2021

Fault “Auto-correction” for HVAC Systems: A Preliminary Study

Guanjing Lin
Lawrence Berkeley National Lab, United States of America, gjlin@lbl.gov

Marco Pritoni
Lawrence Berkeley National Lab

Yimin Chen
Lawrence Berkeley National Lab

Christopher Weyandt
Lawrence Berkeley National Lab

Raphael Vitti
Lawrence Berkeley National Lab

See next page for additional authors

Lin, Guanjing; Pritoni, Marco; Chen, Yimin; Weyandt, Christopher; Vitti, Raphael; and Granderson, Jessica, "Fault “Auto-correction” for HVAC Systems: A Preliminary Study" (2021). *International High Performance Buildings Conference.* Paper 369.
https://docs.lib.purdue.edu/ihpbc/369

Authors

Guanjing Lin, Marco Pritoni, Yimin Chen, Christopher Weyandt, Raphael Vitti, and Jessica Granderson

Fault "Auto-correction" for HVAC Systems: A Preliminary Study

Guanjing Lin1[1], Marco Pritoni[1], Yimin Chen[1], Christopher Weyandt[1], Raphael Vitti[1], Jessica Granderson[1]*

[1]Lawrence Berkeley National Laboratory,
Berkeley, CA, USA
Contact Information (gjlin@lbl.gov, mpritoni@lbl.gov, yiminchen@lbl.gov, cweyandt@lbl.gov, cweyandt@lbl.gov, JGranderson@ lbl.gov)

* Corresponding Author

ABSTRACT

A Fault Detection and Diagnostics (FDD) tool is a type of energy management and information system that is designed to continuously identify the presence of faults and efficiency improvement opportunities through a 1-way interface to the building automation system and application of automated analytics. It is estimated that 5-30% energy saving can be achieved by employing FDD tools and implementing efficiency measures based on FDD findings. Although the potential of this technology is high, actual savings are only realized when an operator takes an action to fix the problem. There is a subset of faults that can be potentially addressed automatically by the system, without operator intervention. Automating this fault "correction" can significantly increase the savings generated by FDD tools and reduce the reliance on human intervention.

This paper presents preliminary efforts towards delivering automated fault correction. It describes nine fault auto-correction algorithms for heating ventilation and air conditioning (HVAC) systems that were developed to automatically correct faults or improve controls operation. It also presents preliminary testing results of one auto-correction algorithm (improve air handling unit static pressure setpoint reset) in a commercial building, located in Berkeley, California, US. The auto-correction algorithms and implementation frameworks of this initial study provide a foundation for future auto-correction algorithm development and novel schemes for improving building operation performance and reliability.

1. INTRODUCTION

Fault detection and diagnostics (FDD) uses building operational data to identify the presence of faults or efficiency opportunities and isolate their root causes. Over the last three decades the development of FDD methods for building heating ventilation and air conditioning (HVAC) systems has been an area of active research. Well-cited review publications in the HVAC FDD area include two International Energy Agency Annex Reports (Hyvarinen *et al*., 1996, Dexter *et al*., 2001) and literature reviews by Katipamula *et al.* (2005a, 2005b) and Kim *et al*. (2018).

Commercially available FDD tools provide a means of monitoring-based commissioning, through which instances of operational inefficiency can be continuously identified, isolated, and surfaced for resolution by operations and maintenance staff. FDD tools represent one of the fastest growing markets in technologies for building analytics. There are over 30 FDD products available in the US and new software products continue to enter the market (Kramer *et al*., 2020). Today's FDD technology has been documented to enable whole building savings of 9 percent on average, across users (Kramer *et al.*, 2020). Although FDD tools are being used to enable cost-effective energy savings, there is a capability gap in current products. Today's FDD technologies operate in an open loop manner. Faults are identified by the FDD tools, which may also provide a report of the duration and frequency of faults, cost and/or energy impacts, and relative priority levels. However, the identified faults must be corrected through manual human intervention. Building Automation System (BAS) and automated HVAC control optimization technologies offer closed loop supervisory control, but do not provide full-fledged, robust, continuous FDD. In practice, the need for human intervention to fix faults once they are identified often results in delay or inaction, causing additional operations and maintenance costs or deteriorating comfort conditions. This capability gap is not only technical, but also represents market-relevant desired functionality on behalf of FDD users and vendors (Granderson *et al*., 2017). Therefore, this work seeks to develop automated fault correction approaches and integrate them with commercial FDD technology offerings, thereby closing the loop between passive diagnostics and active control. Automating the correction of these types of faults can increase the savings realized through the use of FDD tools, and reduce the extent to which savings are dependent upon human intervention.

