



(51) International Patent Classification:

F24F 11/63 (2018.01) *F28F 27/00* (2006.01)
F24F 11/70 (2018.01) *G05B 17/02* (2006.01)
F24F 11/83 (2018.01) *G05D 23/19* (2006.01)

(21) International Application Number:

PCT/CA2018/051640

(22) International Filing Date:

20 December 2018 (20.12.2018)

(25) Filing Language:

English

(26) Publication Language:

English

(30) Priority Data:

62/609,545 22 December 2017 (22.12.2017) US

(71) Applicant: **MCMASTER UNIVERSITY** [CA/CA]; 175 Longwood Road South, Suite 305, Hamilton, Ontario L8P 0A1 (CA).

(72) Inventors: **GARCIA, Fernando Martínez**; 19 St. Clair Avenue, Hamilton, Ontario L8M 2N4 (CA). **BADAWY, Ghada**; 4305 Taywood Drive, Burlington, Ontario L7M 4S7 (CA). **DOWN, Douglas Graham**; 60 South Oval, Hamilton, Ontario L8S 1P8 (CA).

(74) Agent: **ROSS, Alex**; Gowling WLG (Canada) LLP, 1 Main Street West, Hamilton, Ontario L8P 4Z5 (CA).

(81) Designated States (unless otherwise indicated, for every kind of national protection available): AE, AG, AL, AM, AO, AT, AU, AZ, BA, BB, BG, BH, BN, BR, BW, BY, BZ, CA, CH, CL, CN, CO, CR, CU, CZ, DE, DJ, DK, DM, DO, DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, GT, HN, HR, HU, ID, IL, IN, IR, IS, JO, JP, KE, KG, KH, KN, KP, KR, KW, KZ, LA, LC, LK, LR, LS, LU, LY, MA, MD, ME, MG, MK, MN, MW, MX, MY, MZ, NA, NG, NI, NO, NZ, OM, PA, PE, PG, PH, PL, PT, QA, RO, RS, RU, RW, SA, SC, SD, SE, SG, SK, SL, SM, ST, SV, SY, TH, TJ, TM, TN, TR, TT, TZ, UA, UG, US, UZ, VC, VN, ZA, ZM, ZW.

(84) Designated States (unless otherwise indicated, for every kind of regional protection available): ARIPO (BW, GH, GM, KE, LR, LS, MW, MZ, NA, RW, SD, SL, ST, SZ, TZ, UG, ZM, ZW), Eurasian (AM, AZ, BY, KG, KZ, RU, TJ, TM), European (AL, AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI, FR, GB, GR, HR, HU, IE, IS, IT, LT, LU, LV, MC, MK, MT, NL, NO, PL, PT, RO, RS, SE, SI, SK, SM, TR), OAPI (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, KM, ML, MR, NE, SN, TD, TG).

Published:

— with international search report (Art. 21(3))

(54) Title: CONTROLLING OPERATIONAL PARAMETERS OF A COOLING UNIT

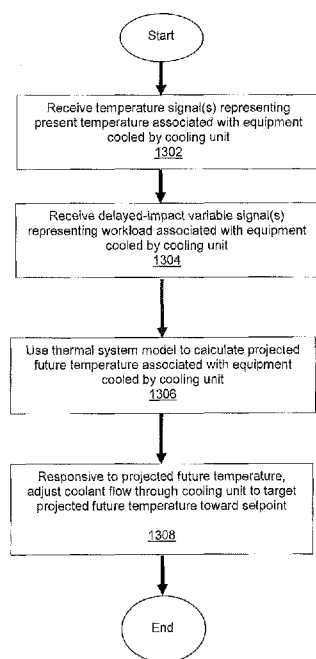


FIG. 13

(57) Abstract: A cooling unit having a liquid coolant flow through the cooling unit and a gas coolant flow through the cooling unit is controlled using a temperature signal representing a present temperature associated with equipment cooled by the cooling unit and a delayed-effect variable signal representing a variable having a potentially delayed impact on the temperature associated with the equipment cooled by the cooling unit. A thermal system model to which the temperature signal, the delayed-effect variable signal, the liquid coolant flow and the gas coolant flow are inputs is used to calculate a projected future temperature associated with the equipment. Responsive to the projected future temperature, coolant flow is adjusted to target the projected future temperature toward a set-point by adjusting the liquid coolant flow and the gas coolant flow. The delayed-effect variable signal may represent, for example, a processing workload or a current drawn by the equipment.

CONTROLLING OPERATIONAL PARAMETERS OF A COOLING UNIT

CROSS-REFERENCE TO RELATED APPLICATION

[0001] The present application claims priority to United States Provisional Application No. 62/609,545 filed December 22, 2017, the teachings of which are hereby incorporated by reference.

TECHNICAL FIELD

[0002] The present disclosure relates to cooling units for cooling of equipment.

BACKGROUND

[0003] Data center (DC) energy consumption has attracted a lot of attention in recent years. According to J. Koomey, "Growth in data center electricity use 2005 to 2010," *A report by Analytical Press, completed at the request of The New York Times*, 2011, DC energy consumption ranges from 1.1% to 1.5% of total global electricity consumption, which also has a tendency to increase; see J. Whitney and P. Delforge, "Scaling up energy efficiency across the data center industry: Evaluating key drivers and barriers," NRDC, Issue Paper No. IP 08-14-A, Aug, 2014, pp. 1–35. A significant portion of this energy utilization is devoted to cooling systems that aim to keep server temperatures within a safe region, necessary to avoid damage to servers. The safe temperature ranges for different classes of equipment are provided by guidelines such as those of ASHRAE, "ASHRAE TC9.9", from Data Center Power Equipment Thermal, 2011, p. 11. Traditionally in DCs, cooling infrastructure is either room-based or row-based; see M. K. Patterson, "The Effect of Data Center Temperature on Energy Efficiency," in *2008 11th Intersociety Conference on Thermal and Thermomechanical Phenomena in Electronic*, Orlando, 2008 and K. Dunlap and N. Rasmussen, "Choosing Between Room, Row, and Rack-based Cooling for Data Centers," in *Schneider Electr. White Paper*, 2012 ("K. Dunlap et al.").

[0004] However, in recent years rack mountable cooling units have been introduced to cope with the increasing demand for high performance computations (HPC). These new cooling units bring servers and cooling units closer to each other with an aim to decrease cooling infrastructure energy consumption; see K. Dunlap et al.

[0005] In addition to reducing energy consumption, maintaining a stable temperature inside a data center is one of the most important requirements since constant oscillations in cold air temperature, even by 1 or 2 degrees, increases the probability of server failures; see N. El-Sayed, I. Stefanovici, G. Amvrosiadis and H. Andy A., "Temperature Management in Data Centers: Why Some (Might) Like It Hot," in *2012 ACM SIGMETRICS / International Conference on Measurement and Modeling of Computer Systems*, London, 2012. These oscillations are an inherent characteristic of the ON/OFF or proportional integral derivative (PID) controllers which have been widely used in cooling infrastructure; see A. Afram and F. Janabi-Sharifi, "Theory and applications of HVAC control systems A review of model predictive control (MPC)," *Building and Environment*, vol. 72, pp. 343-355, 2014. Due to the proximity of the rack mounted cooling units to the servers, any variation in airflow created by these controllers, as a response to changes in workload, will be experienced immediately by the servers, which will consequently lead to higher server failure rates.

[0006] Model Predictive Control (MPC) could avoid these oscillations by estimating the behavior of the system using a fixed model approach. MPC implementation in DC has already been explored (see Q. Fang, J. Wang, and Q. Gong, "QoS-Driven Power Management of Data Centers via Model Predictive Control," *IEEE Transactions on Automation Science and Engineering*, vol. 13, pp. 1557-1566, 2016) showing good results via simulations. Although MPC can handle a wide variety of problems (see M. Kheradmandi, P. Mhaskar, "Model predictive control with closed-loop re-identification," *Computers & Chemical Engineering*, 109:249-60, 2018), it usually requires the solution of an iterative optimization problem at all sample times, which makes it infeasible in real world deployments since it requires expensive advanced micro-controllers that can handle this computational effort in real-time. Moreover, mismatches due to fixed models in MPC could have negative effects on the performance of the system.

SUMMARY

[0007] In one aspect, the present disclosure is directed to a method for controlling operational parameters of a cooling unit, wherein a coolant flow through the cooling unit comprises a liquid coolant flow through the cooling unit and a gas coolant flow through the cooling unit.

The method comprises receiving, at a controller for the cooling unit, at least one temperature signal representing a present temperature associated with equipment cooled by the cooling unit and receiving, at the controller, at least one delayed-effect variable signal representing a delayed-effect variable associated with equipment cooled by the cooling unit. The method further comprises calculating, by the controller, using a thermal system model to which the temperature signal(s), the delayed-effect variable signal(s), the liquid coolant flow and the gas coolant flow are inputs, a projected future temperature associated with the equipment cooled by the cooling unit, and responsive to the projected future temperature, adjusting coolant flow through the cooling unit to target the projected future temperature toward a set-point.

[0008] The thermal system model may include a cooling control function model for selectively cooperating the liquid coolant flow and the gas coolant flow to target the set-point according to an optimization parameter. The optimization parameter may be one of energy consumption and speed of set-point targeting. The thermal system model may be, for example, a static model or an adaptive model.

[0009] In some embodiments, the equipment cooled is information technology equipment, and the delayed-effect variable signal(s) comprise a processing workload signal representing an indication of a workload that will be assigned to the information technology equipment.

[0010] In some embodiments, the delayed-effect variable signal(s) comprise a current signal representing a current drawn by the equipment cooled by the cooling unit.

BRIEF DESCRIPTION OF THE DRAWINGS

[0011] These and other features will become more apparent from the following description in which reference is made to the appended drawings wherein:

Figure 1 shows, in schematic form, an illustrative test bed for an illustrative implementation of cooling according to an aspect the present disclosure;

Figure 2 shows a two-dimensional representation of the possible infinite set of solutions that arise in the proposed approach when $\mathbf{u}^i \in \mathcal{U}$ (left side of Figure 2) and $\mathbf{u}^i \notin \mathcal{U}$ (right side of Figure 2);

Figure 3 is a graph showing performance of an APC with standard GPC and an APC according to an aspect of the present disclosure when the random sinusoidal amplitude parameters of the water temperature are set to $\mu = 0.4^{\circ}\text{C}$ and $\sigma = 0.3^{\circ}\text{C}$;

Figure 4 is a graph showing performance of an APC with standard GPC and an APC according to an aspect of the present disclosure when the random sinusoidal amplitude parameters of the water temperature are set to $\mu = 0.145^{\circ}\text{C}$ and $\sigma = 0.11^{\circ}\text{C}$;

Figure 5 shows a block diagram representation of an illustrative control arrangement implemented on an illustrative system;

Figure 6 is a graph comparing performance results for APC according to an aspect of the present disclosure to those for PID from an experiment using the test bed of Figure 1 in which the top 12 servers were turned on, the bottom servers turned off and the air ducts of the latter blocked;

Figure 7 is a graph comparing PWM manipulation of an APC according to an aspect of the present disclosure to that for PID, with resolution of 1, within the range [35,255] (8 bit representation);

Figure 8 is a graph comparing water flow manipulation of an APC according to an aspect of the present disclosure and PID, with resolution of 0.02, within the range [9,21];

Figure 9 shows performance of APC according to an aspect of the present disclosure with monetary cost reduction APCs;

Figure 10 shows cumulative monetary reduction through time as savings in percentage of APC with $C_s(\mathbf{u}(k))$ enabled with respect to APC;

Figure 11 shows an illustrative implementation of cooling according to an aspect of the present disclosure similar to that in Figure 1 but with a current signal as an additional input to the thermal system model;

Figure 12 shows an illustrative implementation of cooling according to an aspect the present disclosure similar to that in Figure 1 but with a processing workload signal as an additional input to the thermal system model; and

Figure 13 is a flow chart showing an illustrative method for controlling operational parameters of a cooling unit.

DETAILED DESCRIPTION

[0012] The present disclosure describes a computationally inexpensive Adaptive Predictive Controller (APC) which is implementable on an over the shelf general purpose microcontroller. The proposed APC may incorporate a projected gradient-based algorithm, so that, unlike certain other low complexity controllers, the proposed APC can reduce power consumption and operating costs.

[0013] The proposed APC proactively acts to predict the behavior of the system and also adapts to new operating conditions by employing a dynamic model. The use of APC has been explored in a variety of applications through simulations (see R. Hedjar, "Adaptive Neural Network Model Predictive Control," *International Journal of Innovative*, vol. 9, no. 3, pp. 1245 - 1257, 2013; V. Bobal, M. Kubalcik, P. Dostal and J. Matejcek, "Adaptive predictive control of time-delay systems," *Computers and Mathematics with Applications*, vol. 66, p. 165–176, 2013; I. Mizumotoa, Y. Fujimoto and M. Ikejiri, "Adaptive output predictor based adaptive predictive control with ASPR constraint," *Automatica*, vol. 57, pp. 152-163, 2015); T.-W. Yoon, & D. W. Clarke "Adaptive predictive control of the benchmark plant," *Automatica*, vol. 30, pp. 621–628, 1994).

[0014] Nonetheless, the real time application of APC in DCs has not yet been addressed. The proposed APC controller is partially adapted from an APC formulation for Multi-Input Single-Output Systems (MISO), for a specific type of minimization cost function. This formulation is combined with a proposed variable prediction horizon algorithm to increase its ability to deal with time-varying systems. Additionally, the APC described herein adds a method to transform infeasible solutions computed by the predictive formulation into feasible solutions when there is at least one. Moreover, predictive constraints based on general

knowledge of the system are added to cope with transitory adaptation to the system. A variable forgetting factor algorithm for the adaptive method is described, so that the learning rate associated with new data varies according to the dynamics of the system. This is combined with an algorithm to avoid numerical instability when the adaptive algorithm faces low system excitation.

Illustrative Hardware System Description

[0015] Broadly speaking, the present disclosure is directed to the cooling of equipment by way of one or more cooling units in which the cooling unit includes a heat exchanger, at least one liquid flow control element adapted to control flow of liquid coolant through the heat exchanger, and at least one gas flow control element adapted to control flow of gas coolant across the heat exchanger. Figure 1 shows, in schematic form, an illustrative system, indicated generally at reference 10, used as a test bed for the present disclosure. The system 10 consists of a single rack 12 containing twenty (20) servers 14 with an average maximum power consumption of 250W per server, and a rack-mounted cooling unit 16 located at the top of the rack 12. The illustrative cooling unit comprises a heat exchanger 18, for example a plate-fin heat exchanger, and a set of five identical compact industrial fans 20; this is merely one illustrative example and is not intended to be limiting. Liquid coolant flows into the heat exchanger 18 by way of an inlet pipe 22 and flows out of the heat exchanger 18 by way of an outlet pipe 24. Thus, a coolant flow through the cooling unit 16 comprises a liquid coolant flow through the cooling unit 16, particularly the heat exchanger 18, by way of the inlet pipe 22 and the outlet pipe 24, and a gas coolant flow through the cooling unit by way of the fans 20. In the illustrated embodiment, the liquid coolant is water and the gas coolant is ambient air; in other embodiments other suitable types of liquid coolant and gas coolant may also be used. Construction of a suitable cooling unit is within the capability of one skilled in the art and is not described further.

[0016] The controlled variable is a temperature associated with the system or equipment to be cooled. Thus, at least one temperature sensor adapted to sense a temperature associated with equipment to be cooled by the cooling unit is provided. In the illustrated embodiment temperature is measured using a temperature sensor 26 of 0.06°C resolution and located in

front of the 12th server 14 from the cooling unit 16, which is the hottest relevant point in front of the rack 12 in the illustrated embodiment. In other embodiments, other locations and/or resolutions for the temperature sensor may be employed. The temperature sensor 26 monitors the temperature of the air in the rack 12, which is presently the preferred embodiment, although other temperatures may be measured (e.g. CPU temperature, chassis temperature, or other temperature, with appropriate model adjustment). Optionally, more than one temperature sensor may be provided, and these may provide a plurality of temperature signals. The manipulated variables are the water flow in the heat exchanger 18 and the pulse-width modulation (PWM) signals that drive rotation of the fans 20, which are controlled by a controller 28. In the illustrated embodiment, the fans 20 have a maximum power consumption of 168W per unit and are directly and uniformly manipulated by the controller 28 through 8-bit resolution PWM signals at 488 Hz. Thus, the controller 28 for the cooling unit receives a temperature signal temperature sensor 26, representing a present temperature associated with the equipment, in this case the servers 14, cooled by the cooling unit 16. Optionally, where multiple temperature sensors are provided, the temperature signals may be preprocessed by a sub-processor into a single signal before being sent to the controller, or may be mathematically processed (e.g. average or weighted average) by the controller to provide an indication of the present temperature associated with the equipment.

