]. Additionally, the SA monitored PDU power (Figure 10(e)), CRAC power (Figure 10(f)), and PUE (Figure 10(g)). The change in PDU power was completely dependent on the data center IT operations. The CRAC power increased initially due to reheat coil action. The energy efficiency performance of the SA was quantified by PUE—ratio of total power to IT (PDU) power. The lower the value of PUE, the higher the cooling energy efficiency. Figure 10(g) shows PUE, decreasing from 1.61 to 1.45.

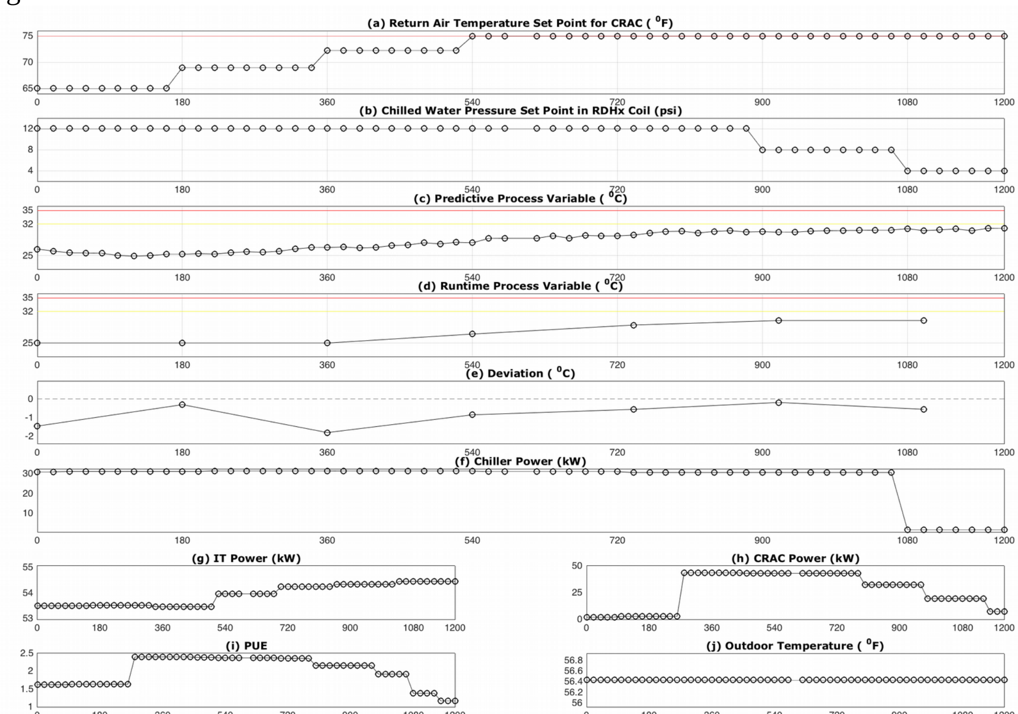


Figure 11: AdeptDC's SA-enabled predictive control sequence at GT DCL. The X-axis shows time (in second) from 0-1200 s. This sequence started at 10:53 AM on 2/24/2016.

Figure 11 shows the results from predictive control sequence at GT DCL at 10:53 AM on 2/24/2016. To feed predictive control sequence, a database is populated with process variable, manipulated variables, and disturbance variables from reactive control sequences. A filtered part of the database is used as the training data for the predictive control algorithm (Table 2). Figure 11 shows the results of the predictive control sequence (Table 3). While the measurement time window was equal to 1200s at 20s data sampling interval, the check interval for the fidelity assessment was set to 180 s. The remaining parameters were, as follows:

$$T_{Threshold}=35\,℃, k_p=0.5, C=75\,℉, \Delta_1=3\,℃, \Delta_2=2\,℃, D=1\,℃.$$

As shown in Figure 11(d), the runtime process variable value was collected at every 180s, instead of 20s. That amounts to data compression of 88.9% (=8/9). The prediction algorithm forecasted the process variable, as shown in Figure 11(c). The deviations between the runtime data and the corresponding predictions were noted (Figure 11(e)) at an 180 s interval (check interval). The deviation values were always found to be lying below +1 ^{0}C, hence the error-based feedback control procedure was not invoked. The respective dynamic profiles of CRAC temperature set point and RDHx pressure set point are shown in Figure 11(a) and Figure 11(b). The chiller power savings is shown in Figure 11(f). Other disturbance variables shown are IT power (Figure 11(g)), CRAC power (Figure 11(h)), and outdoor air temperature (Figure 11(j)). Figure 11(i) shows performance metric, PUE which decreased from 1.55 to 1.15.