Minimal research and development has been performed on self-correcting or fault-tolerant controls for buildings and HVAC systems. Fernandez *et al*. (Fernandez *et al*., 2009a; Fernandez, *et al*., 2009b) and Brambley *et al*. (Brambley *et al*., 2011) developed passive and proactive fault auto-correction algorithms for an air-handler unit (AHU) and a variable-air-volume (VAV) box. The methods are proposed to correct faults occurring in temperature and humidity sensors, dampers, control hunting, and manual overrides. A subset of these algorithms (sensor bias and minimum outdoor air damper position) were implemented and tested in a laboratory experiment. Related to the concept of fault correction is a body of work in the building control literature that focuses on fault tolerant control. The purpose of a fault tolerant controller is to maintain proper operation of a system despite the presence of faults (Zhang and Jiang, 2008). Wang *et al*. (2002) developed a supervisory control scheme that adapts to the presence of a measurement error in outdoor air flow rate. Hao *et al*. (2005) employed principal component analysis to develop fault tolerant control and data recovery in the HVAC monitoring system. Bengea *et al*. (2015) developed a fault-tolerant optimal control strategy for a HVAC system integrating FDD and model predictive control. While the literature focuses on the development of these advanced controllers, it is not yet readily implemented in today's buildings control systems and also does not explore their integration with existing FDD technologies.

This paper presents a preliminary study of automated fault correction. It describes the development of fault auto-correction algorithms that are designed to be integrated with commercial FDD tools. It also presents early testing results of one auto-correction algorithm (improve air handling unit static pressure setpoint reset). In the testing, the algorithm was implemented in a commercial FDD product and successfully changed the static pressure setpoint in two AHUs of a commercial building in Berkeley, California, US.

2. FAULT AUTO-CORRECTION ALGORITHMS

2.1 Fault Auto-correction Algorithms

The primary objective of this study is to develop automated fault correction algorithms that can be integrated with existing commercial FDD and BAS products. Therefore, the developed auto-correction algorithms are decoupled from the FDD algorithms embedded in the existing FDD tools. This permits applicability and feasibility of the developed correction algorithms across a variety of FDD technologies that employ different FDD rules and algorithms. Further, it is assumed that the FDD tools are able to detect the faults of focus, as they represent some of the more commonly encountered faults in commercial buildings.

Nine innovative fault auto-correction algorithms for HVAC system were developed to automatically correct the faults or improve the operation, including: (1) HVAC schedules are incorrectly programmed, (2) revert manual overrides, (3) AHU outdoor air or supply air temperature sensor bias, (4) control hunting, (5) rogue zone, (6) improve economizer high lockout temperature setpoint, (7) improve zone temperature setpoint setback, (8) improve AHU static pressure setpoint reset , and (9) improve AHU supply air temperature setpoint reset (Lin *et al*., 2020). Figure 1 shows a flow chart of the general auto-correction process. In this process, after the FDD algorithm generates a fault flag of a specific fault, the fault auto-correction algorithm is initiated to correct this fault with approval from the building operator. A correctable variable (referred to as Control_variable_being_overwritten), is the key element in the auto-correction process. The algorithm overwrites this variable (Control_variable_being_overwritten_current) via BACnet or other protocol to a new value (Control_variable_being_overwritten_new). The control_variable_being_overwritten_current is the one identified in the FDD algorithm to be associated with the problematic value (fault) or potential to improve (opportunity). The control_variable_being_overwritten_new is the same variable that has the correct value (fault) or optimized value (opportunity). All of the auto-correction algorithms developed in this work follow this structure, with different control variables overwritten in BAS, and different ways to determine the correct or improved value of the variable.

For example, the correctable variable in algorithm "revert manual overrides" is the variable that indicates the equipment manual control status or equivalent flag: 1 – equipment is in manual control, 0 – equipment is in automatic control.