[0017] In the illustrated embodiment, the water flow in the heat exchanger 18 is regulated by an adjustable valve 30. In the illustrated embodiment, the valve 30 is a ball valve whose aperture is changed by a local feedback loop model-based algorithm which generates electric pulses to manipulate the aperture, although other types of valve may be used. Hence, the input to the latter algorithm is the desired value of water flow calculated by the controller 28, which is coupled to the valve 30. In the illustrated embodiment, the water flow, with a maximum value near 21L/min, is measured by a sensor 32 with average resolution of 1.06L/min whose value is fed back to the water regulation algorithm; other sensors and/or resolutions may be used. In the illustrated embodiment, water is supplied by a branch of the building's water system, and its temperature is regulated by an outside controller (not shown) using cooling tower technology. Since the outside controller is not designed for delivering constant water temperature, there are changes in the water inlet temperature of the heat exchanger 18 in the

system 10, which can have significant impact on the system. Therefore, the water inlet temperature is considered as a disturbance for the system. In other embodiments, water or other coolant may come from a dedicated supply, which may have greater precision with respect to temperature.

5 [0018] The controller 28 uses a thermal system model 34 to calculate a projected future temperature 36 of the equipment (e.g. servers 14) cooled by the cooling unit 16. The thermal system model 34 may be a static model or a dynamic model, and preferably is an adaptive model, for example an adaptive predictive model as described below. A temperature signal 38, a liquid coolant flow signal 40 and a gas coolant flow signal 42 are inputs to the thermal
10 system model 34. The liquid coolant flow signal 40 and the gas coolant flow signal 42 may be actual flow measurements, e.g. the liquid coolant flow signal 40 from the flow sensor 32, or may be surrogate/proxy values, for example PWM or fan speed as a proxy for gas coolant flow. In response to the projected future temperature 36, the controller 28 adjusts the coolant flow through the cooling unit 16 to target the projected future temperature 36 toward a set-
15 point 44. For example the controller 28 may compare the projected future temperature 36 to the set-point 44. Adjustment of the coolant flow through the cooling unit 16 by the controller 28 may comprise adjusting the liquid coolant flow and the gas coolant flow independently of one another, or there may be a mathematical relationship between the liquid coolant flow and the gas coolant flow.

20 [0019] The thermal system model 34 may include a cooling control function model 46 for selectively cooperating the liquid coolant flow and the gas coolant flow to target the set-point 38 according to an optimization parameter, which may be, for example, minimization of energy consumption, maximization of speed of set-point targeting, or another parameter.

25 [0020] The above-described server rack 12 and cooling unit 16 are merely illustrative embodiments provided for the purpose of illustration, and are not intended to be limiting in any way. Other configurations of server racks and cooling units may also be used. Moreover, while the present technology is well-suited to information technology equipment (ITE) applications and hence an illustrative implementation is described in respect of a server rack, the present technology is not so limited, and may be used in cooling applications generally.

[0021] Implementation of an exemplary thermal system model will now be described.

Adaptive Predictive Controller (APC) Formulation

[0022] One illustrative formulation for an APC is based on the Weighted Recursive Least Squares (WRLS) and Generalized Predictive Controller (GPC) algorithms. The characteristics of both algorithms will now be briefly described.

Weighted Recursive Least Squares.

[0023] The classic WRLS algorithm estimates the parameters of a plant with the following form: $\hat{y}(k) = \boldsymbol{\varphi}^T(k)\boldsymbol{\theta}(k)$, where $\boldsymbol{\varphi}(k) \in \mathbb{R}^n$ denotes the vector of outputs or measured variables, and $\boldsymbol{\theta}(k) \in \mathbb{R}^n$ is the current estimate of the coefficients. The general objective of the algorithm is to recursively update the value of $\boldsymbol{\theta}(k)$ to minimize the modeling error:

$J(k) = \sum_{i=0}^{\infty} \lambda^i \hat{e}^2(k-i)$, where $\hat{e}(k-i)$ is the error between the actual, $y(k-i)$, and the estimated, $\hat{y}(k-i)$, outputs and $\lambda \in (0,1]$ is a forgetting factor used to place greater emphasis on recent errors. The WRLS algorithm is summarized in equations (1)-(3).

$$\boldsymbol{\theta}(k) = \boldsymbol{\theta}(k-1) + \mathbf{G}(k)(y(k) - \boldsymbol{\varphi}^T(k)\boldsymbol{\theta}(k-1)), \quad (1)$$

$$\mathbf{G}(k) = \frac{\mathbf{P}(k-1)\boldsymbol{\varphi}(k)}{\lambda + \boldsymbol{\varphi}(k)^T \mathbf{P}(k-1) \boldsymbol{\varphi}(k)}, \quad (2)$$

$$\mathbf{P}(k) = \frac{\mathbf{P}(k-1) - \mathbf{G}(k)\boldsymbol{\varphi}(k)^T \mathbf{P}(k-1)}{\lambda}, \quad (3)$$

where $\mathbf{P}(k) \in \mathbb{R}^{n \times n}$ is the inverse of the estimated covariance matrix, denoted as $\mathbf{M}(k)$, around $\mathbf{0} \in \mathbb{R}^n$ and $\mathbf{G}(k) \in \mathbb{R}^n$ is the gradient descent direction, also known as the gain vector used to update $\boldsymbol{\theta}(k)$. When there is initial information available, the value of $\boldsymbol{\theta}(0)$ and $\mathbf{P}(0)$ can be computed using a standard least squares algorithm. Otherwise, they are initialized as $\boldsymbol{\theta}(0) = \mathbf{0}$ and $\mathbf{P}(0) = c\mathbf{I}$, where $c \geq 1$ and $\mathbf{I}$ is the identity matrix.

Generalized Predictive Controller

[0024] The GPC algorithm can be traced back to D. W. Clarke, C. Mohtadi and P. S. Tuffs, "Generalized Predictive Control Part II. Extensions and Interpretations*," *Automatica*, vol. 23, no. 2, pp. 149-160, 1987 and D. W. Clarke, C. Mohtadi and P. S. Tuffs, "Generalized Predictive Control Part I. The Basic Algorithm," *Automatica*, vol. 23, no. 2, pp. 137-148, 1987 (collectively, "D. W. Clarke et al."), and has been a popular predictive control algorithm, with a wide variety of applications. For a MISO system with m inputs, considering a prediction horizon γ and a control horizon $\gamma_c \leq \gamma$, the forecasted behavior of the system $\hat{\mathbf{y}}(k + \gamma|k)$ made by the GPC can be summarized in equations (4) and (5).

$$\hat{\mathbf{y}}(k + \gamma|k) = \mathbf{A}\mathbf{y}_{past}(k) + \mathbf{B}\mathbf{u}_{past}(k) + \mathbf{H}\mathbf{u}_{future}(k), \quad (4)$$

or the most commonly used

$$\hat{\mathbf{y}}(k + \gamma|k) = \mathbf{A}\mathbf{y}_{past}(k) + \mathbf{B}\mathbf{u}_{past}(k) + \mathbf{H}_r\mathbf{u}(k) + \mathbf{G}\Delta\mathbf{u}_{future}(k), \quad (5)$$

where

$$\hat{\mathbf{y}}(k + \gamma|k) = \begin{bmatrix} \hat{y}(k + 1) \\ \vdots \\ \hat{y}(k + \gamma) \end{bmatrix}, \quad \mathbf{y}_{past}(k) = \begin{bmatrix} y(k) \\ \vdots \\ y(k - a) \end{bmatrix},$$

$$\mathbf{u}_{past}(k) = \begin{bmatrix} \mathbf{u}_1(k) \\ \vdots \\ \mathbf{u}_m(k) \end{bmatrix}, \quad \mathbf{u}_j(k) = \begin{bmatrix} u_j(k) \\ \vdots \\ u_j(k - b_j) \end{bmatrix},$$

$$\mathbf{u}_{future}(k) = \begin{bmatrix} \mathbf{u}(k + 1) \\ \vdots \\ \mathbf{u}(k + \gamma_c) \end{bmatrix}, \quad \mathbf{u}(k + j) = \begin{bmatrix} u_1(k + j) \\ \vdots \\ u_m(k + j) \end{bmatrix},$$

$$\mathbf{u}(k) = \begin{bmatrix} u_1(k) \\ \vdots \\ u_m(k) \end{bmatrix}, \quad \Delta\mathbf{u}_{future}(k) = \begin{bmatrix} \Delta\mathbf{u}(k + 1) \\ \vdots \\ \Delta\mathbf{u}(k + \gamma_c) \end{bmatrix},$$

$$\Delta\mathbf{u}(k + j) = \begin{bmatrix} \Delta u_1(k + j) \\ \vdots \\ \Delta u_m(k + j) \end{bmatrix}$$

[0025] The values a and b_j represent the number of previous instances of the output and input j , respectively. The matrices $\mathbf{A} \in \mathbb{R}^{\gamma \times a}$, $\mathbf{B} \in \mathbb{R}^{\gamma \times m(b_1 + \dots + b_m)}$, $\mathbf{H}$ and $\mathbf{G} \in \mathbb{R}^{\gamma \times (m \cdot \gamma_c)}$ can be obtained using the recursive formulation given in D. W. Clarke et al. Part II, and $\mathbf{H}_r \in \mathbb{R}^{\gamma \times m}$ is a submatrix of $\mathbf{G}$. Also, it is important to mention that, as illustrated in D. W. Clarke et al. Part II, $\mathbf{H}$ and $\mathbf{G}$ are lower block triangular matrices with the same structure, shown in equation (6), where $\mathbf{g}_j \in \mathbb{R}^{1 \times m}$.

$$\mathbf{G} = \begin{bmatrix} \mathbf{g}_1 & \mathbf{0}^{1 \times m} & \dots & \mathbf{0}^{1 \times m} \\ \mathbf{g}_2 & \mathbf{g}_1 & \ddots & \vdots \\ \vdots & \mathbf{g}_2 & \ddots & \mathbf{g}_1 \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{g}_\gamma & \mathbf{g}_{\gamma-1} & \dots & \mathbf{g}_{\gamma-\gamma_c+1} \end{bmatrix} \quad (6)$$

[0026] By considering the current conditions of the system, the GPC algorithm can compute the future values of the manipulated variables, so that the predicted behavior of the system closely tracks a desired vector of set-points $\mathbf{w} \in \mathbb{R}^\gamma$, i.e. $\hat{\mathbf{y}}(k + \gamma|k) \approx \mathbf{w}$. The input sequence is computed by minimizing the cost function shown in equation (7) whose analytical solution is given by equation (8).

$$J(k) = (\mathbf{w} - \hat{\mathbf{y}}(k + \gamma|k))^T \mathbf{Q} (\mathbf{w} - \hat{\mathbf{y}}(k + \gamma|k)) + \Delta \mathbf{u}_{future}^T(k) \mathbf{R} \Delta \mathbf{u}_{future}(k) \quad (7)$$

$$\Delta \mathbf{u}_{future}(k) = (\mathbf{G}^T \mathbf{Q} \mathbf{G} + \mathbf{R})^{-1} \mathbf{G}^T \mathbf{Q} (\mathbf{w} - \mathbf{A} \mathbf{y}_{past}(k) - \mathbf{B} \mathbf{u}_{past}(k) - \mathbf{H}_r \mathbf{u}(k)) \quad (8)$$

[0027] The matrices $\mathbf{Q}, \mathbf{R} \in \mathbb{R}^{\gamma \times \gamma}$ are positive semi-definite block diagonal matrices that assign weights to future errors and penalize the input rate of change. It is important to note that like most predictive-based algorithms, GPC is of receding-horizon form, which implies that at each sampling time the value of the sequence $\mathbf{u}_{future}(k)$ is recomputed.

[0028] When the control horizon γ_c is set to 1, only one increment in the inputs is considered, after which no further changes will be implemented, implying that a constant input $\mathbf{u}$ will be held during the prediction horizon. In this case, since $\mathbf{u}(k + j) = \mathbf{u}, \forall j \leq \gamma_c$, using (6) the matrix $\mathbf{G}$ is reduced to $[\mathbf{g}_1^T \dots \mathbf{g}_\gamma^T]^T$ and the complexity of computing equation (8) is significantly decreased.

[0029] In *D. W. Clarke et al. Part II* it is explained that the selection of $\mathbf{u}$ made by the GPC algorithm when $\gamma_c = 1$ would be the optimal “mean-level” controller that places the system output close to the set-point. Hence, for stable systems with possible dead-time, $\gamma_c = 1$ in general would compute a near optimal solution due to the receding-horizon approach of the GPC algorithm.

[0030] The next portion of the disclosure describes a novel APC which combines the previous formulation with the use of a variable forgetting factor and a variable prediction horizon for a specific case of the GPC algorithm. This portion of the disclosure also describes how various specific implementation issues can be handled within the proposed framework when there is low excitation or richness of data in the RLS algorithm, which can lead to poor performance and computational instability. In addition, the modified formulation is compared against classical formulations in simulation. Then, the proposed controller algorithm is implemented in a low-cost microcontroller which is installed in the illustrative cooling system described above. The novel APC is integrated with a monetary cost reduction function within the predictive formulation framework. Finally, the performance of the proposed framework is compared with the nominal approach.

Formulation Changes

(1) Variable Forgetting Factor

[0031] The idea of a variable forgetting factor has been addressed in W. W. Woo, S. A. Svoronos and O. D. Crisalle, "A Directional Forgetting Factor for Single-Parameter Variations," in *Proceedings of the American Control Conference*, Seattle, 1995, A. K. Rao and Y.-F. Huang, "ARMA Parameter Estimation Using a Novel Recursive Estimation Algorithm with Selective Updating," *IEEE Transactions on Acoustics, Speech, And Signal Processing*, vol. 38, no. 3, pp. 447-457, 1990 and S. Bittanti and P. C. M. Bolzern, "Convergence and Exponential Convergence of Identification Algorithms with Directional Forgetting Factor," *Automatica*, vol. 26, no. 5, pp. 929-932, 1990. The approach described herein is simpler compared to these previous methods, making it easier to implement but at the same time robust enough, by avoiding overfitting too few samples. For a variable forgetting factor, two important

points should be considered. First, it should be able to be maintained close to 1, so that the adaptive error is low enough that relevant information is not lost. Secondly, it should avoid values too close to 0, to avoid overfitting new data.

[0032] The proposed variable forgetting factor, $\lambda(k)$, considers these points by introducing three constraints. First, a minimum time window w_{min} of information is enforced. Secondly, a maximum adaptive error $\hat{e}_{max}$ and, finally a minimum adaptive error $\hat{e}_{min}$ are introduced, so that $\hat{e}(k) \geq \hat{e}_{max}$ will imply that $\lambda(k)$ will be assigned a minimum value, λ_{min} . The method will forget previous data but will remember at least the information from the previous w_{min} sampling times. Also, $\hat{e}(k) \leq \hat{e}_{min}$ implies that $\lambda(k) \approx 1$. Next, the present disclosure describes how these ideas are implemented, omitting k to simplify notation when possible.

