Nonlinear controller design methods for modern building heating and ventilation systems using inverse dynamics

John M. Counsell, Joseph Brindley, Gavin B. Murphy and Yousaf Khalid

University of Strathclyde, Glasgow, Scotland, UK

This paper presents the control of a HVAC system using a nonlinear, multivariable controller design based on a Nonlinear Dynamic Inverse method. The proposed benefits of this design compared to a traditional PID controller are shown through the improved decoupling of the interactions between the heating and ventilation systems, better disturbance rejection and tighter setpoint tracking for humidity and temperature and reduced over-driving of actuator systems resulting in less energy usage. The robustness of the controller when faced with varying degrees of sensor lag is investigated and a method of reducing this sensitivity by dynamically varying the controller’s response is developed using a nonlinear error actuated controller. A model of a HVAC controlled indoor zone is presented and the proposed controller’s performance is compared to that of a PI design. The simulation results demonstrate the improved comfort and energy saving potential of the proposed advanced controller design.

# Introduction

Heating Ventilation and Air-conditioning (HVAC) systems require a control system to regulate the temperature, humidity or CO2 levels of an indoor environment. The performance of the control systems has a large impact on the overall performance of the HVAC system in terms of air quality and energy efficiency (1). A poorly designed control system may be difficult to tune, result in overshoot of the setpoint or behave in an unstable an oscillatory manner – all of which can contribute to excessive energy usage, as well as reduced occupant comfort (2).

Traditionally, controllers with relatively simple algorithms, such as Proportional Integrator Derivative (PID), are used for HVAC control, this is in spite of the complexity of the control problem. One of the most common problems in HVAC control is interaction between controlled variables (3). Heating and ventilation systems are strongly coupled, in that the action of one has an effect on the other. Hence, the simultaneous control of temperature and humidity can be difficult to achieve without the two systems fighting each other and resulting in inefficient energy usage. Furthermore, HVAC systems are subject to many disturbances such as rapidly changing free heats (heat from people and appliances) and external weather conditions as well as the nonlinear thermodynamics of the internal environment. The dynamics of HVAC systems themselves also present a challenge to controller design: heating and ventilation systems have limits on the amount of heat or air flow they are able to provide which means that a poorly design control system can overdrive the actuation systems, resulting in wasted energy.

 Advanced nonlinear dynamic inverse control algorithms have been developed to deliver high performance control for highly interactive systems and have been successfully applied in various fields, such as the aerospace, automotive and process control industries (4) (5). The acceptance of these nonlinear dynamic inverse (NDI) types of controller design has been slow in the field of HVAC control, partially due to the often complex nature of the design process and the practicalities of their implementation. This paper presents the design of an advanced controller that is relatively simple to implement, practical and robust.

The algorithms are based on the principle of nonlinear inverse dynamics and directly address the problems of interaction between controlled channels, nonlinear system dynamics and actuator limitations. Particular attention is paid to the performance of the controller with undesirable operating conditions where sensor measurements are slow to respond. The Variable Transient Response (VTR) control algorithm(6) that ensures that the sensitivity of the control system’s performance to sluggish sensors is minimized whilst still providing high performance control is described.

Simulations of HVAC system control for a modern office space are performed using a simplified multi-input multi-output thermodynamic model of the buildings physics and systems. The simulations compare the performance of a PI controller to an advanced NDI controller in tracking air temperature and relative humidity. The sensitivity of the performance of the system using the advanced NDI controller to sensor measurement delays are simulated and the performance of the VTR algorithm in reducing sensitivity is studied.

# Building Energy System Mathematical Model

In order to test the controller designs a building thermodynamic model with particular requirements for controller design must be established. The advanced controller design methods of RIDE and VTR require that the system to be controlled can be represented in a relatively low order state space form. A high order, extremely detailed energy simulation could be used to assess performance but would not be suitable for controller design, where stability must be investigated. Therefore, a simplified, low order model that still captures the essential dynamic properties of the building thermodynamics is required(1)(7). The resulting system is based on the models developed by Khalid and Murphy (3)(8)but with the addition of a ventilation system and relative humidity model.