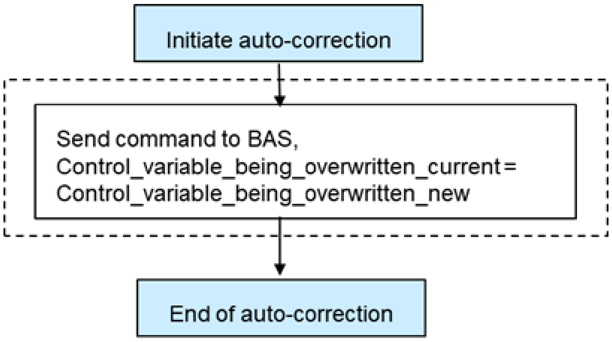

Figure 1: Flow chart of the general auto-correction process

2.2 Auto-correction of AHU Static Pressure Setpoint

The auto-correction algorithm "Improve AHU static pressure setpoint reset" is presented here as an example. The other eight algorithms are described in detail in Lin *et al.* (2020). AHU static pressure reset is one of the top ten efficiency measures implemented by organizations through use of FDD technology (Kramer *et al*., 2020). The auto-correction algorithm for this opportunity is closely related to ASHRAE High-Performance Sequences of Operation Guideline 36 (ASHRAE, 2018), but deployed via the FDD tool. The algorithm corrects the fault "continuously" as it adjusts control variables to optimize equipment operation (e.g., setpoints). It is relevant for AHUs with downstream VAV boxes without sophisticated reset strategies, such as no reset or simple resets based on the average or maximum of downstream VAV damper positions.

The AHU static pressure setpoint is the correctable variable. The auto-correction algorithm uses the ASHRAE High-Performance Sequences of Operation Guideline 36 (ASHRAE, 2018) "Trim and Respond" logic for the new static pressure setpoint. To optimize the operation of the AHU and minimize discomfort, the static pressure setpoint is continually reset using Trim and Respond logic between a minimum and maximum setpoint. When the supply air fan turns on, after a period of inactivity, the setpoint starts from an initial value. The reset logic is active while the supply air fan is proven on. When active, for every time step t, when the sum of pressure requests (R) from the downstream zones is less than or equal to a defined number of ignored requests (I), the setpoint is trimmed by a fixed trim amount. If R is more than I, the setpoint changes by a respond amount based on R, but no more than the maximum response per time interval.

3. PRELIMINARY TESTING

This section illustrates the test results of one auto-correction algorithm: "improve AHU static pressure setpoint reset" (Section 2.2). The algorithm was deployed in a commercial FDD product and tested on two AHUs between March 4th and August 4th, 2020. The goal of this preliminary test was to determine whether the enhanced FDD solution is able to correct the fault without adverse operational effects.

3.1 Description of the Testing Site and Equipment

The test building is a commercial building operated by Lawrence Berkeley National Laboratory (LBNL) and utilized for offices and laboratories. Under normal conditions, the building is operated all week and its lab zones have 24/7 ventilation. Offices are ventilated and conditioned from 4 am to 9 pm. The HVAC system is primarily composed of a water-cooled chilled water plant, a gas-powered heating hot water plant, multiple AHUs and Variable Air Volume (VAV) boxes. AHU01 and AHU02, that serve 90% of the building floor area, and their connected zones (n=83 and n=80) are used for the test of this algorithm. Starting from March 17th the occupancy of the buildings decreased significantly due to COVID-19, but the Labs kept running 24/7. Figure 2 shows the BAS graphics (i.e. native dashboard) for one of the two AHUs.