[0033] First, note that in the cost function of the WRLS algorithm, the factor λ^i is the weight associated with $\hat{e}^2(k-i)$. Hence, it is not difficult to see that the first m terms have a weight of $\frac{1-\lambda^{m+1}}{1-\lambda}$ and the remainder $\frac{\lambda^{m+1}}{1-\lambda}$, or relative proportion $1 - \lambda^{m+1}$ and λ^{m+1} , respectively. Note that it is possible to assign a minimum desired fraction p_{min} to the last terms with a minimum desired time window w_{min} so that $\lambda \geq \lambda_{min} = (p_{min})^{\frac{1}{r}}$, where $r = \frac{w_{min}}{T_{sampling}}$ and $T_{sampling}$ is the sampling period. This can be done by using a monotonically decreasing function $f(\hat{e}) \geq 1$ that satisfies equation (9).

$$\lambda^{f(\hat{e})} = \lambda_{min} \quad (9)$$

[0034] The function $f(\hat{e})$ should be able to rapidly reach a maximum value, denoted by K , when $\hat{e}$ is “close” to $\hat{e}_{min}$ so that $\lambda \approx 1$ and also, it should be capable of rapidly reaching a value close to 1 when $\hat{e}$ is close to $\hat{e}_{max}$. Based on these properties, $f(\hat{e})$ is chosen as in equation (10), which allows the time window to be extended as appropriate.

$$f(\hat{e}) = K^{\left(\frac{\hat{e}-\hat{e}_{min}}{\hat{e}_{max}-\hat{e}_{min}}\right)} \quad (10)$$

[0035] As desired, $f(\hat{e})$ has a minimum value of 1 and a maximum value of K . Also, note that $\lambda^{f(\hat{e})r} = p_{min}$, from this it is possible to conclude that Kw_{min} represents the maximum time

window considered in the algorithm, which implies that for λ approaching 1, K does not need to be too large, simplifying the computation of λ .

[0036] Since redundant information should be avoided, to make $\mathbf{P}(k)$ computationally stable (avoid numerical overflow), an additional parameter $\hat{e}_z$ was added so that the adaptive algorithm is executed only when $\hat{e}_z < \hat{e}$, otherwise $\boldsymbol{\theta}(k) = \boldsymbol{\theta}(k-1)$ and $\mathbf{P}(k) = \mathbf{P}(k-1)$. Here, $\hat{e}_z$ represents the acceptable adaptive error below which no information needs to be added to the adaptive algorithm.

(2) *Variable prediction horizon.*

[0037] The idea of using a variable prediction horizon in MPC has been addressed; see R. C. Shekhar and J. M. Maciejowski, "Robust variable horizon MPC with move blocking," *Systems & Control Letters*, vol. 61, pp. 587-594, 2012 and S. S. KEERTH and E. GILBERT, "Optimal Infinite-Horizon Feedback Laws for a General Class of Constrained Discrete-Time Systems: Stability and Moving-Horizon Approximations," *Journal of Optimization Theory and Applications*, vol. 57, no. 2, pp. 265-293, 1988.

[0038] A variable prediction horizon for the GPC is described. Based on the formulation and assumptions of the GPC for a MISO system with m inputs, when the control horizon γ_c is set to 1, the computational cost of calculating $\Delta \mathbf{u}(k+1)$ is significantly decreased. However, this computation can be further simplified by setting all elements in $\mathbf{R}$ to one, and all elements in $\mathbf{Q}$ to zero except for the last element, which is set to 1. Under this specific case the cost function in equation (7) is simplified and can be solved to find an analytical solution in equation (11), with a solution for the corresponding control update given in equation (12).

$$\mathbf{g}_\gamma \Delta \mathbf{u}(k+\gamma) = w_\gamma - \mathbf{a}_\gamma \mathbf{y}_{past}(k) - \mathbf{b}_\gamma \mathbf{u}_{past}(k) - \mathbf{g}_\gamma \mathbf{u}(k) \quad (11)$$

$$\Delta \mathbf{u}(k+\gamma) = \frac{\mathbf{g}_\gamma^T}{\mathbf{g}_\gamma \mathbf{g}_\gamma^T} (w_\gamma - \mathbf{a}_\gamma \mathbf{y}_{past}(k) - \mathbf{b}_\gamma \mathbf{u}_{past}(k) - \mathbf{g}_\gamma \mathbf{u}(k)) \quad (12)$$

[0039] The main advantage of equation (12) is that it only considers the last set-point in the prediction horizon. Hence, it is important to select an appropriate prediction horizon γ , so that

γ can be small enough to make the system achieve the desired set-point as fast as possible and, at the same time, large enough to avoid generating infeasible solutions $\mathbf{u}(k+1) \notin \mathcal{U}$. The latter tradeoff can be handled by exploiting the simplicity of equation (12).

[0040] From now on, $(w_\gamma - \mathbf{a}_\gamma \mathbf{y}_{past}(k) - \mathbf{b}_\gamma \mathbf{u}_{past}(k) - \mathbf{g}_\gamma \mathbf{u}(k))$ will be denoted as τ_γ , and it will be assumed that the set of feasible inputs $\mathcal{U}$ is a hyperbox. Then, by introducing a set of prediction horizons $\Gamma = \{\gamma_{min}, \gamma_{min} + 1, \dots, \gamma_{max}\}$, each $\Delta \mathbf{u}^i = \Delta \mathbf{u}(k+i)$ corresponding to each $i \in \Gamma$ can be computed using equation (12). Then, the set of elements $\mathcal{U}_\Delta = \{\Delta \mathbf{u}^{\gamma_{min}}, \Delta \mathbf{u}^{\gamma_{min}+1}, \dots, \Delta \mathbf{u}^{\gamma_{max}}\}$ can be created, representing the control increment vectors obtained at prediction horizon values ranging from γ_{min} to γ_{max} . For each element $\Delta \mathbf{u}^i = [\Delta u_1^i, \dots, \Delta u_m^i]^T$ in $\mathcal{U}_\Delta$ it is possible to compute its corresponding input $\mathbf{u}^i = [u_1^i, \dots, u_m^i]^T$. From here, there exist two cases that must be considered.

[0041] First, if the system has a single input, after computing $\mathbf{u}^i$ it is also necessary to compute $\mathbf{u}_{sat}^i = SAT(\mathbf{u}^i, \mathbf{u}_{min}, \mathbf{u}_{max})$, where $SAT(\cdot)$ is a function that saturates $\mathbf{u}^i$ so that $\mathbf{u}_{min} \leq \mathbf{u}_{sat}^i \leq \mathbf{u}_{max}$. Then, the prediction horizon γ^* is selected according to equation (13).

$$\gamma^* = \underset{i \in \Gamma}{\operatorname{argmin}} |\tau_i - \mathbf{g}_i \mathbf{u}_{sat}^i| \quad (13)$$

[0042] In the case that there exists more than one element $i \in \Gamma$ which minimizes $|\tau_i - \mathbf{g}_i \mathbf{u}_{sat}^i|$, the minimum value is selected for γ^* . This selection is justified by the fact that the smallest elements of Γ will create less *posteriori* error with higher probability, since any small error between $\boldsymbol{\theta}(k)$ and the ideal $\boldsymbol{\theta}^*(k)$ is propagated to $\mathbf{A}$, $\mathbf{B}$, $\mathbf{H}$ and $\mathbf{G}$ which are constructed recursively. Hence, the bigger j is, the bigger the mismatch in the prediction (with high probability).

[0043] For the second case, when the system has more than one input and the solution obtained using the pseudo-inverse is not feasible i.e. $\mathbf{u}^i \notin \mathcal{U}$, it is possible to add one additional step before implementing the saturation function, in order to convert the vector $\mathbf{u}^i$ into a vector $\mathbf{u}_f^i$ with the properties $\mathbf{g}_i \Delta \mathbf{u}^i = \mathbf{g}_i \Delta \mathbf{u}_f^i$ and $\mathbf{u}_f^i \in \mathcal{U}$. The latter process can be implemented by observing that the pair $(\mathbf{g}_i, \tau_i)$ represent the parameters of a hyperplane, the

vertices of hyperbox $\mathcal{U}$ are defined by the closed intervals formed in $(\mathbf{u}_{min}, \mathbf{u}_{max})$, and $\mathbf{u}^i = \Delta \mathbf{u}^i + \mathbf{u}(k)$, where $\mathbf{u}(k)$ is the solution implemented in the previous iteration of the algorithm. Hence, it is necessary to find a point in the hyperplane that is inside the boundaries of the hyperbox, as shown in Figure 2, which shows a two-dimensional representation of the possible infinite set of solutions that arise in the proposed approach when $\mathbf{u}^i \in \mathcal{U}$ (left) and $\mathbf{u}^i \notin \mathcal{U}$ (right).

[0044] By observing Figure 2 it is possible to identify that there can be an infinite number of candidate solutions for the value of $\mathbf{u}_f^i$, no solutions or one solution. When there are an infinite number of solutions, the main task is to select the closest solution to $\mathbf{u}^i$ so that the smallest feasible $\Delta \mathbf{u}_f^i$ change in inputs can be achieved. In this way, the candidates are reduced to points in the intersection of the hyperplane and the hyperbox. First, the p faces of the hyperbox that are crossed are identified by using equation (14).

$$S = \left\{ s \mid (1 \leq s \leq m) \wedge ([\mathbf{u}^i]_s > [\mathbf{u}_{max}]_s \vee [\mathbf{u}^i]_s < [\mathbf{u}_{min}]_s) \right\} \quad (14)$$

[0045] Then, the hyperplane l of dimension $m - p$ is defined by the pair $(\check{\mathbf{g}}_i, \check{\tau}_i)$ where $\check{\mathbf{g}}_i$ is the original vector $\mathbf{g}_i$ with components $s \in S$ deleted and $\check{\tau}_i = \tau_i - \sum_{s \in S} [\mathbf{u}_{min-max}]_s [\mathbf{g}_i]_s$, where $\mathbf{u}_{min-max}$ denotes either $\mathbf{u}_{min}$ or $\mathbf{u}_{max}$. Then, assuming at least one of the components of $\check{\mathbf{g}}_i$ is non-zero, the vector $\Delta \mathbf{u}_f^i$ that connects $\mathbf{u}^i$ to the closest point in l is computed using equations (15) and (16).

$$\Delta \check{\mathbf{u}}_f^i = \frac{\check{\mathbf{g}}_i^T}{\check{\mathbf{g}}_i \check{\mathbf{g}}_i^T} \check{\tau}_i \quad (15)$$

$$[\Delta \mathbf{u}_f^i]_s = [\Delta \check{\mathbf{u}}_f^i]_s, \forall s \notin S, [\Delta \mathbf{u}_f^i]_s = [\mathbf{u}_{min-max}]_s, \forall s \in S \quad (16)$$

[0046] Finally, using $\mathbf{u}_f^i = \mathbf{u}^i + \Delta \mathbf{u}_f^i$ the corrected solution can be calculated. If, after this process $\mathbf{u}_f^i \notin \mathcal{U}$, then equations (14)-(16) are carried out again but this time using the hyperplane defined by the l parameters $(\check{\mathbf{g}}_i, \check{\tau}_i)$ instead of $(\check{\mathbf{g}}_i, \tau_i)$, so that the constraints already considered are not violated. Once this is carried out for each possible value of i , the

set $\Gamma_{\mathcal{F}} = \{i | i \in \Gamma \wedge \mathbf{u}_{\mathcal{F}}^i \text{ exists}\}$ is created from which the prediction horizon γ^* is obtained using equation (17).

$$\gamma^* = \min \Gamma_{\mathcal{F}} \quad (17)$$

[0047] If the set $\Gamma_{\mathcal{F}}$ is empty, due to no intersections between any hyperplane $(\tilde{\mathbf{g}}_i, \tau_i)$ and the hyperbox $\mathcal{U}_{\mathcal{F}}$, implying infeasible inputs for the whole range $[\gamma_{min}, \gamma_{max}]$, then j is selected based on equation (13).

3) Low excitation and low information diversity

[0048] With the use of a variable forgetting factor, the WRLS algorithm is better at handling low excitation situations. However, this does not imply that the eigenvalues of the matrix $\mathbf{P}$ will be greater than zero at all sample times in practical implementations. Hence, two algorithms are described which can be implemented independent of each other creating an increase in the data diversity and assuring the eigenvalues of $\mathbf{P}$ will remain greater than zero.

[0049] First, it is observed that when there is a feasible solution $\Delta \mathbf{u}^i$ or modified feasible solution $\Delta \mathbf{u}_{\mathcal{F}}^i$, there are usually an infinite number of solutions. Hence, when there exists steady state error (defined under some criterion), one can randomly select one of the solutions. This random selection can be implemented by finding all possible intersections between the edges of the hyperbox $\mathcal{U}$ and the hyperplane $(\mathbf{g}_i, \tau_i)$, and computing a random weighted average of them to select a new $\Delta \mathbf{u}^i$ which will be in the convex hull generated by the intersection. Looking for intersections at each edge for the case of m inputs would require $O(m2^{m-1})$ iterations, which is manageable for small m . However, for $m \geq 4$ a more sophisticated algorithm such as the one in C. Lara, J. J. Flores and F. Calderon, "On the Hyperbox – Hyperplane Intersection Problem," *INFOCOMP*, vol. 8, no. 4, pp. 21-27, 2009 can be implemented. The steady state error identification criterion which is utilized in this work considers a weighted average of the last k_s errors.

[0050] The second algorithm keeps track of the trace of the matrix $\mathbf{P}$, since the lack of diversity in $\boldsymbol{\varphi}$ could create a situation in which some of the eigenvalues of the covariance matrix $\mathbf{M}$ become close to zero due to hardware resolution. Hence, given a minimum

eigenvalue μ_{min} (defined by the user and hardware-dependent), if $Tr(\mathbf{P}) \geq \frac{m}{\mu_{min}}$ then a multiple of the identity matrix $c\mathbf{I}$ is “injected” so that $c \geq \mu_{min}$ and $\mathbf{P}' = \mathbf{D}(c\mathbf{I} + \boldsymbol{\mu})^{-1}\mathbf{D}^T$, where $\mathbf{D}\boldsymbol{\mu}\mathbf{D}^T$ is the spectral decomposition of $\mathbf{M}$. The “injection” of $c\mathbf{I}$ into $\mathbf{P}$ can easily be done by running the step of the WRLS algorithm given in (3) m times, with $\lambda = 1$ and using each of the m rows of $c\mathbf{I}$ as input to the algorithm.

4) Knowledge-based output and predictive constraints.

[0051] One of the main issues with the adaptive algorithm in the APC approaches is that, even when the adaptive error is close to zero, the use of a limited amount of information can generate coefficients $\boldsymbol{\theta}(k)$ which do not necessarily represent the true dynamics of the system, which in turn can have a significant impact on the performance of the predictive algorithm. Therefore, there are two options that can be considered to avoid this problem, constraining $\boldsymbol{\theta}(k)$ to a specific space or establishing constraints for $\mathbf{g}_i$ in the predictive algorithm. Without foreclosing the use of the former, the present disclosure will focus on the latter.

[0052] Typically, the output of the system is physically constrained so that $y_{min} \leq y(k) \leq y_{max}(k), \forall k$. Hence, in the predictive algorithm, the following constraint is imposed: $v_i = SAT(\mathbf{a}_i\mathbf{y}_{past}(k) + \mathbf{b}_i\mathbf{u}_{past}(k) + \mathbf{g}_i\mathbf{u}(k), y_{min}, y_{max})$ and $r_i = w_i - v_i$. In addition, since (11) captures in $\mathbf{g}_i$ the physical effect that each manipulated variable has on the output, it is possible to constrain the values of $\mathbf{g}$ by setting $\mathbf{g}_i^c = SAT(\mathbf{g}_i, \mathbf{g}_{min}, \mathbf{g}_{max})$.

[0053] Even when $\mathbf{g}_{min}$ and $\mathbf{g}_{max}$ are not known, most industrial plants do have a minimum and maximum gain for each input variable, as a result of physical constraints. Hence, by selecting these minimum and maximum possible gains a lower bound for the true $\mathbf{g}_{min}$ and an upper bound for the true $\mathbf{g}_{max}$, information can be obtained or estimated with much less effort by the user in general. Based on this, the negative effects of transitory lack of information and large perturbations that dramatically change the value of $\boldsymbol{\theta}(k)$ in a sudden manner that propagate to $\mathbf{g}_i$ can be decreased.

[0054] For cases in which the sign of the gain of the manipulated variables is known and constant, constraining $\mathbf{g}_i$ can avoid counterintuitive controller actions that do not prevent

achieving the desired set-point value but waste unnecessary energy. Such cases can occur for a multiple input system in which the output behavior of one variable can be attributed to multiple inputs, and limited information about the inputs create cases such as attributing positive effects to a negative effect which is compensated by other input variables.

5) Monetary cost reduction algorithm.