Thermodynamics

There are four thermodynamic states in the building model; air temperature, $T\_{a}$, internal structure temperature, $T\_{si}$, external structure tempertaure, $T\_{se}$ and internal thermal mass temperature, $T\_{tm}$. The model assumes a single indoor air zone that is fully mixed and at constant pressure and density.

The energy and mass balance equations that govern the rates of change of temperature in the zone are

 $M\_{a}C\_{a}\dot{T\_{a}}\left(t\right)=\dot{Q}\_{h}\left(t\right)+\dot{Q}\_{free}\left(t\right)-\dot{Q}\_{si}\left(t\right)-\dot{Q}\_{f}\left(t\right)-\dot{Q}\_{r}\left(t\right)-\dot{Q}\_{w}\left(t\right)-\dot{Q}\_{nv}\left(t\right)-\dot{Q}\_{mv}\left(t\right)-\dot{Q}\_{tm}\left(t\right)$ (1)

 $M\_{si}C\_{si}\dot{T}\_{si}\left(t\right)=\dot{Q}\_{si}\left(t\right)-\dot{Q}\_{wall}(t)$ (2)

 $M\_{se}C\_{se}\dot{T}\_{se}\left(t\right)=\dot{Q}\_{wall}\left(t\right)-\dot{Q}\_{se}\left(t\right)$ (3)

 $M\_{tm}C\_{tm}\dot{T}\_{tm}\left(t\right)=\dot{Q}\_{tm}\left(t\right)$ (4)

The roof, floor and glazing are assumed to be at constant temperature and as such there is negligible energy stored in them. The ground temperature is also assumed to be constant. The heat losses through the floor windows and roof are given by the following

 $\dot{Q}\_{f}\left(t\right)=U\_{f}A\_{f}\left(T\_{a}\left(t\right)-T\_{g}\right)$ (5)

 $\dot{Q}\_{r}\left(t\right)=U\_{r}A\_{r}\left(T\_{a}\left(t\right)-T\_{ex}\left(t\right)\right)$ (6)

 $\dot{Q}\_{w}\left(t\right)=U\_{w}A\_{w}\left(T\_{a}\left(t\right)-T\_{ex}\left(t\right)\right)$ (7)

The heat losses due to ventilation are expressed in equations 8 and 9. The mass flow rate due to natural ventilation is assumed constant.

 $\dot{Q}\_{nv}\left(t\right)=\dot{m}\_{nv}C\_{a}\left(T\_{a}\left(t\right)-T\_{ex}\left(t\right)\right)$ (8)

 $\dot{Q}\_{mv}\left(t\right)=\dot{m}\_{mv}(t)C\_{a}\left(T\_{a}\left(t\right)-T\_{ex}(t)\right)$ (9)

Heat is lost to thermal mass due to the temperature difference between the zone air and the thermal mass.

 $\dot{Q}\_{tm}\left(t\right)=U\_{tm}A\_{tm}\left(T\_{a}\left(t\right)-T\_{tm}\left(t\right)\right)$ (10)

It is assumed that the walls of the zone are composite and are comprised of an internal surface, the wall structure and an external surface. Equation 11 represents the heat exchange between the zone air and the internal surface of the structure, Equation 12 describes the heat exchange through the wall between the internal and external surfaces and Equation 13 describes the heat exchange between the external surface and the outside air.

 $\dot{Q}\_{si}\left(t\right)=h\_{i}A\_{s}\left(T\_{a}\left(t\right)-T\_{si}\left(t\right)\right)$ (11)

 $\dot{Q}\_{wall}\left(t\right)=\frac{k}{wt}A\_{s}\left(T\_{si}\left(t\right)-T\_{se}\left(t\right)\right)$ (12)

 $\dot{Q}\_{se}\left(t\right)=h\_{e}A\_{s}\left(T\_{se}\left(t\right)-T\_{ex}\left(t\right)\right)$ (13)

The rate of change absolute humidity of the zone air is modeled as the difference between the moisture added and removed from the zone. In terms of occupant comfort relative humidity is a more practical measure of the quality of the internal air. An empirical relation between the absolute and relative humidity is derived from data from the Psychrometric chart (9). This relation is taken from an operating point of 23°C and an absolute humidity of 0.007 *kg/kg*.