(b)Testing

The SA was exhaustively tested for its functionality and reliability. The test metric for functionality is cooling energy savings and for reliability is IT component overheating.

(c)Test Result Compilation and Refinement

A database was generated with the records of change in PUE, chiller power reduction, and maximum value of the process variable. Additionally, CPU dip percentage—the percentage of CPU population for which frequency has gone down—was noted. The CPU dip percentage greater than an operator-specified safety limit (20% for GT DCL) indicates IT performance issues from component overheating.

For GT DCL, more than 50 independent cooling control sequences were observed at different computing conditions and outdoor air temperatures. It was observed the the SA reduces PUE by 0.2 (min 0.1; max 0.45), on average. The PUE reduction comes from the decrease in the chiller power, which reduced by 20% (min 8%; max 35%), on average.

AdeptDC essentially reduces data center cooling cost by increasing its operating temperature. Therefore, some overheating problem was expected. During the Phase I, there were two major overheating incidents, both on November 18, 2015 at GT DCL. Figure 12 shows the CRAC temperature and CPU frequency dip percentage trends in that time-frame.

- The first instance was started around 9:50 AM with the maximum CPU temperature as the process variable. It caused rapid escalation of CRAC set point to an extent that the CRAC temperature set point jumped from 65 ^{0}F to 92 ^{0}F within 15 min. This rapid escalation did not cause any IT downtime because the CRAC set point was immediately retracted back to 70 ^{0}F. The temperature escalation was caused by the wrong choice of the process variable; the dominant heat-dissipating components for the HP ProLiant servers were GPUs, not CPUs. For the GPU-dominated HP ProLiant servers, there was no CPU temperature rise proportional to CRAC set point increase.
- The second instance was started around 2:22 PM with the maximum motherboard temperature as the process variable. It caused a sudden spike in the CRAC set point from 70 ^{0}F to 88.5 ^{0}F. This overheating caused multiple server shutdowns with the CPU frequency dip percentage violating the 20% limit. The spike was caused by the flawed control logic implemented by AdeptDC: *if the process variable is somehow higher than the Threshold, the controlled cooling set point would roll back to the last set point to provide rapid cooling*. At 2:22 PM, when the maximum motherboard temperature was 32 ^{0}C (> Threshold, set at 30 ^{0}C), the CRAC set point rolled back to 88.5 ^{0}F, the set point at the end of the previous control sequence.

After these overheating events, several changes were in made in the software system:

- Removal of the cooling set point roll-back logic
- Real-time power dissipation monitoring for different chip components to determine the appropriate process variable (the component with highest and most dynamic temperature should be the process variable).
- Introduction user-drive fail-safe or cut-off condition. The cut-off ensures cooling set point always remains below a critical value set by the operator.

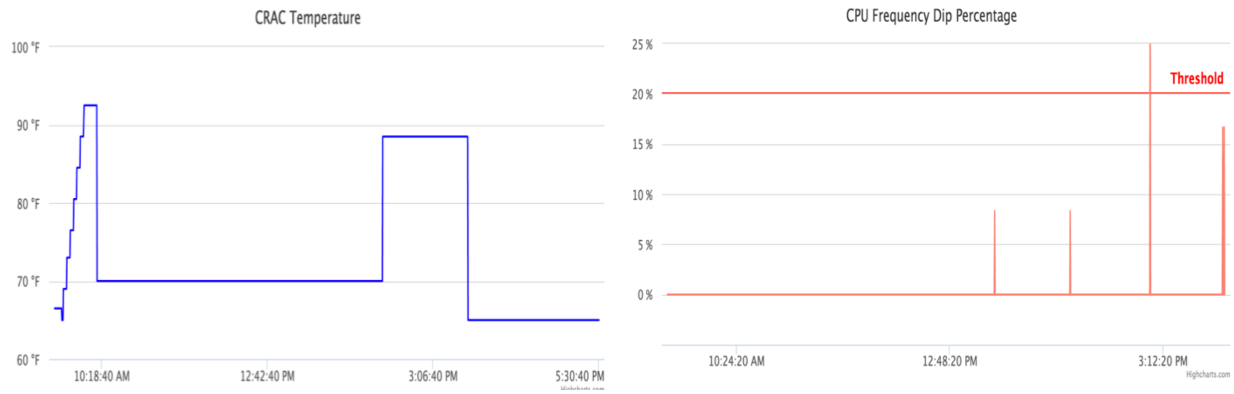


Figure 12: Two IT Service Performance Issues Caused by AdeptDC's SA in GT DCL

5.Performance Benchmarking

(a)Live Deployment

After a thorough offline testing regime, the SA was deployed in GT DCL and Arrow Data Center for live implementation. The live implementations include strict monitoring features and conservation fallback criteria to avoid inadvertent overheating issues. The SA deployed in GT DCL was discussed in detail in reference to Figure 10 and Figure 11.