[0055] Based on equation (11) it is possible to implement an algorithm that minimizes the monetary cost rate $C_{\S}(\mathbf{u}(k))$ of the manipulated variables. Assuming the monetary cost function is differentiable, the gradient descent algorithm can be implemented restricted to be orthogonal to $(\mathbf{g}_j^c)^T$. The resulting direction for minimizing $C_{\S}(\mathbf{u}(k))$ is then observed in (18).

$$-\nabla C_{\S}(\mathbf{u}(k)) + \frac{(\mathbf{g}_j^c)^T}{\mathbf{g}_j^c(\mathbf{g}_j^c)^T} \mathbf{g}_j^c \nabla C_{\S}(\mathbf{u}(k)) \quad (18)$$

[0056] Even though formula (18) can be used at any of the controller operations, its use is suggested only when the system output has already settled around the desired set-point, so that more freedom is given to the trajectories described by $\mathbf{u}(k)$.

Simulations

[0057] The proposed controller was first tested on a MATLAB® software simulation environment. MATLAB is a trademark of, and is available from, The MathWorks, Inc. having an address at 1 Apple Hill Drive, Natick, MA 01760-2098. The system considered for the simulations has similar characteristics to the physical experimental test-bed described above with reference to Figure 1. The model is composed of independent linearized subsystems of water flow, water inlet temperature and PWM signals with respect to the output temperature of the system. Also, simulations of the controller replacing the above-described predictive algorithm according to the present disclosure with the original version of the GPC were performed. For this case the parameters of the GPC were set to $\gamma_c = 1$, $\mathbf{Q} = \mathbf{R} = \mathbf{I}$.

[0058] The parameters for both controllers were set as follows. For the adaptive algorithm, $\hat{e}_{min} = 0.05$, $\hat{e}_{max} = 0.2$, $\hat{e}_z = 0.01$, $p_{min} = 0.1$, $w_{min} = 350s$ and $\mu_{min} = 0.001$. The

model considered 8 previous values of outputs and inputs for a total of 24 coefficients, since the water inlet temperature variable is not considered for the controllers. For both predictive algorithms the prediction horizons γ and γ_{max} were set to 24. This value was selected so that the performance of the APC with standard GPC is optimized. In addition, γ_{min} was set to 8
 5 and the output constraints were set to $y_{min} = 15$, $y_{max} = 35$.

[0059] Since the restriction of g_i decreased the performance of standard GPC in simulations, it was implemented only in the proposed variable prediction horizon algorithm, where g_{min} and g_{max} values were set to $[-\frac{0.2}{255}, -\frac{0.2}{30}]$ and $[-\frac{10}{255}, -\frac{10}{30}]$, respectively. Random selection of inputs was implemented using the naive algorithm of looking at each edge since the number
 10 of inputs considered was low, 5 in this case. This approach was not compatible with standard GPC; therefore it was not implemented in that setting.

[0060] For the simulated system some additional physical restrictions were implemented. The output temperature was discretized to 0.06°C, which is identical to the resolution of the physical sensors used in the physical test-bed described above with reference to Figure 1.

Also, the water inlet temperature variable was set to a constant value of 12°C plus a
 15 sinusoidal function with a period of 300s and bounded random amplitude. The water flow and PWM signal values calculated by the controllers were discretized to 0.02L/min and 1 unit, respectively, and their variables were constrained to [100, 255] for PWM and [10, 27] for the water flow. Finally, a sampling time of 10 seconds was used. The results of the simulations
 20 are shown in Figure 3 and Figure 4. Figure 3 shows performance of the APC with standard GPC and the above-described APC according to the present disclosure when the random sinusoidal amplitude parameters of the water temperature are set to $\mu = 0.4^\circ\text{C}$ and $\sigma = 0.3^\circ\text{C}$. Figure 4 shows performance of the APC with standard GPC and the above-described APC according to the present disclosure when the random sinusoidal amplitude parameters of the
 25 water temperature are set to $\mu = 0.145^\circ\text{C}$ and $\sigma = 0.11^\circ\text{C}$. Thus, in Figure 3, the random amplitude of the water sinusoidal component has a normal distribution with mean absolute value of 0.4°C and standard deviation of 0.3°C. Similarly, in Figure 4 those values were changed to 0.145°C and 0.11°C, respectively.

[0061] The results observed in Figure 3 indicate two relevant aspects. First, since the APC with standard GPC considers the entire set of values $(\mathbf{h}_i^T, \tau_i)$, $1 \leq i \leq \gamma$, in an average-like approach, it tends to be slightly smoother than its counterpart. However, because it makes no individual analysis to identify which values $(\mathbf{h}_i^T, \tau_i)$ could be generating infeasible solutions, it appears to be affected by them when such a case arises. In this aspect, since the above-described APC according to the present disclosure first identifies and discards the solutions that could generate this problem, it tends to provide better regulation.

[0062] Based on Figure 4, it can be concluded that the task of correctly identifying the behavior of the system seems to be much more challenging for the APC with standard GPC. This can be deduced by observing that the reduction in the oscillations of the water temperature affected the performance of the algorithm, creating a situation in which it was not able to converge at all. On the other hand, the above-described APC according to the present disclosure copes well with the lack of frequency richness in the inputs and outputs.

[0063] The Mean Square Error of the simulation results for Figure 3 and Figure 4 can be found in Table 1 (see below) as Experiment 1 and Experiment 2, respectively.

Experimental results

[0064] The complete set of algorithms, including the valve manipulation, was implemented on an Arduino Mega microcontroller with 8KB of SRAM memory and 256KB of Flash memory, which proved to be more than enough for the current implementation, and about the same space needed by a standard APC with control horizon of 1. This is merely one example of a suitable controller which was used for experimental purposes, and is not intended to be limiting. Other suitable controllers, including but not limited to suitably-programmed general purpose computers and other processing devices, may also be used.

[0065] Note that even though the water regulation algorithm is encoded at the same level as the controller, this is transparent to the controller and is considered part of the physical system.

[0066] Based on the previous formulation and the practical changes implemented on it, some experiments were carried out on the single rack system. The algorithm was programmed on the Arduino Mega microcontroller using only 60% of the SRAM for static variables and about 30% for non-static variables, for a total of 90%. For all experiments the performance of the APC controller was compared against a multi-PID controller which was tuned for the temperature system. The multi-PID controller consists of independent single loop PID controllers for each of the manipulated variables. A general diagram of the implementation on the system of both controllers, APC and multi-PID, can be observed in Figure 5, which shows a block diagram representation of the APC and PID controllers implemented on the system. As can be seen in Figure 5, a set-point 544 is fed as input to a controller 528 for the system 510. The controller 528 also receives a temperature signal 538 from the temperature sensor 526 associated with the rack 512 and servers 514. The controller 528 in turn controls the fans 520 and the valve 530 that governs water flow.

[0067] The performance of the APC and multi-PID controllers was compared for the following system configuration. In the experiment the top 12 servers were turned on and the rest of them were turned off, the air ducts of the “off” servers were blocked to force the air to recirculate only through the “on” servers. The temperature sensor is allocated at the level of the lowest server which is on, which is also the location of the hottest temperature at the front of the system where the air recirculates. For this experiment no initial conditions were assumed, hence 400s were given to the APC to stabilize the output before the comparison was made.

[0068] The parameters for the APC were set as follows: $\hat{e}_{min} = 0.045$, $\hat{e}_{max} = 0.2$, $\hat{e}_z = 0.001$, $p_{min} = 0.1$, $w_{min} = 150$ and $\mu_{min} = 0.1$. The 8 previous values of outputs and inputs yield a total of 24 coefficients. Also, $\gamma_{max} = 14$, $\gamma_{min} = 3$, $y_{min} = 20$, $y_{max} = 40$, g_{min} and g_{max} values were set to $[-\frac{0.2}{255}, -\frac{0.2}{30}]$ and $[-\frac{10}{255}, -\frac{10}{30}]$, respectively. The water flow and PWM values calculated by the APC were discretized to 0.02L/min and 1 unit. Finally, the manipulated variables were constrained to [35,255] for PWM and [9,21]L/min for water flow. The results of the experiment can be observed in Figures 6 to 8. Figure 6 is a graph showing performance results for APC and PID from the above experiment, Figure 7 is a graph

showing PWM manipulation of APC and PID, with resolution of 1, within the range [35,255] (8 bit representation), and Figure 8 is a graph showing water flow manipulation of APC and PID, with resolution of 0.02, within the range [9,21].

[0069] After the 400s gap, when the adaptive algorithm has obtained enough information, one can observe that the APC performance tends to be better than the multi-PID in Figure 6, since it generates less overshoot when a new value for the set-point is established and it is still able to achieve the desired set-point. The decreased overshoot can be mainly attributed to the system behavior forecasting made by the predictive algorithm of the APC.

[0070] Because of the feedback relationship among the inputs and the predictive output, less oscillations occur in the manipulated variables, showing on average a more consistent behavior in the inputs for the APC algorithm. Finally, since the APC considers the effects of the manipulated variables on the output simultaneously and not independent of each other as the PID implicitly does, saturated behaviors on the inputs are also significantly decreased since a more accurate selection of their values is made.

[0071] Additional experiments were conducted to test the monetary cost reduction algorithm in formula (18). The monetary cost function of the manipulated variables for the current system have the form of $C_s(u(k)) = b_s u^j$, where b_s contains the cost of the energy used by the fans and the cost of water per liter/min, $b_s = \begin{bmatrix} 5.94(10^{-6}) \frac{\text{cents}}{\text{s}} & 6.340(10^{-3}) \frac{\text{cents}}{\text{L} \cdot \text{min}} \end{bmatrix}$. These values were obtained by using information from the province of Ontario, Canada (O. E. Board, "Ontario Energy Board," 2018 [Online] at <https://www.oeb.ca/rates-and-your-bill/electricity-rates> [Accessed 10 July 2018] and City of Toronto, "2018 Water Rates & Fees," 2018 [Online] at <https://www.toronto.ca/services-payments/property-taxes-utilities/utility-bill/water-rates-and-fees/> [Accessed 10 July 2018]).

[0072] The monetary cost reduction algorithm was activated only when the system arrived at the desired set-point and remained stable for 4 iterations. The performance of the APC with the monetary cost reduction APC_s is shown in Figure 9, and the cumulative monetary reduction through time is observed in Figure 10, which shows savings in percentage of the APC with $C_s(u(k))$ enabled with respect to APC.

[0073] From Figure 10 it is possible to observe that in the above experiment, APC_s generated a reduction near 15% of $C_s(\mathbf{u}(k))$ when compared to the APC, which could represent a significant saving for the user. Also, both APC_s and APC are similar in general performance. However, since the former slowly changes the state of the system by making small adjustments to $\mathbf{u}(k)$, it can become more vulnerable to disturbance effects caused by the water temperature, as observed in Figure 9. Despite this, when the system changes the desired set-point the algorithm which minimizes $C_s(\mathbf{u}(k))$ is not active until it returns to the set-point.

[0074] The Mean Square Error of the simulations' results for Figure 6 and Figure 9 can be found in Table 1 as Experiment 3 and Experiment 4, respectively.

	Controller	MSE
Experiment 1	standard APC	0.5611
	APC	0.4011
Experiment 2	standard APC	4.0564
	APC	0.3929
Experiment 3	PID	0.4239
	APC	0.3650
Experiment 4	APC	0.0814
	APC_s	0.1088

Table 1. MSE index of the performance of the controllers during the experiments

[0075] According to the foregoing description, an industrially feasible APC from economic and computational points of view was designed and implemented for temperature control in data centers. The proposed APC according to the present disclosure was compared against a nominal APC with similar computational cost via simulation and experiment. The simulation results indicated that improved performance was achieved on a constrained system. In order to calculate control action an economical microcontroller was utilized. The experimental results indicated that the proposed APC outperforms classical PIDs in particular in reducing the oscillations in the temperature and moderating changes in the manipulated variables. Also, an illustrative projected gradient descent-based algorithm was proposed to include monetary cost minimization to develop an APC_s controller. The experimental results of the proposed method indicated a reduction in the energy cost of the cooling unit with respect to the nominal APC.

Additional Factors

[0076] An APC according to the present disclosure can be expanded to account for additional factors which may impact future temperature. For example, in a server rack, the current drawn by the servers, or the workload assigned to the servers, may be expected to affect the temperature, albeit with a delayed effect. Thus, workload and/or current, or other delayed-effect variables may be additional inputs into the APC.

[0077] Reference is now made to Figure 11, which shows an arrangement similar to Figure 1, with like reference numerals denoting like features, except with the prefix “11”. A current sensor 1150 on the rack 1112 monitors current drawn by the servers 1114 and generates a current signal 1152 which is received at the controller 1128. The current signal 1152 represents a current drawn by the equipment (servers 1114 in this case) cooled by the cooling unit 1116, and is a further input to the thermal system model 1134. Optionally, current drawn by the servers 1114 can be measured indirectly. For example, the current signal may be, or be derived from, a CPU temperature for one or more of the servers 1114. CPU temperature is a reasonable proxy for current drawn by the server that includes that CPU. The current signal may be instantaneous current measured by a transducer, etc. or may be a measurement (with or without additional processing/calculation) by a thermocouple, thermistor, IR camera, or other suitable sensor.

[0078] The current will have a somewhat delayed effect on the measured temperature; the current will typically change faster and more frequently than the temperature and some changes in current may offset one another without materially effecting the temperature associated with the equipment. Because the current is expected to have a delayed effect on the temperature, it is preferred to modify the APC model to account for the delay.

[0079] As noted above, the classic WRLS algorithm estimates the parameters of a plant with the following form: $\hat{y}(k) = \boldsymbol{\varphi}^T(k)\boldsymbol{\theta}(k)$, where $\boldsymbol{\varphi}(k) \in \mathbb{R}^n$ denotes the vector of outputs or measured variables, and $\boldsymbol{\theta}(k) \in \mathbb{R}^n$ is the current estimate of the coefficients. This equation can be detailed further in terms of the type of values contained in $\boldsymbol{\varphi}$, as shown in equation (19):

$$\hat{y}(k) = \mathbf{a}y_{past}(k) + \mathbf{b}u_{past}(k) + \mathbf{h}u(k) \quad (19)$$

where $\hat{y}(k)$ is the output estimated behavior in the next iteration, $\mathbf{a}y_{past}(k)$ is the past output behavior, $\mathbf{b}u_{past}(k)$ is the past manipulated inputs behavior and $\mathbf{h}u(k)$ is the current manipulated inputs, and where:

$$\boldsymbol{\varphi} = \begin{bmatrix} y_{past}(k) \\ u_{past}(k) \\ u(k) \end{bmatrix} \quad \boldsymbol{\theta}(k) = \begin{bmatrix} \mathbf{a}^T \\ \mathbf{b}^T \\ \mathbf{h}^T \end{bmatrix}$$

[0080] Now, if non-manipulated variables $u(k)_{nm}$ such as power consumption (e.g. current), water inlet temperature, etc. are added then equation (19) is transformed as shown in equation (20):

$$\hat{y}(k) = \mathbf{a}y_{past}(k) + \mathbf{v}u_{past}(k) + \mathbf{h}u(k) + \mathbf{b}_{nm}u(k)_{past-nm} + \mathbf{h}_{nm}u(k)_{nm} \quad (20)$$

where $\hat{y}(k)$ is the output estimated behavior in the next iteration, $\mathbf{a}y_{past}(k)$ is the past output behavior, $\mathbf{v}u_{past}(k)$ is the past manipulated inputs behavior, $\mathbf{h}u(k)$ is the current manipulated inputs, $\mathbf{b}_{nm}u(k)_{past-nm}$ is the past non-manipulated inputs behavior (water temperature, power, etc.) and $\mathbf{h}_{nm}u(k)_{nm}$ is the current non-manipulated inputs behavior.