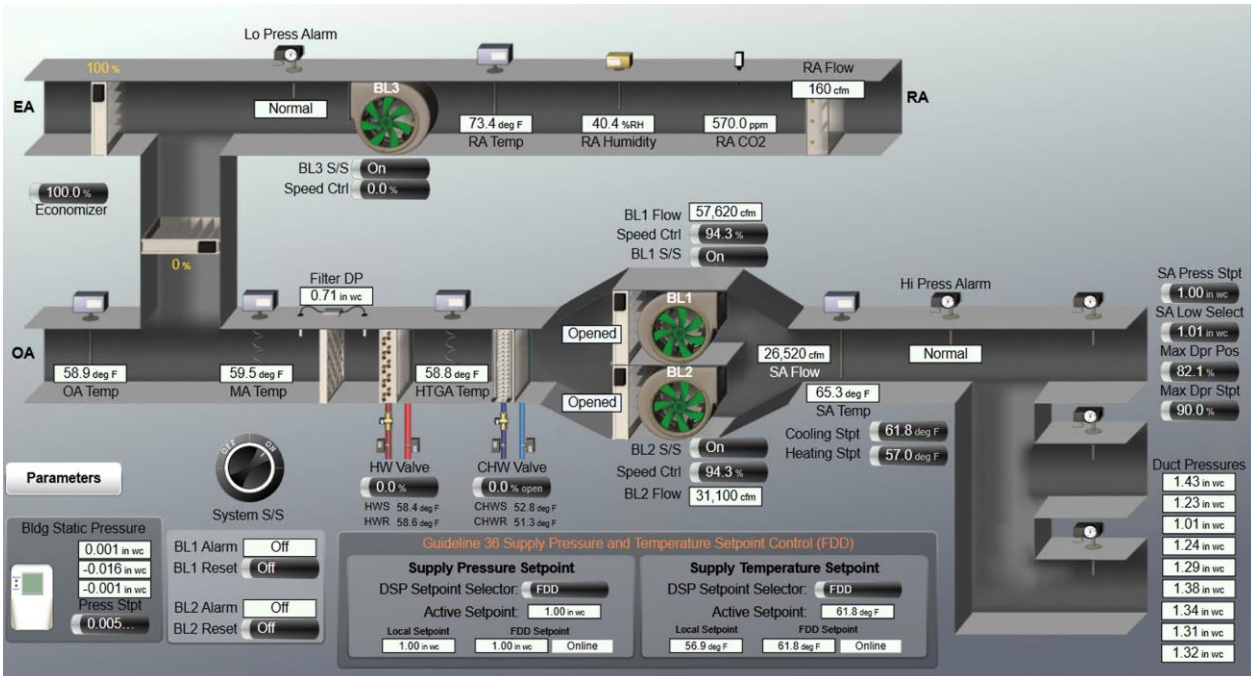

Figure 2: BAS graphics for AHU02 at the LBNL test site

Both AHU01 and 02 are controlled by a "sequence of operation" implemented in the native control language and hosted on local controllers. Each AHU is controlled independently. The baseline Static Pressure Setpoint (SSP_SP) Reset used by AHU01 and AHU02 is coded in the AHU controllers and it is summarized below in English. The supply air fan speed of each AHU is regulated by a direct-acting proportional–integral–derivative loop to keep static pressure at its setpoint, determined by the following reset logic.

Baseline Static Pressure Setpoint Reset Strategy
● If the AHU is enabled (based on schedules, normally 24/7): ○ Calculate the max damper position (MAX_DPR) for all the zones served by an AHU. ○ Use a direct-acting proportional–integral (PI) control loop to reset the Supply Static Pressure Setpoint (SSP_SP) between a minimum and maximum setpoint 249 pa and 373 pa respectively) to keep the max damper position at 90%. The setpoint is reset continuously.

3.2 Implementation of Auto-correction into the FDD Software

The limitation of the baseline strategy is that using the maximum damper position to trigger the reset can easily be negatively impacted by rogue zones (i.e., a stuck damper would cause the SSP_SP to be at its max value the whole time). This opportunity can be addressed by implementation of the correction algorithm "Improve AHU static pressure setpoint reset" into the FDD software. The implementation process includes (1) Change settings in the FDD tool to enable the write capability to the BAS setpoints (2) Program the auto-correction routine in the analytics engine of FDD tool (3) Commission the algorithm, review the auto-correction algorithm outputs, and verify the point value to be auto-corrected.

The FDD tool connected to the BAS is a commercial product managed by the sustainability department of the site. The tool allows for custom programming and bi-directional communication to the BAS via the BACnet network. In contrast to the BAS, the FDD tool coding language is a modern scripting language with the ability to use high-level functions that allow portability of the code between buildings and equipment. The auto-correction algorithm was coded using this platform and tested on the two AHUs. The architecture of the FDD tool and the BAS to support fault auto-correction is presented in Figure 3. The solid line shows the original infrastructure and the red dash line shows the upgrade. Two auto-corrective commands are sent to the BAS BACnet network and then to the writable properties used to control the static pressure. One is the new static pressure setpoint SSP_SP calculated from the correction algorithm (Section 3.3), the other is a “heartbeat” message indicating the FDD tool’s status (online or offline). If the heartbeat is not received by the BAS every 15 minutes (adjustable), the BAS reverts back to the baseline pressure reset control program. On the BAS side, some programming changes are also required. A setpoint variable (highlighted in red and marked with 2 in Figure 4) was added to receive the new SSP_SP coming from the FDD tool and then overwrite the static pressure setpoint calculated by the baseline control logic. A watchdog input (1 in Figure 4) was also added to pair with the heartbeat command from the FDD tool. Finally, an enable input (3 in Figure 4) was added to provide the BAS operator with the ability to switch between the baseline setpoint and the FDD autocorrected setpoint.