Complying with the beta customer specification, a Splunk app was developed for implementing reactive cooling controls in Arrow Data Center. Arrow Data Center is a 3,000 sq. ft. facility, with 100 kW operating IT load and heterogeneous IT equipment pool from vendors like NetAPP, HP, Lenovo, IBM, Brocade, Cisco, EMC. Arrow data center has one 7 Ton rooftop unit (RTU-14), one 10 Ton rooftop unit (RTU-25), one 12 Ton Schultz CRAC unit, and one 160 Ton water chiller connected to rack mounted RDHx units. Figure 13 shows the Splunk dashboard for controlling RTU-14 and RTU-25 cooling set points. It includes user input boxes for choosing process variable (either motherboard or CPU temperature), safety limit for the process variable, cut-offs for RTU-14 and RTU-25, drop-down menu for choosing manual (as shown in Figure 8) or automatic mode (based on the AI technology), and an emergency "Stop" button.

The dashboard in Figure 13 shows an automated cooling control sequence with CPU temperature as the process variable, 55 ^{0}C as the safety limit, 80 ^{0}F as the cut-off for RTU-14, and 75 ^{0}F as the cut-off for RTU-25. Unlike GT DCL, both cooling devices in Arrow Data Center are of comparable cooling capacities. For this control sequence, both of them were treated as high-power cooling devices with different cut-offs. Other pertinent reactive control sequence parameters were: k_p=0.5 and Δ_1= 3 ^{0}C. During the reactive control sequence on February 26, 2016 (Figure 13), the RTU set points were increased until they reached their cut-offs as the process variable decreased marginally. Surprisingly, there was a decrease in CPU temperature which could be attributed to corresponding changes in IT operations and low cooling capacity of the RTU units. The aggregate cooling capacity of two RTU units are only 9% of the total cooling capacity at Arrow. In future, AdeptDC will onboard their high-power cooling units to generate appreciable cooling cost savings.

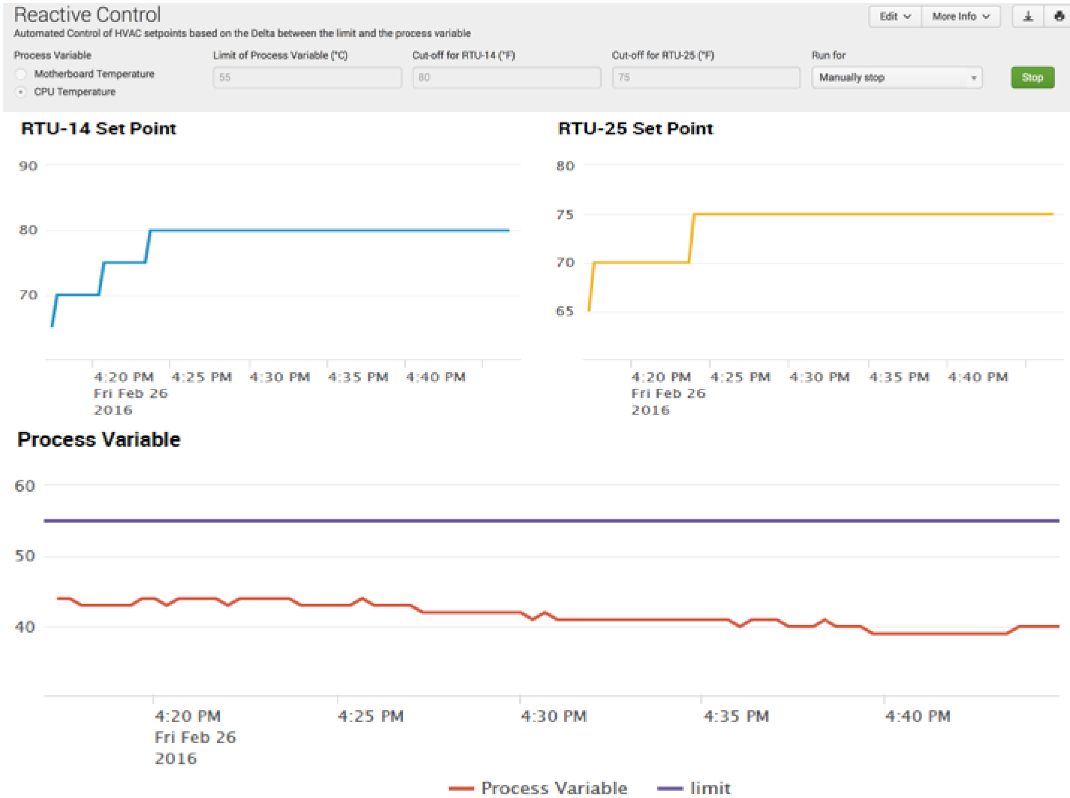


Figure 13: Splunk Dashboard for Reactive Cooling Controls in Arrow Data Center. RTU-14 and RTU-25 set points are in ^{0}F and Process Variable is in ^{0}C.