[0081] If the “j” non-manipulated variable has a discrete delay of order “ d_j ”, then the first “ d_j ” elements of the coefficients associated to that variable within $\mathbf{h}_{nm}$ will be close to 0. Hence, If the value of “ d_j ” is known, then the following correction can be made to the previous linear model:

$$\hat{y}(k) = \mathbf{a}y_{past}(k) + \mathbf{b}u_{past}(k) + \mathbf{h}u(k) + \mathbf{b}_{nm}^d u_{past-nm}(k-d) + \mathbf{h}_{nm}^d u_{nm}(k-d) \quad (21)$$

where $\mathbf{b}_{nm}^d$ and $\mathbf{h}_{nm}^d$ are matrices containing elements which together (in absolute value) are farther from zero than the first d_j elements of $\mathbf{h}_{nm}$ and d contains information on all of the delays of the non-manipulated variables (some of them possibly zero):

$$\mathbf{d} = \begin{bmatrix} d_1 \\ \vdots \\ d_q \end{bmatrix}$$

[0082] This modification creates an effect in the predictive equation shown as equation (4) above, which will be replaced by the following equation:

$$\hat{\mathbf{y}}(k + \gamma|k) = \mathbf{A}\mathbf{y}_{past}(k) + \mathbf{B}\mathbf{u}_{past}(k) + \mathbf{H}\mathbf{u}_{future}(k) + \mathbf{H}_{nm}^d \mathbf{u}_{past-nm}^d(k - \mathbf{d}) + \mathbf{H}_{nm}^p \mathbf{u}_{nm}^p(k - \mathbf{d}) + \mathbf{H}_{nm}^f \mathbf{u}_{nm}^f(k - \mathbf{d}) \quad (22)$$

5

where $\hat{\mathbf{y}}(k)$ is the output estimated behavior in the next iteration, $\mathbf{A}\mathbf{y}_{past}(k)$ is the past output behavior, $\mathbf{B}\mathbf{u}_{past}(k)$ is the past manipulated inputs behavior, $\mathbf{H}\mathbf{u}_{future}(k)$ is the current and future manipulated inputs, $\mathbf{H}_{nm}^d \mathbf{u}_{past-nm}^d(k - \mathbf{d})$ is the past non-manipulated inputs considering delay, $\mathbf{H}_{nm}^p \mathbf{u}_{nm}^p(k - \mathbf{d})$ is the known non-manipulated inputs behavior considering delay and $\mathbf{H}_{nm}^f \mathbf{u}_{nm}^f(k - \mathbf{d})$ is the estimated non-manipulated inputs behavior considering delay.

10

[0083] Similarly, equation (5) above would be replaced by the following equation:

$$\hat{\mathbf{y}}(k + \gamma|k) = \mathbf{A}\mathbf{y}_{past}(k) + \mathbf{B}\mathbf{u}_{past}(k) + \mathbf{H}_r \mathbf{u}(k) + \mathbf{G}\Delta \mathbf{u}_{future}(k) + \mathbf{H}_{nm}^d \mathbf{u}_{past-nm}^d(k - \mathbf{d}) + \mathbf{H}_{nm}^p \mathbf{u}_{nm}^p(k - \mathbf{d}) + \mathbf{H}_{nm}^f \mathbf{u}_{nm}^f(k - \mathbf{d}) \quad (23)$$

where:

$$\mathbf{u}_{past-nm}^d(k - \mathbf{d}) = \begin{bmatrix} \mathbf{u}_{1-nm}^d(k - d_1) \\ \vdots \\ \mathbf{u}_{q-nm}^d(k - d_q) \end{bmatrix}, \mathbf{u}_{j-nm}^d(k - d_j) = \begin{bmatrix} u_j(k - d_j) \\ \vdots \\ u_j(k - d_j - b_j) \end{bmatrix}$$

15

$$\mathbf{u}_{nm}^p(\mathbf{k} - \mathbf{d}) = \begin{bmatrix} \mathbf{u}_{1-nm}^p(k - d_1) \\ \vdots \\ \mathbf{u}_{q-nm}^p(k - d_q) \end{bmatrix}, \mathbf{u}_{j-nm}^p(k - d_j) = \begin{bmatrix} u_j(k - d_j + \min(d_j, \gamma)) \\ \vdots \\ u_j(k - d_j + 1) \end{bmatrix}$$

Note that $\mathbf{u}_{nm}^p(\mathbf{k} - \mathbf{d})$ are known values, not captured in the system model, but considered in the predictive equation that will affect the future behavior of the system.

$$\mathbf{u}_{nm}^f(\mathbf{k} - \mathbf{d}) = \begin{bmatrix} \mathbf{u}_{1-nm}^f(k - d_1) \\ \vdots \\ \mathbf{u}_{q-nm}^f(k - d_q) \end{bmatrix}, \mathbf{u}_{j-nm}^f(k - d_j) = \begin{bmatrix} u_j(k + \gamma - d_j) \\ \vdots \\ u_j(k + 1) \end{bmatrix}$$

5 [0084] If d_i is equal to 0 then $\mathbf{u}_{i-nm}^p(k - d_j)$ is merely an empty null space that should not be considered on $\mathbf{u}_{nm}^p(\mathbf{k} - \mathbf{d})$. Also, if $d_i \geq \gamma$ then $\mathbf{u}_{j-nm}^f(k - d_j)$ is merely a null space that should not be considered on $\mathbf{u}_{nm}^f(\mathbf{k} - \mathbf{d})$.

[0085] While the solution to equation (7) will remain unchanged, equation (8) will be replaced by:

$$\Delta \mathbf{u}_{future-m}(k) = (\mathbf{G}^T \mathbf{Q} \mathbf{G} + \mathbf{R})^{-1} \mathbf{G}^T \mathbf{Q} \left(\mathbf{w} - \mathbf{A} \mathbf{y}_{past}(k) - \mathbf{B} \mathbf{u}_{past}(k) - \mathbf{H}_r \mathbf{u}(k) - \mathbf{H}_{nm}^d \mathbf{u}_{past-nm}^d(\mathbf{k} - \mathbf{d}) - \mathbf{H}_{nm}^p \mathbf{u}_{nm}^p(\mathbf{k} - \mathbf{d}) - \mathbf{H}_{nm}^f \mathbf{u}_{nm}^f(\mathbf{k} - \mathbf{d}) \right) \quad (24)$$

10

[0086] Equation (11) will be replaced by:

$$\mathbf{g}_\gamma \Delta \mathbf{u}(k + \gamma) = \mathbf{w}_\gamma - \mathbf{a}_\gamma \mathbf{y}_{past}(k) - \mathbf{b}_\gamma \mathbf{u}_{past}(k) - \mathbf{g}_\gamma \mathbf{u}(k) - \mathbf{h}_{nm-\gamma}^d \mathbf{u}_{past-nm}^d(\mathbf{k} - \mathbf{d}) - \mathbf{h}_{nm-\gamma}^p \mathbf{u}_{nm}^p(\mathbf{k} - \mathbf{d}) - \mathbf{h}_{nm-\gamma}^f \mathbf{u}_{nm}^f(\mathbf{k} - \mathbf{d}) \quad (25)$$

[0087] Additionally, equation (12) will be replaced by:

$$\Delta \mathbf{u}(k + \gamma) = \frac{\mathbf{g}_\gamma^T}{\mathbf{g}_\gamma \mathbf{g}_\gamma^T} \left(\mathbf{w}_\gamma - \mathbf{a}_\gamma \mathbf{y}_{past}(k) - \mathbf{b}_\gamma \mathbf{u}_{past}(k) - \mathbf{g}_\gamma \mathbf{u}(k) - \mathbf{h}_{nm-\gamma}^d \mathbf{u}_{past-nm}^d(\mathbf{k} - \mathbf{d}) - \mathbf{h}_{nm-\gamma}^p \mathbf{u}_{nm}^p(\mathbf{k} - \mathbf{d}) - \mathbf{h}_{nm-\gamma}^f \mathbf{u}_{nm}^f(\mathbf{k} - \mathbf{d}) \right) \quad (26)$$

15 [0088] Moreover, the expression $(\mathbf{w}_\gamma - \mathbf{a}_\gamma \mathbf{y}_{past}(k) - \mathbf{b}_\gamma \mathbf{u}_{past}(k) - \mathbf{g}_\gamma \mathbf{u}(k))$, which had been denoted above as τ_γ , will be replaced by:

$$w_\gamma - \mathbf{a}_\gamma \mathbf{y}_{past}(k) - \mathbf{b}_\gamma \mathbf{u}_{past}(k) - \mathbf{h}_{nm-\gamma}^d \mathbf{u}_{past-nm}^d(k-d) - \mathbf{h}_{nm-\gamma}^p \mathbf{u}_{nm}^p(k-d) - \mathbf{h}_{nm-\gamma}^f \mathbf{u}_{nm}^f(k-d)$$

[0089] The constraint $v_i = SAT(\mathbf{a}_i \mathbf{y}_{past}(k) + \mathbf{b}_i \mathbf{u}_{past}(k) + \mathbf{g}_i \mathbf{u}(k), y_{min}, y_{max})$ is now replaced by:

$$v_i = SAT(\mathbf{a}_i \mathbf{y}_{past}(k) + \mathbf{b}_i \mathbf{u}_{past}(k) + \mathbf{g}_i \mathbf{u}(k) + \mathbf{h}_{nm-i}^d \mathbf{u}_{past-nm}^d(k-d) + \mathbf{h}_{nm-i}^p \mathbf{u}_{nm}^p(k-d) + \mathbf{h}_{nm-i}^f \mathbf{u}_{nm}^f(k-d), y_{min}, y_{max}) \quad (27)$$

[0090] The above modified equations can, like equations (11) and (12) above, be solved analytically using suitable software tools, such as those offered commercially under the trademark MATLAB®, or implemented in a suitable programming language, such as Python.

[0091] If d_j is unknown but expected to have a fixed value or only small variations, a value can be determined experimentally, possibly with periodic updates to detect changes in conditions. If d_j is expected to change over time, an algorithm can be used to detect and/or model the delay.

[0092] The above-described approach to including a delayed-effect variable as a further input to the thermal system model of an APC is not limited to current and proxies therefor. For example, where the equipment being cooled is information technology equipment such as servers, a projected server workload may also be used as a further input to the thermal system model of an APC according to the present disclosure. Signals representing current and its proxies (“current signals”) and signals representing projected server workload (“processing workload signals”) can be characterized broadly as a “workload signals”.

[0093] Figure 12 shows an arrangement similar to Figure 1, with like reference numerals denoting like features, except with the prefix “12”. A hypervisor 1260 controls processing workload assigned to the servers 1214 on the rack 1212 (as well as to other servers on other racks, not shown) and provides a processing workload signal 1262 which is received at the controller 1228. The processing workload signal 1262 represents a workload that will be assigned to the equipment (servers 1214 in this case) cooled by the cooling unit 1216, and is a

further input to the thermal system model 1234. An increase or decrease in workload can be expected to result, after a delay, in a corresponding increase or decrease in temperature.

Where the equipment is servers 1234 or similar information technology equipment, the processing workload signal 1262 may merely indicate the workload assigned to the servers 1214 in the rack 1212 as a collective, or may be more granular. For example, if the thermal system model 1234 is sensitive to heat output from individual servers (e.g. there may be temperature sensors for each server or subsets of servers), the processing workload signal 1262 may indicate workloads for individual servers or subsets of servers in the rack.

[0094] While described in the context of information technology equipment such as servers, the use of projected future workload as an input to the thermal system model can be extended to other contexts as well.

[0095] Optionally, both a current signal and a processing workload signal may be used as inputs to the thermal system model.

Method and Implementation

[0096] Current and its proxies, and projected server workload, are both examples of delayed-effect variables. A delayed-effect variable is one whose impact on the temperature of the equipment to be cooled is subject to potential delay and, unlike the gas coolant flow rate and liquid coolant flow rate, is not manipulated by the controller. The above-described mathematical approaches are not limited to “workload signals” such as current and projected server workload, but can be applied in respect of delayed-effect variables more generally. In this regard, a signal received by the controller and representing a delayed-effect variable associated with the equipment to be cooled is referred to as a “delayed-effect variable signal”.

[0097] Reference is now made to Figure 13, in which a method 1300 for controlling operational parameters of a cooling unit is shown schematically in flow chart form. The method 1300 would be executed by a controller of a cooling unit, wherein a coolant flow through the cooling unit comprises a liquid coolant flow through the cooling unit and a gas coolant flow through the cooling unit.

[0098] At step 1302, the controller receives a temperature signal representing a present temperature associated with equipment cooled by the cooling unit. At step 1304, the controller receives at least one delayed-effect variable signal representing a delayed-effect variable associated with the equipment cooled by the cooling unit. The delayed-effect variable signal(s) may be, for example, a workload signal such as a processing workload signal representing an indication of a workload that will be assigned to the equipment cooled by the cooling unit (e.g. from a hypervisor controlling information technology equipment), or a current signal representing a current drawn by the equipment cooled by the cooling unit, or a combination (e.g. two distinct signals or a signal representing a mathematic combination of the two signals). At step 1306, the controller, using a thermal system model to which the temperature signal, the delayed-effect variable signal(s), the liquid coolant flow and the gas coolant flow are inputs, calculates a projected future temperature associated with the equipment cooled by the cooling unit. The thermal system model may include a cooling control function model for selectively cooperating the liquid coolant flow and the gas coolant flow to target the set-point according to an optimization parameter, for example energy consumption or speed of set-point targeting. The thermal system model may be a static model, but is preferably an adaptive model and particularly preferably is an adaptive predictive model of the type described above.

[0099] At step 1308, responsive to the projected future temperature, the controller adjusts coolant flow through the cooling unit to target the projected future temperature toward a set-point by adjusting the liquid coolant flow and the gas coolant flow.

[00100] As can be seen from the above description, the cooling technology described herein represent significantly more than merely using categories to organize, store and transmit information and organizing information through mathematical correlations. The apparatus and methods described herein are in fact an improvement to the technology of equipment cooling, as they may provide the ability to deal with time-varying systems, to transform infeasible solutions computed by the predictive formulation into feasible solutions when there is at least one, to cope with transitory adaptation to the system, to vary the learning rate associated with new data according to the dynamics of the system, and to avoid

numerical instability when the adaptive algorithm faces low system excitation. These abilities of the technology facilitate efficient and effective cooling. Moreover, the technology is applied by using a particular machine, namely a cooling unit that includes heat exchanger with both a liquid coolant flow and a gas coolant flow. As such, the technology described herein is confined to cooling applications.

[00101] The present technology may be embodied within a system, a method, a computer program product or any combination thereof. The computer program product may include a computer readable storage medium or media having computer readable program instructions thereon for causing a processor to carry out aspects of the present technology. The computer readable storage medium can be a tangible device that can retain and store instructions for use by an instruction execution device. The computer readable storage medium may be, for example, but is not limited to, an electronic storage device, a magnetic storage device, an optical storage device, an electromagnetic storage device, a semiconductor storage device, or any suitable combination of the foregoing.

[00102] A non-exhaustive list of more specific examples of the computer readable storage medium includes the following: a portable computer diskette, a hard disk, a random access memory (RAM), a read-only memory (ROM), an erasable programmable read-only memory (EPROM or Flash memory), a static random access memory (SRAM), a portable compact disc read-only memory (CD-ROM), a digital versatile disk (DVD), a memory stick, a floppy disk, a mechanically encoded device such as punch-cards or raised structures in a groove having instructions recorded thereon, and any suitable combination of the foregoing. A computer readable storage medium, as used herein, is not to be construed as being transitory signals *per se*, such as radio waves or other freely propagating electromagnetic waves, electromagnetic waves propagating through a waveguide or other transmission media (e.g., light pulses passing through a fiber-optic cable), or electrical signals transmitted through a wire.

[00103] Computer readable program instructions described herein can be downloaded to respective computing/processing devices from a computer readable storage medium or to an external computer or external storage device via a network, for example, the Internet, a

local area network, a wide area network and/or a wireless network. The network may comprise copper transmission cables, optical transmission fibers, wireless transmission, routers, firewalls, switches, gateway computers and/or edge servers. A network adapter card or network interface in each computing/processing device receives computer readable
5 program instructions from the network and forwards the computer readable program instructions for storage in a computer readable storage medium within the respective computing/processing device.

[00104] Computer readable program instructions for carrying out operations of the present technology may be assembler instructions, instruction-set-architecture (ISA)
10 instructions, machine instructions, machine dependent instructions, microcode, firmware instructions, state-setting data, or either source code or object code written in any combination of one or more programming languages, including an object oriented programming language or a conventional procedural programming language. The computer readable program instructions may execute entirely on the user's computer, partly on the user's computer, as a
15 stand-alone software package, partly on the user's computer and partly on a remote computer or entirely on the remote computer or server. In the latter scenario, the remote computer may be connected to the user's computer through any type of network, including a local area network (LAN) or a wide area network (WAN), or the connection may be made to an external computer (for example, through the Internet using an Internet Service Provider). In some
20 embodiments, electronic circuitry including, for example, programmable logic circuitry, field-programmable gate arrays (FPGA), or programmable logic arrays (PLA) may execute the computer readable program instructions by utilizing state information of the computer readable program instructions to personalize the electronic circuitry, in order to implement aspects of the present technology.