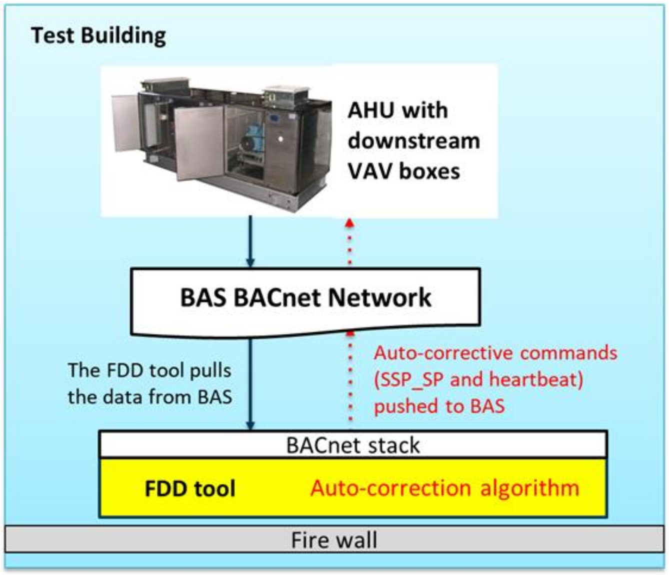

Figure 3: The FDD-BAS architecture to support auto-correction

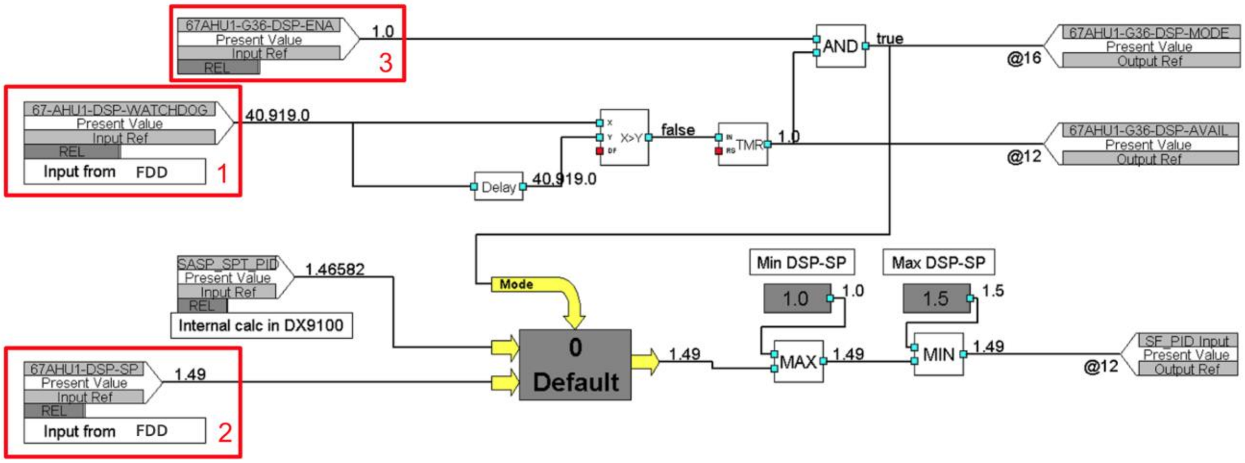

Figure 4: BAS logic that determines what algorithm is in control of the static setpoint. If the enable (3) and the watchdog (1) flag (that checks whether the FDD toolis online) are both TRUE the FDD tool setpoint is used, otherwise the BAS setpoint is used.

3.3 Auto-Correction Code Developed for FDD Tool

The code adopts the algorithm in Section 2.2. The static pressure setpoint (SSP_SP) is continually reset using "Trim and Respond" logic between a minimum and maximum setpoint (SSP_SP_MIN = 249 pa and SSP_SP_MAX =373 pa). When the supply air fan is turned on, the initial setpoint is set to SSP_SP0 = 249 pa and the reset logic is active immediately. When active, every time step t = 5 min, the following algorithm controls the SSP_SP. The advantages of this new algorithm are its simplicity, and the ability to operate efficiently regardless of rogue zones (i.e., these can be handled by increasing the number of ignored zones I)

New SPP Reset Strategy

Variables definition and instantiation:

- Supply Static Pressure Actual: SSP
- Supply Static Pressure Setpoint: SSP_SP
- Min Supply Static Pressure: SSP_MIN = 249 pa
- Max Supply Static Pressure: SSP_MAX = 373 pa
- Initial Supply Static Pressure Setpoint: SSP_SP0 = 249 pa
- VAV Damper Position: DMP
- Requests: R
- Ignores: I =1
- Net Requests: R*

Logic:

- If the AHU is enabled (based on schedules, normally 24/7):
  - For each zone, every 5 min calculate the number of requests based on the VAV damper position (using hysteresis conditions):
    - (trigger condition) if DMP_i > 95% generate a request
    - (release condition) if DMP_i < 85% remove the request
  - Sum the requests R for all the zones
  - Subtract the fixed number of ignored requests I=1
  - Calculate the net number of requests R* = R - I
  - If R* <=0
    - SSP_SP is trimmed by a fixed amount (SSP_trim = 2.5 pa) until the SSP_SP reaches SSP_MIN
  - If R* >0
    - SSP_SP is increased (respond) by an amount proportional to R* (SSP_res = 5 x R* pa with a maximum response of 25 pa) until the SSP_SP reaches SSP_MAX