(b)Continuous Performance Monitoring

The key performance metrics used were cooling electricity savings, cooling capacity savings, coordinating between different cooling devices for overall cooling energy optimization, and maintenance of process variable below a safety limit to ensure no IT performance degradation from overheating. These metrics were monitored using a web-based user interface and a SQL database.

(c)Evaluation and Refinement

The performance metrics were evaluated using the records in the SQL database which notes the dynamic profiles of CRAC set points, RDHx set points, chiller power, CRAC power, PDU power, and UPS power. Based on the CRAC manufacturer data, 1 ^{0}C increase in CRAC temperature set point reduces its operating electricity cost by 2% and improves its capacity utilization by 3%. With the outdoor-air temperature more than 60 ^{0}F in November-December at Atlanta, the chiller cost savings in GT DCL was between 25-35%. In January-February as the outdoor air temperature dropped below 50 ^{0}F, the chiller savings dipped down to 20-25% for GT DCL. Without AdeptDC's SA, the average electricity cost for the chiller was around $10,000/ month. For Arrow, there was no appreciable cooling cost savings because AdeptDC is still working on onboarding the high-power cooling units. However, AdeptDC's software assisted them identify the key thermal vulnerability points in Arrow data centers. This benefits prompted Arrow issue a purchase order for AdeptDC's software solution.

Problems Encountered and Methods of Resolution

During SBIR Phase I project, three major problems faced by AdeptDC were, as follows:

Access to Commercial Data Centers: The SA was initially conceptualized as an entirely data-driven predictive tool. However, it takes a prohibitively long time—at least a few weeks—to populate the database for enabling the predictive model. The mission-critical nature of data center operations prohibits such a long deployment time. This discovery led AdeptDC to changing the predictive solution strategy to a bi-modal AI solution strategy. At the onset of its deployment, the AI solution operates in a learning mode, in which it controls cooling via a PID-type real-time controller and compiles training database for predictive computation. Once a sufficiently large training database is compiled, the AI solution starts to operate in the predictive mode.

IT Data Collection Time: The communication protocols for collecting chip level thermal data vary depending on IT equipment types, vendors, management protocols, and network architectures. This non-standardized nature of communication protocols leads to manual connection of IT equipment to the data aggregation module and consequently a long development time for IT data collection module. Currently, AdeptDC is building a library of communication protocols for connecting to different IT equipment. A sufficiently large library would pave the way for an automated IT equipment connection application.

Cooling Set Point Controls: The development of the automatic cooling control module was difficult because of the incumbents' proprietary technologies. AdeptDC resolved this issue by developing a web-based network control tool that manages network packets and modulates cooling set points independently—without any support from third party BMS solution providers.

Unfilled Research Objectives

All major research objectives were met. The only room for improvement lies in live deployment in Arrow Data Center. During the project time-frame, their high-power chiller unit could not be put onboard. Therefore, a thorough benefit analysis could not be performed. This, however, is due to the BMS and networking issues in Arrow Data Center which are expected to be resolved in next few weeks and were completely beyond AdeptDC's control.

Conclusions

This SBIR Phase I project validates AdeptDC's hypothesis that data center cooling cost can be significantly reduced with a cooling control mechanism based on chip-level thermal data. A prototype software appliance (SA) was developed and tested in two data centers. Although the project demonstrated cooling cost savings benefits of the prototype SA, it also uncovered its limitations:

1. There were two overheating incidents due to poor overheating management features.
2. The transition between reactive to predictive modes is managed manually.
3. The Wireshark-based cooling control involves significantly long deployment time based on data center network complexity.
4. The usage of IPMI commands for chip temperature collection raises security concerns.
5. The varied protocols used for collecting chip thermal data causes long deployment time.

These limitations set the stage for future research and development activities, possibly a SBIR Phase II project. The focus of the SBIR Phase II will be to address these limitations and transform the prototype SA into a production-grade cooling optimization software appliance for commercial data centers.