25 [00105] Aspects of the present technology have been described above with reference to flowchart illustrations and/or block diagrams of methods, apparatus (systems) and computer program products according to various embodiments. In this regard, the flowchart and block diagrams in the Figures illustrate the architecture, functionality, and operation of possible implementations of systems, methods and computer program products according to various

embodiments of the present technology. For instance, each block in the flowchart or block diagrams may represent a module, segment, or portion of instructions, which comprises one or more executable instructions for implementing the specified logical function(s). It should also be noted that, in some alternative implementations, the functions noted in the block may occur out of the order noted in the Figures. For example, two blocks shown in succession may, in fact, be executed substantially concurrently, or the blocks may sometimes be executed in the reverse order, depending upon the functionality involved. Some specific examples of the foregoing may have been noted above but any such noted examples are not necessarily the only such examples. It will also be noted that each block of the block diagrams and/or flowchart illustration, and combinations of blocks in the block diagrams and/or flowchart illustration, can be implemented by special purpose hardware-based systems that perform the specified functions or acts, or combinations of special purpose hardware and computer instructions.

[00106] It also will be understood that each block of the flowchart illustrations and/or block diagrams, and combinations of blocks in the flowchart illustrations and/or block diagrams, can be implemented by computer program instructions. These computer readable program instructions may be provided to a processor of a general purpose computer, special purpose computer, or other programmable data processing apparatus to produce a machine, such that the instructions, which execute via the processor of the computer or other programmable data processing apparatus, create means for implementing the functions/acts specified in the flowchart and/or block diagram block or blocks. Thus, while an Arduino Mega was referenced as an illustrative controller, this is merely one illustrative example and not a limiting one; the term “controller” is intended to refer broadly to any device capable of executing the required processing.

[00107] These computer readable program instructions may also be stored in a computer readable storage medium that can direct a computer, other programmable data processing apparatus, or other devices to function in a particular manner, such that the instructions stored in the computer readable storage medium produce an article of manufacture including instructions which implement aspects of the functions/acts specified in

the flowchart and/or block diagram block or blocks. The computer readable program instructions may also be loaded onto a computer, other programmable data processing apparatus, or other devices to cause a series of operational steps to be performed on the computer, other programmable apparatus or other devices to produce a computer
5 implemented process such that the instructions which execute on the computer or other programmable apparatus provide processes for implementing the functions/acts specified in the flowchart and/or block diagram block or blocks.

[00108] Finally, the terminology used herein is for the purpose of describing particular embodiments only and is not intended to be limiting. As used herein, the singular forms “a”,
10 “an” and “the” are intended to include the plural forms as well, unless the context clearly indicates otherwise. It will be further understood that the terms “comprises” and/or “comprising,” when used in this specification, specify the presence of stated features, integers, steps, operations, elements, and/or components, but do not preclude the presence or addition of one or more other features, integers, steps, operations, elements, components,
15 and/or groups thereof.

[00109] The corresponding structures, materials, acts, and equivalents of all means or step plus function elements in the claims below are intended to include any structure, material, or act for performing the function in combination with other claimed elements as specifically claimed. The description has been presented for purposes of illustration and
20 description, but is not intended to be exhaustive or limited to the form disclosed. Many modifications and variations will be apparent to those of ordinary skill in the art without departing from the scope of the claims. The embodiment was chosen and described in order to best explain the principles of the technology and the practical application, and to enable others of ordinary skill in the art to understand the technology for various embodiments with various modifications as are suited to the particular use contemplated.
25

[00110] One or more currently preferred embodiments have been described by way of example. It will be apparent to persons skilled in the art that a number of variations and modifications can be made without departing from the scope of the claims.

WHAT IS CLAIMED IS:

1. A method for controlling operational parameters of a cooling unit, wherein a coolant flow through the cooling unit comprises:

a liquid coolant flow through the cooling unit; and

5 a gas coolant flow through the cooling unit;

the method comprising:

receiving, at a controller for the cooling unit, at least one temperature signal representing a present temperature associated with equipment cooled by the cooling unit;

10 receiving, at the controller, at least one delayed-effect variable signal representing a delayed-effect variable associated with equipment cooled by the cooling unit;

calculating, by the controller, using a thermal system model to which the at least one temperature signal, the at least one delayed-effect variable signal, the liquid coolant flow and the gas coolant flow are inputs, a projected future temperature associated with the equipment cooled by the cooling unit; and

15 responsive to the projected future temperature, adjusting coolant flow through the cooling unit to target the projected future temperature toward a set-point.

2. The method of claim 1, wherein the thermal system model includes a cooling control function model for selectively cooperating the liquid coolant flow and the gas coolant flow to
20 target the set-point according to an optimization parameter.

3. The method of claim 2, wherein the optimization parameter is one of energy consumption and speed of set-point targeting.

4. The method of claim 1, wherein the thermal system model is a static model.

5. The method of claim 1, wherein the thermal system model is an adaptive model.

5 6. The method of claim 1, wherein:

the equipment is information technology equipment; and

the at least one delayed-effect variable signal comprises a processing workload signal representing an indication of a workload that will be assigned to the information technology equipment.

10 7. The method of claim 1, wherein the at least one delayed-effect variable signal comprises a current signal representing a current drawn by the equipment cooled by the cooling unit.

15 8. A cooling unit, comprising:

a heat exchanger;

at least one liquid flow control element adapted to control flow of liquid coolant through the heat exchanger;

at least one gas flow control element adapted to control flow of gas coolant across the heat
20 exchanger;

at least one temperature sensor adapted to sense a temperature associated with equipment to be cooled by the cooling unit; and

a controller, wherein:

the controller is coupled to the at least one temperature sensor to receive at least one temperature signal representing the temperature associated with the equipment to be cooled by the cooling unit;

5 the controller is adapted to receive at least one delayed-effect variable signal representing a delayed-effect variable associated with equipment cooled by the cooling unit;

10 the controller is coupled to the at least one liquid flow control element and adapted to manipulate the at least one liquid flow control element to control the flow of liquid coolant through the heat exchanger;

the controller is coupled to the at least one gas flow control element and adapted to manipulate the at least one gas flow control element to control the flow of gas coolant across the heat exchanger;

wherein the controller is configured to:

15 calculate, using a thermal system model to which the at least one temperature signal, the at least one delayed-effect variable signal; the liquid coolant flow and the gas coolant flow are inputs, a projected future temperature associated with the equipment cooled by the cooling unit; and

20 responsive to the projected future temperature, adjust coolant flow through the cooling unit to target the projected future temperature toward a set-point.

9. The cooling unit of claim 8, wherein the thermal system model includes a cooling control function model for selectively cooperating the liquid coolant flow and the gas coolant flow to target the set-point according to an optimization parameter.

10. The cooling unit of claim 8, wherein the thermal system model is an adaptive model.

11. The cooling unit of claim 8, wherein:

the equipment is information technology equipment; and

5 the at least one delayed-effect variable signal comprises a processing workload signal representing an indication of a workload that will be assigned to the information technology equipment.

10 12. The cooling unit of claim 8, wherein the at least one delayed-effect variable signal comprises a current signal representing a current drawn by the equipment cooled by the cooling unit.

15 13. A computer program product comprising a tangible computer-readable medium carrying instructions which, when executed by a controller coupled to a cooling unit wherein a coolant flow through the cooling unit comprises a liquid coolant flow and a gas coolant flow across, cause the controller to control operational parameters of the cooling unit by:

receiving, at the controller, at least one temperature signal representing a present temperature associated with equipment cooled by the cooling unit;

20 receiving, at the controller, at least one delayed-effect variable signal representing a delayed-effect variable associated with equipment cooled by the cooling unit;

calculating, by the controller, using a thermal system model to which the at least one temperature signal, the at least one delayed-effect variable signal, the liquid coolant flow and the gas coolant flow are inputs, a projected future temperature associated with the equipment cooled by the cooling unit; and

responsive to the projected future temperature, adjusting coolant flow through the cooling unit to target the projected future temperature toward a set-point.

14. The computer program product of claim 13, wherein the thermal system model
5 includes a cooling control function model for selectively cooperating the liquid coolant flow and the gas coolant flow to target the set-point according to an optimization parameter.

15. The computer program product of claim 14, wherein the optimization parameter is one of energy consumption and speed of set-point targeting.

10 16. The computer program product of claim 13, wherein the thermal system model is a static model.

15 17. The computer program product of claim 13, wherein the thermal system model is an adaptive model.

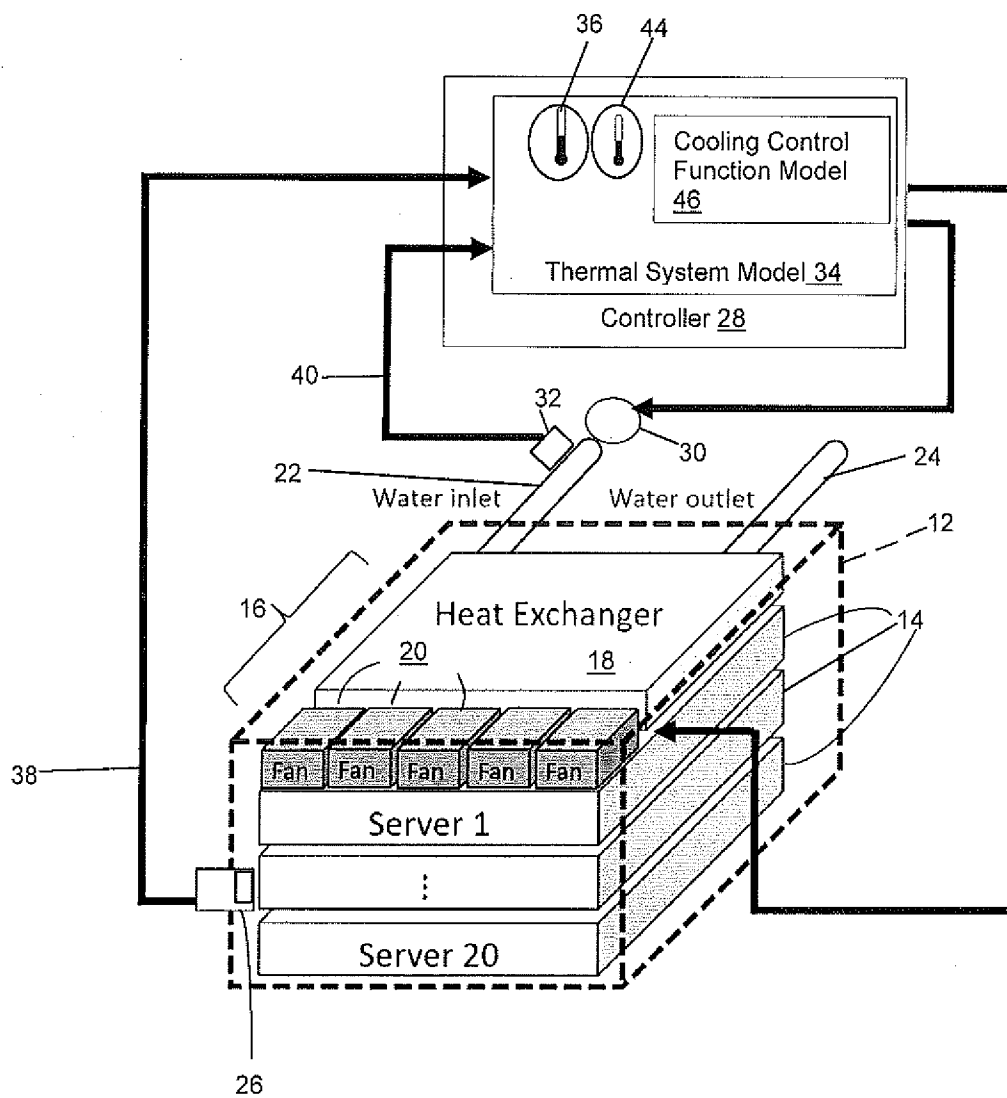
18. The computer program product of claim 13, wherein:

the equipment is information technology equipment; and

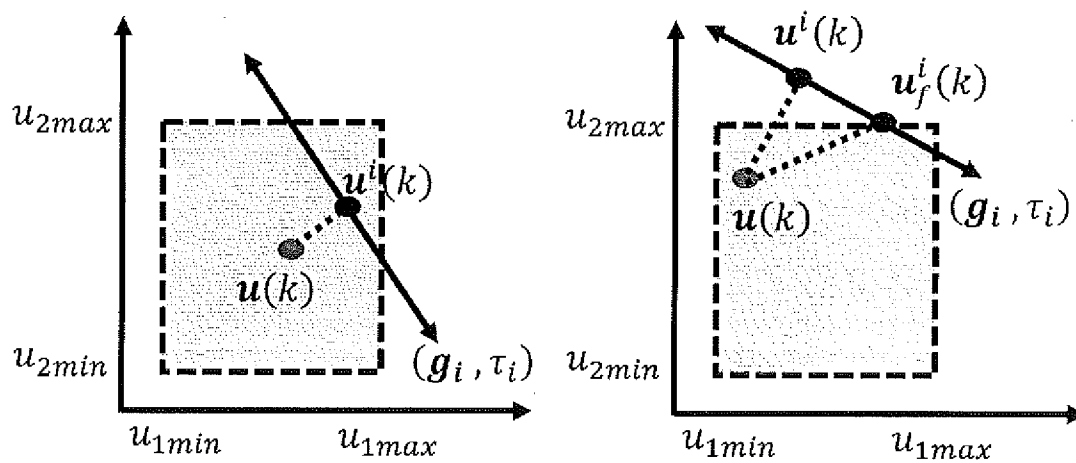
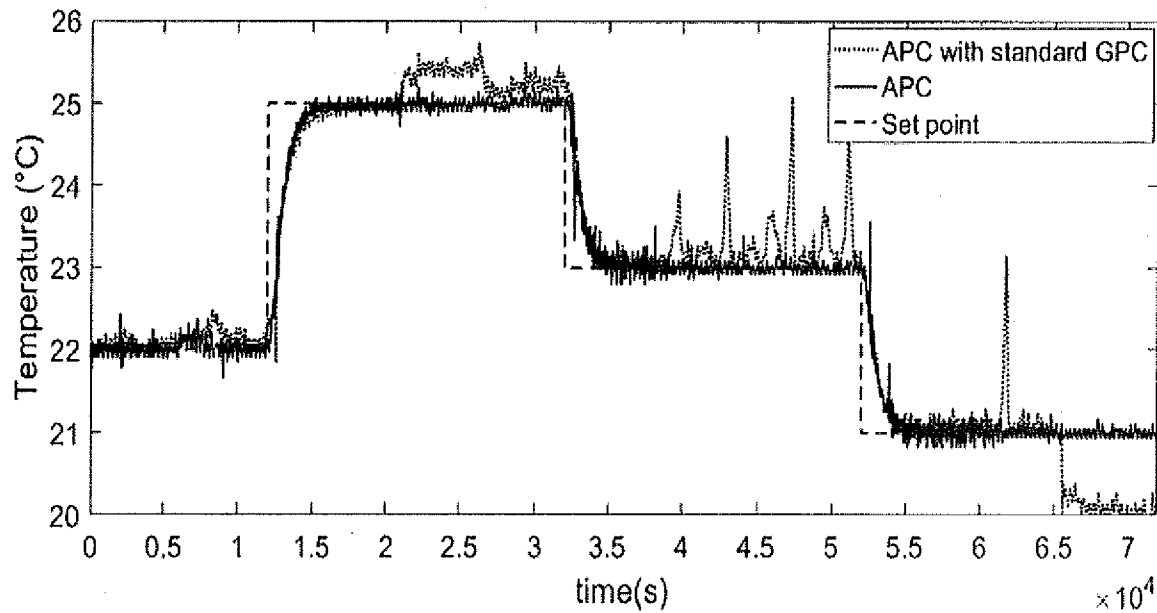
20 the at least one delayed-effect variable signal comprises a processing workload signal representing an indication of a workload that will be assigned to the information technology equipment.

19. The computer program product of claim 13, wherein the at least one delayed-effect variable signal comprises a current signal representing a current drawn by equipment to be cooled.