3.4 Testing Procedure

The algorithm was tested on AHU 01 and AHU 02 between March 4th and August 4th 2020. The test procedure is the following:

- Verify the ability to override all setpoints/parameters to be tested.
- Document pre-implementation behavior.
- Execute the auto-correction routine.
- Observe and document the effect of the automated fault correction

3.5 Testing Results

3.5.1 Baseline reset strategy

An example of the baseline behavior of the SSP_SP reset is displayed in the lower panel (b) of Figure 5. The setpoints profile is smooth because it is continuously reset by a PI loop. The power use of the two fans is depicted in the upper panel (a) of Figure 2. Given the high load due to occupancy and laboratory process load, the pressure setpoint reaches its max between 10-11 am and between 5-6 pm. The fan speed increases over the entire 4 AM - 9 PM period to maintain the static pressure at setpoint with an increased demand from VAV dampers that are only open during the day. Fan speed further increases during the day to meet the increased static pressure setpoint when the demand is even higher, e.g. for meeting the cooling load in some spaces.

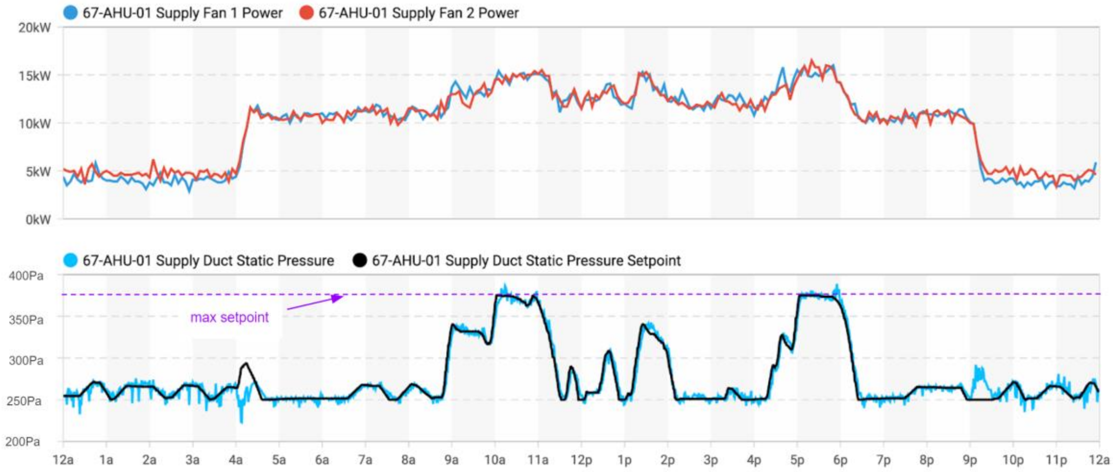

Figure 5. SSP setpoint of AHU01 with the baseline control strategy algorithm (Feb 13, 2020). a) Upper panel: power draw of the two fans controlled using the static pressure; b) Lower panel: SSP_SP using the baseline algorithm (black), actual SSP (light blue).

3.5.2 Test of the Auto-Correction Algorithm

The auto-correction algorithm “Improve AHU static pressure setpoint reset” successfully changed the setpoint of AHU01 and AHU02. Figure 6 shows the successful implementation of the algorithm for one day in the testing period, and the SSP_SP changes followed the routine described in Section 3.3. The total number of requests for all the zones served by AHU01 is displayed in the upper panel of Figure 6. When the net number of requests R* goes above zero the “respond” logic is triggered, increasing the SSP_SP as illustrated in the lower panel of the figure. The first case of this behavior happens around 4am. The actual pressure, SSP follows the setpoints after some oscillations. In this case, the SSP_SP never reaches its max value, but it stops increasing when R* becomes zero again. At that point SSP_SP is constantly “trimmed” until it reaches its minimum SSP_SP_MIN more than an hour later. The same behavior repeats other three times during the same day, although the slope of the “respond” phase is lower, because R* is lower compared to the first case.