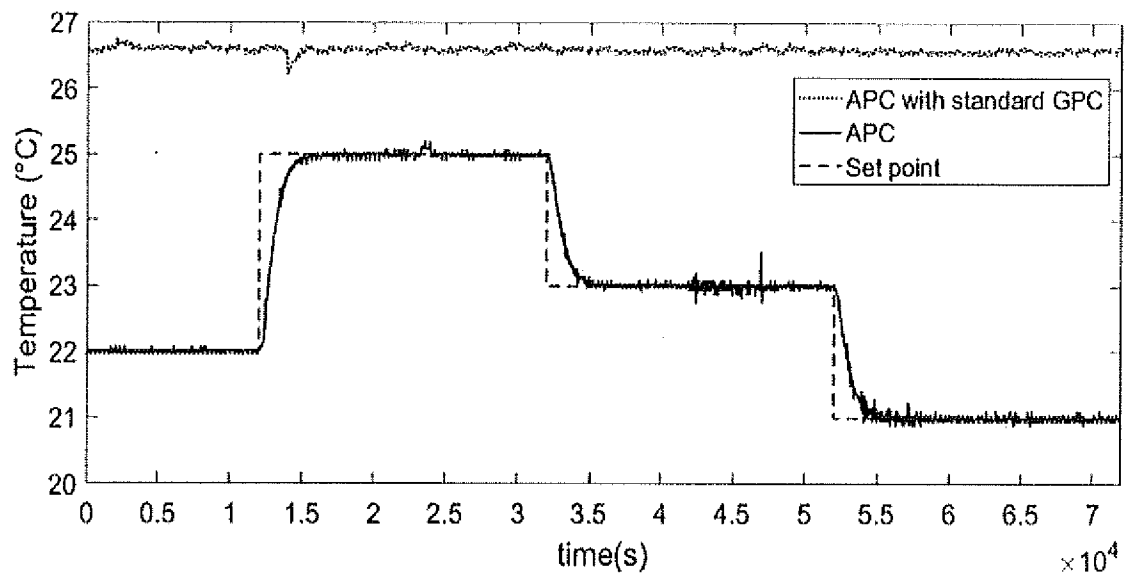
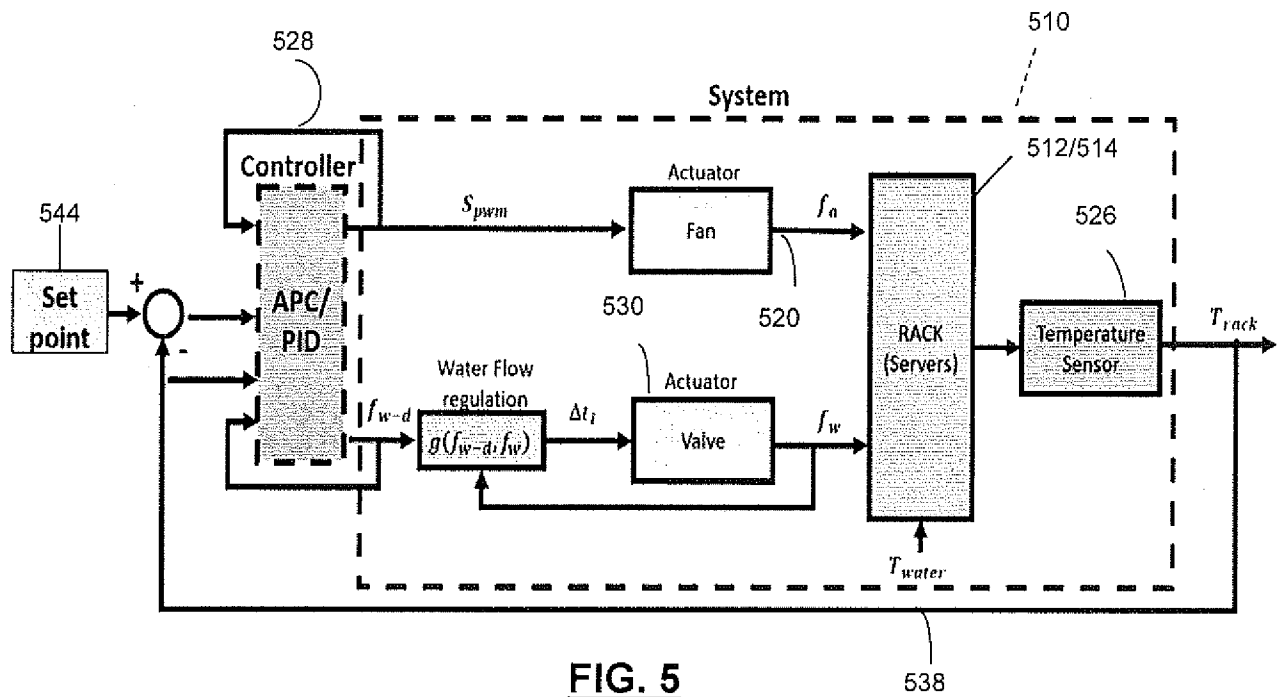
1 / 9

**FIG. 1**

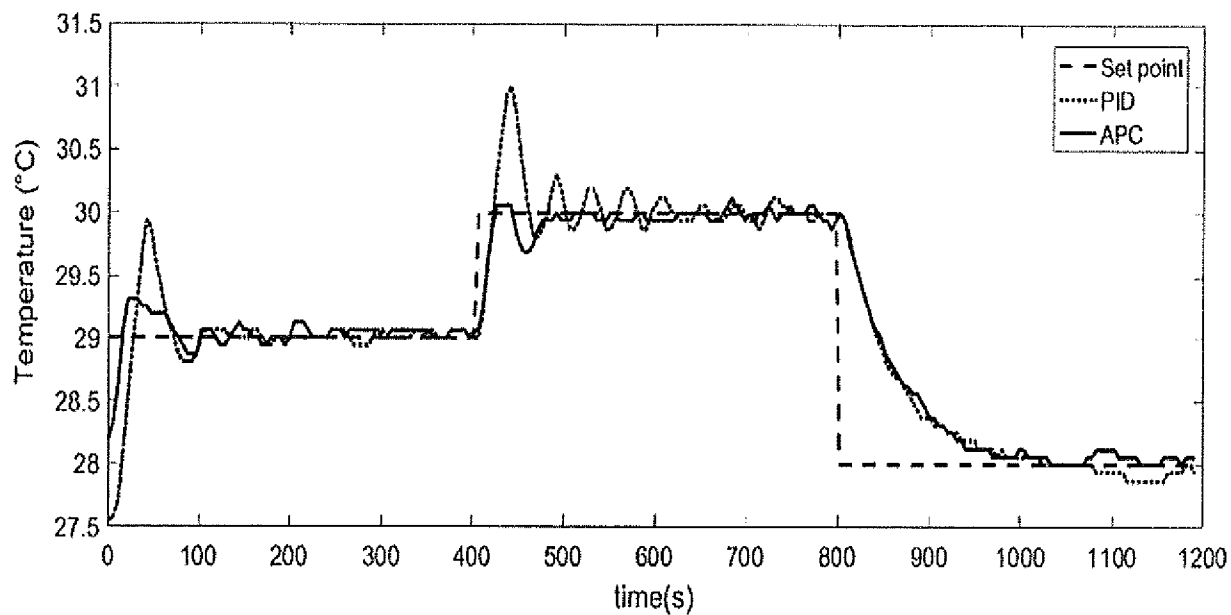
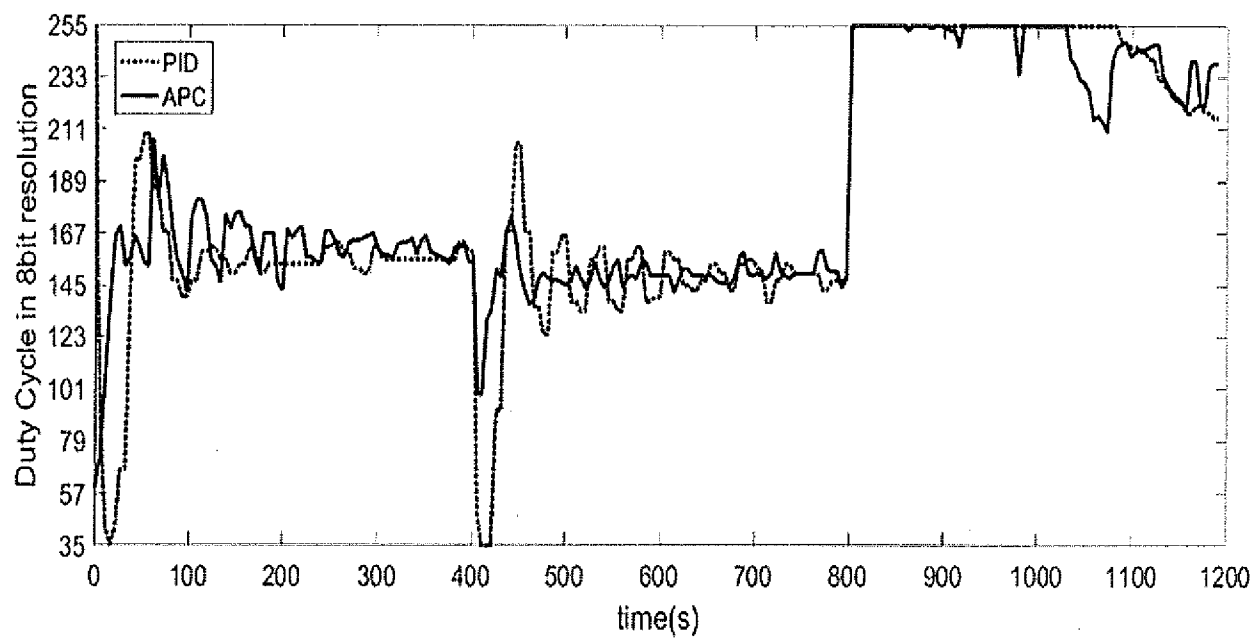
2 / 9

**FIG. 2****FIG. 3**

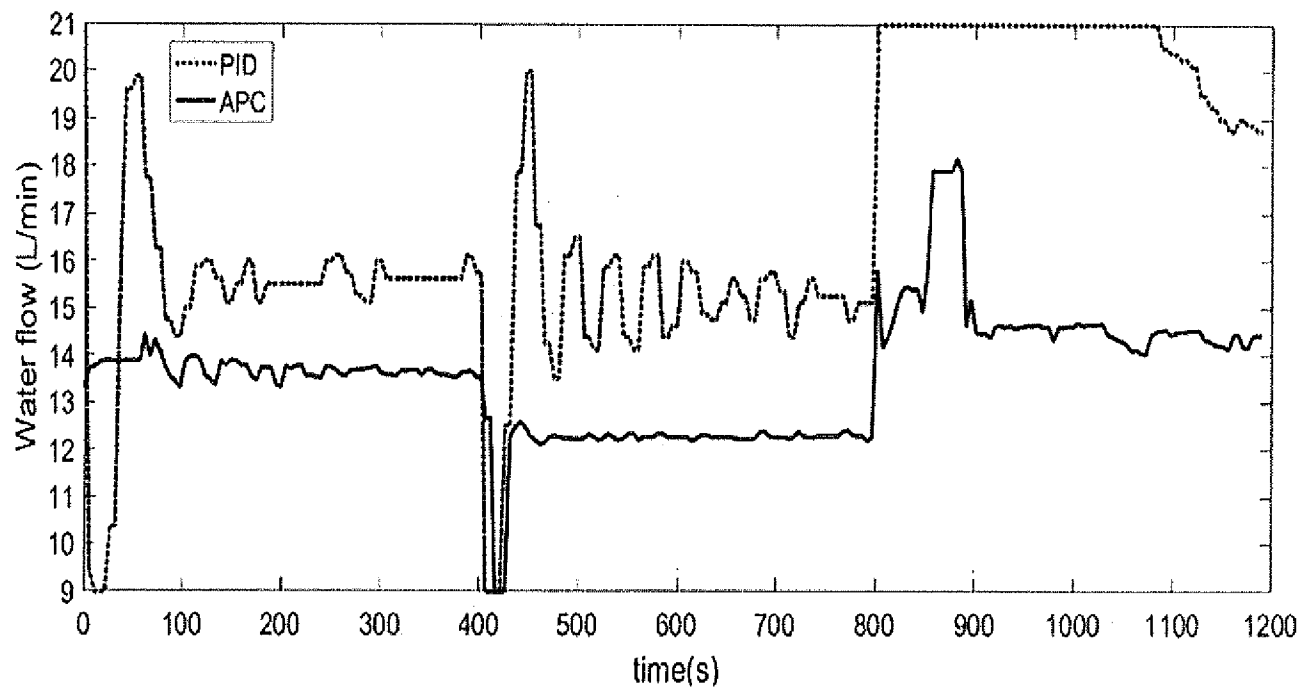
3 / 9

**FIG. 4****FIG. 5**

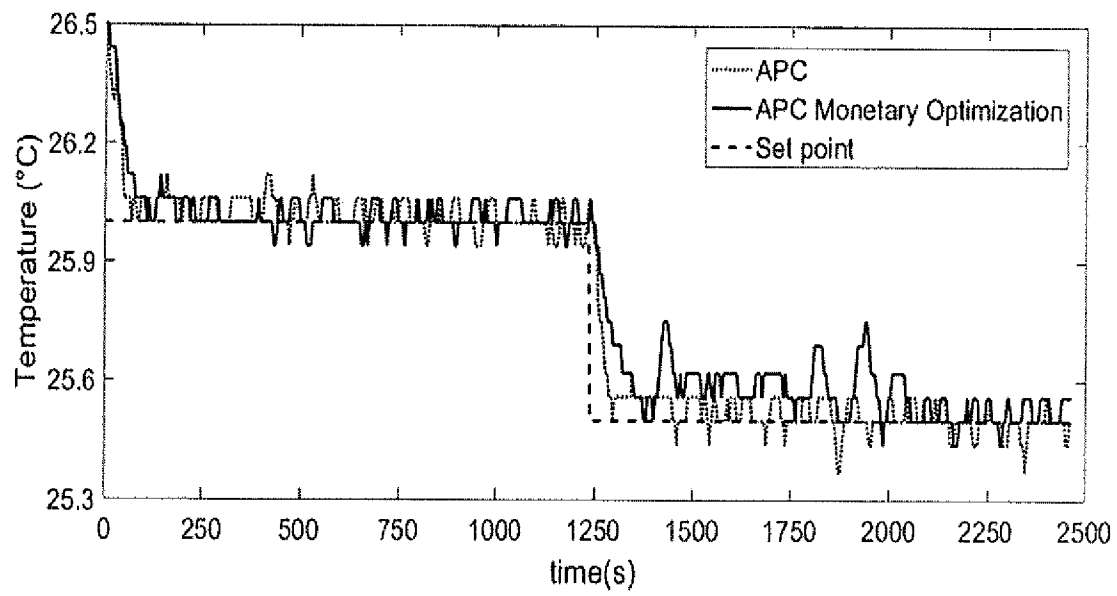
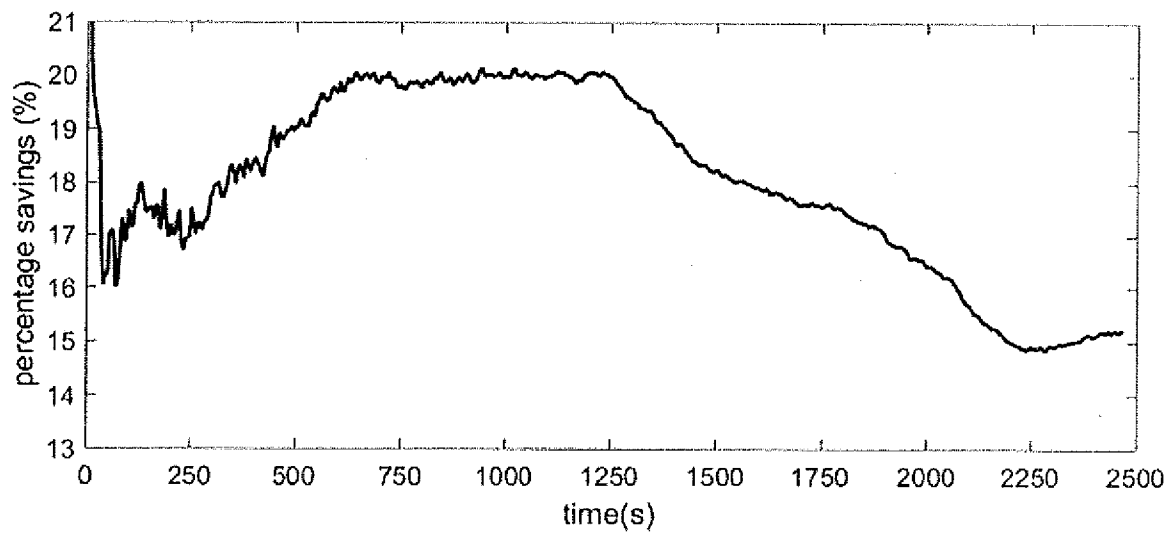
4 / 9

**FIG. 6****FIG. 7**

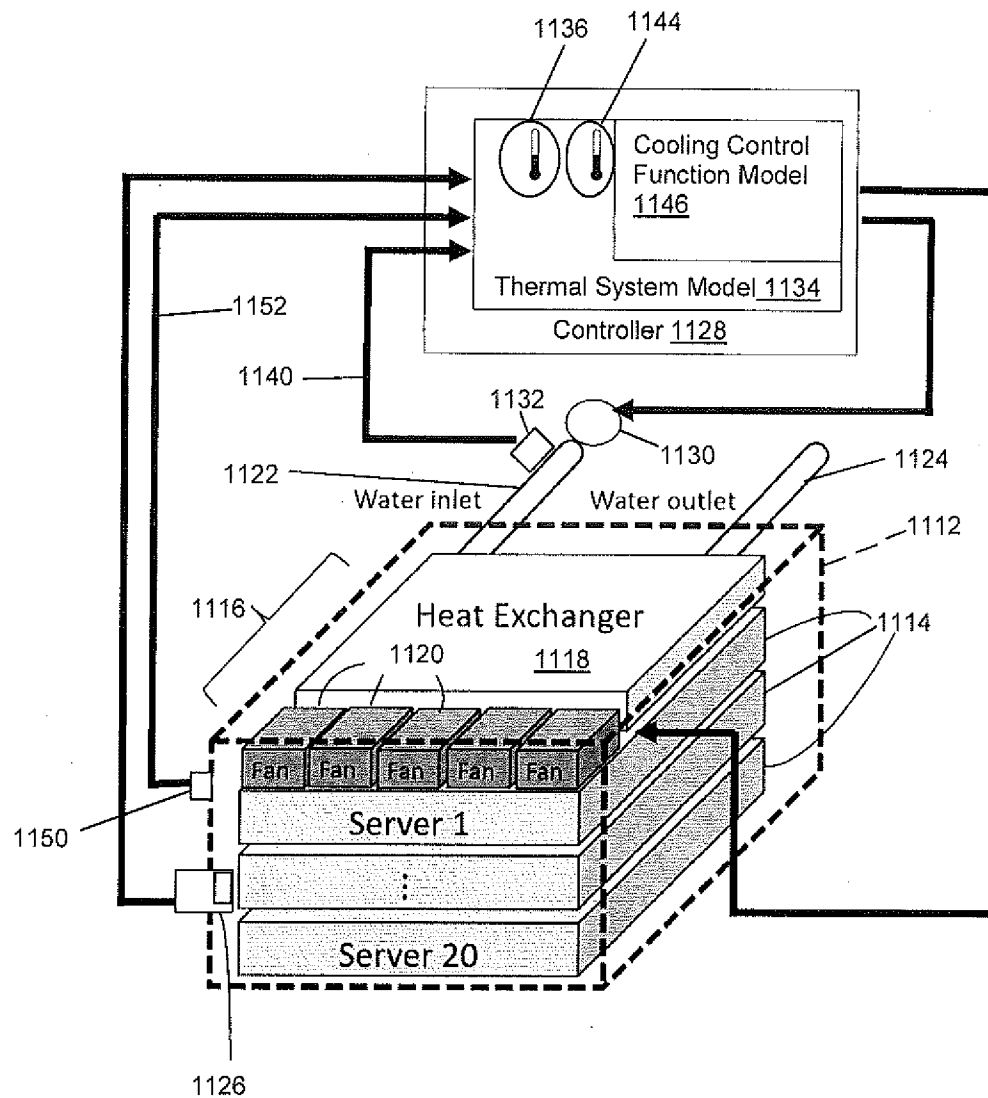
5 / 9

**FIG. 8**

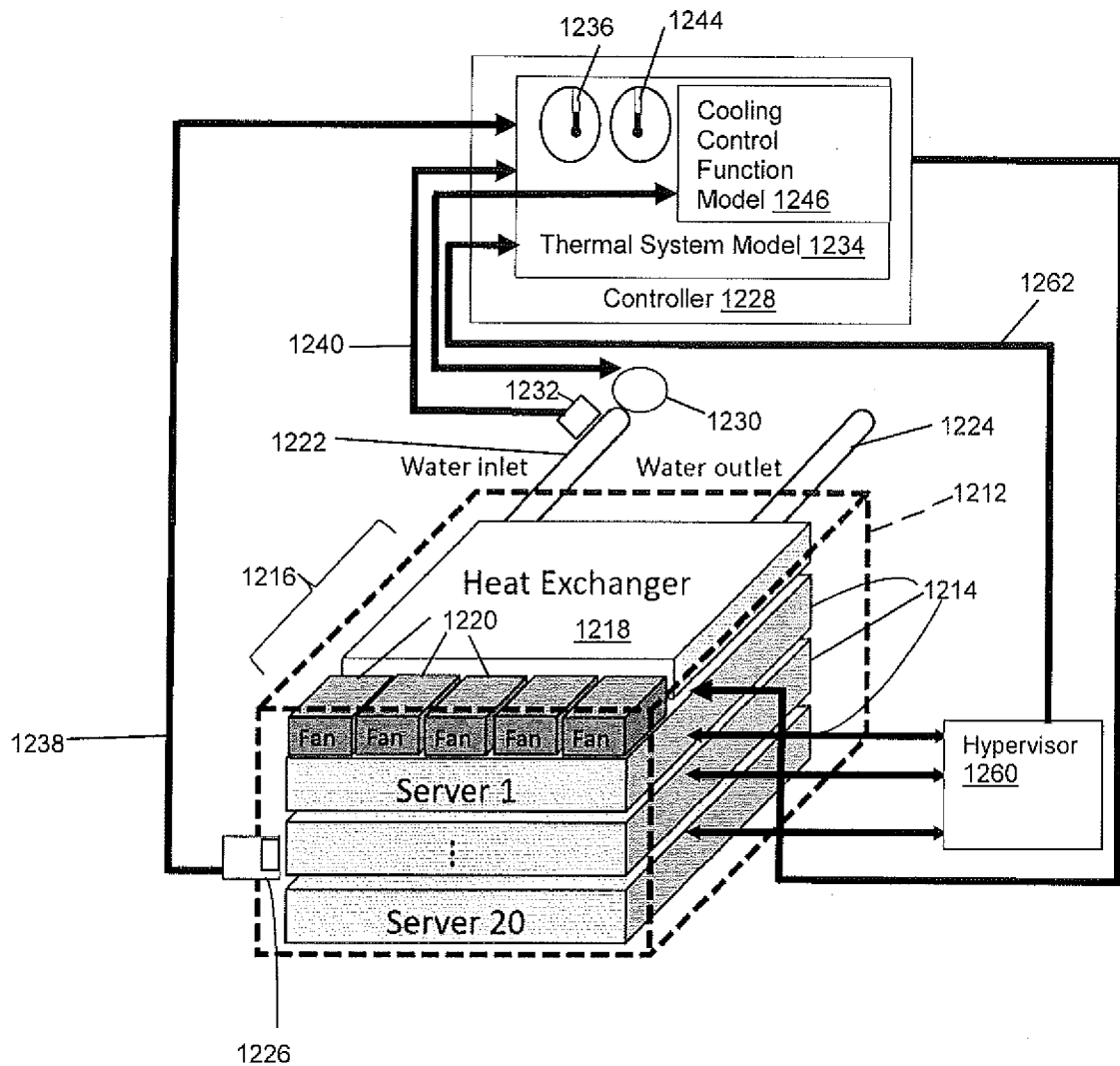
6 / 9

**FIG. 9****FIG. 10**

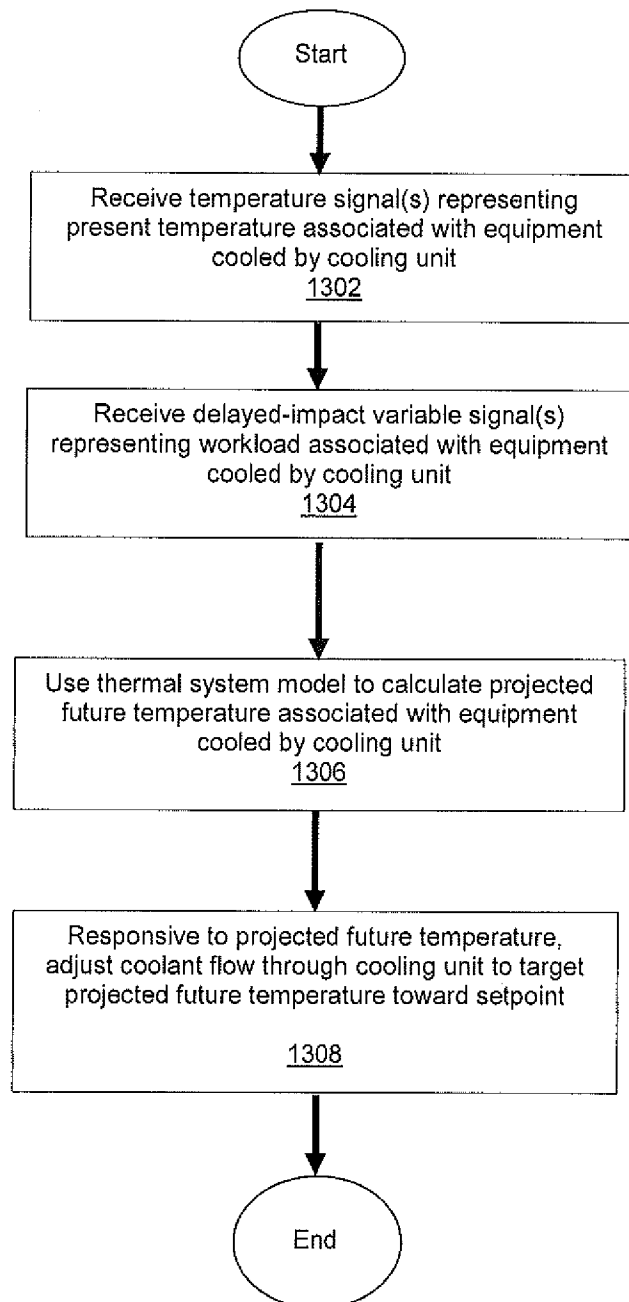
7 / 9

**FIG. 11**

8 / 9

**FIG. 12**

9 / 9

**FIG. 13**

INTERNATIONAL SEARCH REPORT

International application No.

PCT/CA2018/051640

A. CLASSIFICATION OF SUBJECT MATTER

IPC: **F24F 11/63** (2018.01), **F24F 11/70** (2018.01), **F24F 11/83** (2018.01), **F28F 27/00** (2006.01), **G05B 17/02** (2006.01), **G05D 23/19** (2006.01)

According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)

IPC: **F24F 11/63** (2018.01), **F24F 11/70** (2018.01), **F24F 11/83** (2018.01), **F28F 27/00** (2006.01), **G05B 17/02** (2006.01), **G05D 23/19** (2006.01)

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Electronic database(s) consulted during the international search (name of database(s) and, where practicable, search terms used)

Databases: Questel-Orbit, Canadian Patents Database, Google

Keywords: cooling, liquid coolant, gas coolant, control, controller, temperature, input, delayed effect, thermal model, data center, static thermal system model, PID controller, adaptive predictive controller

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
Y	US 2008/245083 A1 (TUTUNOGLU, Ozan et al.), 09 October 2008 (09-10-2008) (pars. 0011, 0012, 0014, 0067, 0075, 0077, 0079, 0080, 0082, 0097, 0098, 0105- 0109, 0112, 0132-0146, 0185 and 0243)	1-19
Y	US 2003/149493 A1 (BLEVINS, Terrence et al.), 07 August 2003 (07-08-2003) (pars. 0003-0005, 0013, 0014, 0021, 0024, 0027-0029, 0045, 0046, 0049, 0055, 0057, figs. 1, 2A, 2B and claims 1, 2, 4, 7, 8, 11, 12, 14, 18, 19, 21-23 and 25)	1-19
Y	WO 2014/055059 A1 (MODI, Yash et al.), 10 April 2014 (10-04-2014) (page 32, line 32 to page 33, line 11)	4 and 16
Y	US 7373268 B1 (VIREDAS, Marc et al.), 13 May 2008 (13-05-2008) (col. 2, line 24 to col. 3, line 6, col. 3, lines 41-62, col. 8, lines 44-53 and fig. 1)	6, 11 and 18

☐ Further documents are listed in the continuation of Box C.

☒ See patent family annex.

* "A" "E" "L" "O" "P"	Special categories of cited documents: document defining the general state of the art which is not considered to be of particular relevance earlier application or patent but published on or after the international filing date document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified) document referring to an oral disclosure, use, exhibition or other means document published prior to the international filing date but later than the priority date claimed	"T" "X" "Y" "&"	later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art document member of the same patent family
--------------------------------------	--	--------------------------	--

Date of the actual completion of the international search
22 February 2019 (22-02-2019)

Date of mailing of the international search report
25 February 2019 (25-02-2019)

Name and mailing address of the ISA/CA
Canadian Intellectual Property Office
Place du Portage I, C114 - 1st Floor, Box PCT
50 Victoria Street
Gatineau, Quebec K1A 0C9
Facsimile No.: 819-953-2476

Authorized officer

Ana Szpaizer (819) 639-8327

INTERNATIONAL SEARCH REPORT
Information on patent family members

International application No.
PCT/CA2018/051640

Patent Document Cited in Search Report	Publication Date	Patent Family Member(s)	Publication Date
US2008245083A1	09 October 2008 (09-10-2008)	US8322155B2 AU2009257922A1 AU2009257922B2 CA2727205A1 CN102112826A CN102112826B CN103175356A CN103175356B EP2313714A1 EP2313714B1 JP2011523026A JP5400147B2 US2008041076A1 US8327656B2 US2013139530A1 US9115916B2 US2010057263A1 US9568206B2 WO2009151841A1 WO2009151841A8	04 December 2012 (04-12-2012) 17 December 2009 (17-12-2009) 10 July 2014 (10-07-2014) 17 December 2009 (17-12-2009) 29 June 2011 (29-06-2011) 27 February 2013 (27-02-2013) 26 June 2013 (26-06-2013) 30 September 2015 (30-09-2015) 27 April 2011 (27-04-2011) 11 July 2018 (11-07-2018) 04 August 2011 (04-08-2011) 29 January 2014 (29-01-2014) 21 February 2008 (21-02-2008) 11 December 2012 (11-12-2012) 06 June 2013 (06-06-2013) 25 August 2015 (25-08-2015) 04 March 2010 (04-03-2010) 14 February 2017 (14-02-2017) 17 December 2009 (17-12-2009) 11 February 2010 (11-02-2010)
US2003149493A1	07 August 2003 (07-08-2003)	US6901300B2 CN1438554A CN1332275C DE10304902A1 DE10304902B4 GB0302813D0 GB2386437A GB2386437B HK1055475A1 HK1057785A1 JP2003295907A JP4194396B2	31 May 2005 (31-05-2005) 27 August 2003 (27-08-2003) 15 August 2007 (15-08-2007) 23 October 2003 (23-10-2003) 13 September 2012 (13-09-2012) 12 March 2003 (12-03-2003) 17 September 2003 (17-09-2003) 09 November 2005 (09-11-2005) 25 January 2008 (25-01-2008) 10 March 2006 (10-03-2006) 17 October 2003 (17-10-2003) 10 December 2008 (10-12-2008)
WO2014055059A1	10 April 2014 (10-04-2014)	CA2885868A1 CN104823119A CN104823119B EP2904462A1 EP2904462A4	10 April 2014 (10-04-2014) 05 August 2015 (05-08-2015) 16 November 2016 (16-11-2016) 12 August 2015 (12-08-2015) 15 June 2016 (15-06-2016)
US7373268B1	13 May 2008 (13-05-2008)	None	