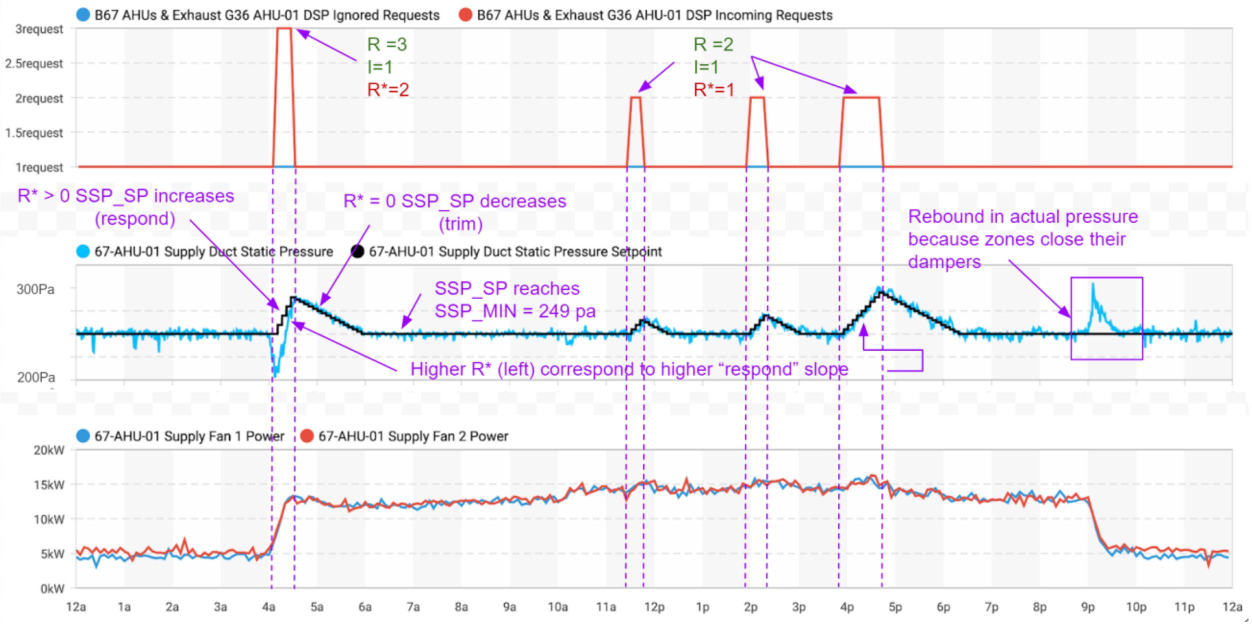

Figure 6: SSP setpoint of AHU01 after the execution of the auto-correction algorithm (Jul 14, 2020). a) Upper panel: number of requests generated by the zones (red). b) Lower panel: SSP_SP using the new algorithm (black) and SSP (light blue).

3.6 Discussion

Since both the baseline logic and the new logic use feedback loops, a quantitative comparison between the two is not possible without modeling the dynamic behavior of the system or collecting enough data to perform a system-level evaluation. However, Figure 7 shows an example of the improved performance of the automated correction routine, in case of a rogue zone. The rogue zone essentially breaks the feedback loop by keeping its damper 100% open regardless of the AHU static pressure. During June 28th 2020, a zone damper was consistently open at 100% (Figure 7, bottom panel). This single zone drove the SSP_SP of the baseline strategy (in red in Figure 7, top panel) to its maximum for the whole day, while the new strategy performed consistently better. The baseline strategy was not in control at the time, but in this situation we know that the baseline strategy setpoint would indeed have stayed at the maximum value of 373 pa.

The auto-correction algorithm has been running for five months without issues and no occupant complaints, although occupancy in the building was significantly reduced due to COVID-19. The building managers have been encouraged by the outcomes of this algorithm. Therefore, they decided to remain the algorithm active in regular operation, because it works better than the original sequence.

Figure 7: Example of 1 rogue zone (lower panel) driving the original SSP_SP setpoint to remain at 373 pa for the entire day (upper panel, in red), while the new strategy generates lower SSP_SP.

The development and deployment of this algorithm stimulated an interesting discussion among the partners and advisors of the project about the role of the FDD and the BAS. Typically, commercial FDD tools are developed as a software layer on top of the existing BAS. There exists a natural separation of roles in this arrangement, in which the BAS actively controls the building and the FDD tool observes its operation and provides insights and recommendations to the building manager. However, some consider FDD tools as a good supplement of BAS that can take over some of its functionalities when it is necessary. In this preliminary testing, implementing the correction algorithm to address the static pressure setpoint opportunity directly in the BAS would be a difficult task. Zone controllers at this testing site cannot directly calculate "requests" necessary for the implementation of the algorithm because both the lab controllers and the office controllers can't easily be modified to run customized logic. Alternatively, calculating requests and rogue zones for each zone in the AHU controller would be overly complex, since each AHU serves more than 80 zones. The existing AHU controllers would likely not have enough computing power to run such calculations. For these reasons we used the FDD tool to implement the algorithm and achieved success.

4. CONCLUSION AND FUTURE WORK

This paper presents preliminary efforts towards delivering automated fault correction. It describes nine fault auto-correction algorithms for heating ventilation and air conditioning (HVAC) systems that were developed to automatically correct faults or improve controls operation. It also presents preliminary testing results of one auto-correction algorithm (improve air handling unit static pressure setpoint reset). In the testing, the algorithm was implemented in a commercial FDD product and successfully changed the static pressure setpoint in two AHUs. The preliminary test confirmed the efficacy of the algorithm, as tested in a real commercial building. It provided a foundation of future auto-correction algorithm development and novel schemes for improving building operation performance and reliability.

In the future, more field testing of the FDD integrated correction algorithms with different FDD tools will be performed in a cohort of commercial buildings. This will include evaluation of the technical efficacy and performance of each correction routine, evaluation of the operations and maintenance benefits for each site in cohort, and characterization of challenges and best practices. A second area of future work will entail design and execution of a techno-economic analysis to quantify the broader market opportunity to inform ongoing commercialization efforts.

REFERENCES

American Society of Heating Refrigerating and Air Conditioning Engineers(ASHRAE) Guideline 36. (2018). High Performance Sequences of Operation for HVAC Systems.

Bengea, S. C., Li, P., Sarkar, S., Vichik, S., Adetola, V., Kang, K., Lovett, T., Leonardi, F., Kelman, A. D. (2015). Fault-tolerant optimal control of a building HVAC system. Science and Technology for the Built Environment, 21(6), 734-751.

Brambley, M. R., Fernandez, N., Wang, W., Cort, K. A., Cho, H., Ngo, H., & Goddard, J. K. (2011). *Final project report: Self-correcting controls for VAV system faults Filter/Fan/Coil and VAV box sections* (No. PNNL-20452). Pacific Northwest National Lab.(PNNL), Richland, WA (United States).

Dexter, A., & Pakanen, J. (Eds.). (2001). *Demonstrating automated fault detection and diagnosis methods in real buildings*. Technical Research Centre of Finland (VTT).

Fernandez, N., Brambley, M. R., & Katipamula, S. (2009). *Self-correcting hvac controls: Algorithms for sensors and dampers in air-handling units* (No. PNNL-19104). Pacific Northwest National Lab.(PNNL), Richland, WA (United States).

Fernandez, N., Brambley, M. R., Katipamula, S., Cho, H., Goddard, J. K., & Dinh, L. H. (2010). *Self-Correcting HVAC Controls Project Final Report* (No. PNNL-19074). Pacific Northwest National Lab.(PNNL), Richland, WA (United States).

Granderson, J., Singla, R., Mayhorn, E., Ehrlich, P., Vrabie, D., & Frank, S. (2017). Characterization and survey of automated fault detection and diagnostics tools. *Report Number LBNL-2001075 (Lawrence Berkeley National Laboratory, 2017)*.

Hao, X., Zhang, G., & Chen, Y. (2005). Fault-tolerant control and data recovery in HVAC monitoring system. *Energy and buildings*, *37*(2), 175-180.

Hyvarinen, J., & Karki, S. (1996). Building optimization and fault diagnosis source book. *IEA Annex*, *25*(1996.8).

Katipamula, S., & Brambley, M. R. (2005). Methods for fault detection, diagnostics, and prognostics for building systems—a review, part I. *HVAC&R Research*, *11*(1), 3-25.

Katipamula, S., & Brambley, M. R. (2005). Methods for fault detection, diagnostics, and prognostics for building systems—A review, part II. *HVAC&R Research*, *11*(2), 169-187.

Granderson, J., Kramer, H., Lin, G., Curtin, C., & Crowe, E. (2020). Proving the business case for building analytics. Lawrence Berkeley National Laboratory, October 2020. https://doi.org/10.20357/B7G022

Kim, W., & Katipamula, S. (2018). A review of fault detection and diagnostics methods for building systems. *Science and Technology for the Built Environment*, *24*(1), 3-21.

Lin, G., Pritoni, M., Chen, Y., & Granderson, J. (2020). Development and Implementation of Fault-Correction Algorithms in Fault Detection and Diagnostics Tools. *Energies*, *13*(10), 2598.

Wang, S., & Chen, Y. (2002). Fault-tolerant control for outdoor ventilation air flow rate in buildings based on neural network. *Building and Environment*, *37*(7), 691-704.

Zhang, Y., & Jiang, J. (2008). Bibliographical review on reconfigurable fault-tolerant control systems. *Annual reviews in control*, *32*(2), 229-252.

ACKNOWLEDGEMENT

This work was supported by the Assistant Secretary for Energy Efficiency and Renewable Energy, Building Technologies Office, of the U.S. Department of Energy under Contract No. DE-AC02-05CH11231. The authors wish to acknowledge Harry Bergmann for his guidance and support of the research. We also thank the fault detection and diagnostics technology and service providers who participated in this study.