 $M\_{a}\dot{W}\_{a}\left(t\right)=\dot{m}\_{mv}\left(t\right)\left(W\_{ex}-W\_{a}\left(t\right)\right)+\dot{m}\_{nv}\left(t\right)\left(W\_{ex}-W\_{a}\left(t\right)\right)+S\_{occ}$ (14)

 $\dot{W}\_{ar}\left(t\right)=5000\dot{W}\_{a}\left(t\right)-1.388\dot{T}\_{a}(t)$ (15)

Actuation Systems

The controllable inputs into the zone are the heat source, $\dot{Q}\_{h}(t)$, and the mechanical ventilation mass flow rate, $\dot{m}\_{mv}$. These inputs are not able to deliver an instantaneous or unlimited amount of heat or fresh air due to the limits of their actuation systems. These actuator limitations are extremely important in controller design as they govern the maximum achievable controller performance, irrespective of the controller design. It is the controller design that determines how close actual performance is to the maximum achievable.

 The actuation systems in this study are approximated as having first order dynamics and amplitude limits. The first order dynamics have a time constant which represents the inertia present in the actuation system between the commanded controller output and the actual output achieved by the actuator(2). Equation (16) describes a simplified relationship for the heat transfer rate for a typical actuator system and can be used to represent many heat delivery systems, such as convectors, radiators, underfloor heating and even storage heaters where the controller demands a certain power from the heater. The amplitude limits represent the power limitations of the heating system. The ventilation system is more straightforward as the mass flow rate is directly proportional to the fan speed, thus the time constant is an approximation of the delay caused by the power limitations of the fan motor and the fan inertia. The amplitude limits of the ventilation system are approximate and directly proportional to the maximum fan speed. Values of the heating and ventilation time constants and amplitude limitations are indicative of a commercial HVAC system with a direct convector and are displayed in section V.

 $\ddot{Q}\_{h}\left(t\right)=\frac{1}{τh}\left(uc\_{Qh}(t)-\dot{Q}\_{h}(t)\right)$ (16)

 $LL\_{Qh}\leq \dot{Q}\_{h}(t)\leq UL\_{Qh}$ (17)

 $\ddot{m}\_{mv}\left(t\right)=\frac{1}{τv}\left(uc\_{mv}\left(t\right)-\dot{m}\_{mv}\left(t\right)\right)$ (18)

 $LL\_{mv}\leq \dot{m}\_{v}(t)\leq UL\_{mv}$ (19)

State Space Representation

Modern controller design methods require that the system to be controlled be represented in a non-linear generalized state space form(10).

 $\dot{x}\left(t\right)=Ax\left(t\right)+Bu\left(t\right)+Dd\left(t\right)$ (20)

 $y\left(t\right)=Cx\left(t\right)$ (21)

Where $x(t)$ is the state vector, $u(t)$ is the input vector, $d(t)$ is the disturbance vector and $y\left(t\right)$ is the output to be controlled. In this study the controller will be required to simultaneously track internal air temperature, $T\_{a}$ and relative humidity, $W\_{ar}$ by varying the power of the heat source, $\dot{Q}\_{h}(t)$ and the mechanical ventilation rate, $\dot{m}\_{mv}(t)$. The inputs into the system that are not controllable are the free heats and the external temperature. Therefore, in this case, the state space vectors are as follows

 $x(t)=\left[\begin{matrix}T\_{a}(t)\\T\_{si}(t)\\T\_{se}(t)\\T\_{tm}(t)\\W\_{a}(t)\\W\_{ar}(t)\end{matrix}\right]$ (22)

 $u\left(t\right)=\left[\begin{matrix}\dot{Q}\_{h}(t)\\\dot{m}\_{mv}(t)\end{matrix}\right]$ (23)

 $d\left(t\right)=\left[\begin{matrix}\dot{Q}\_{free}(t)\\T\_{ex}(t)\\S\_{occ}\\W\_{ex}\\T\_{g}\end{matrix}\right]$ (24)

 $y\left(t\right)=\left[\begin{matrix}T\_{a}(t)\\W\_{ar}(t)\end{matrix}\right]$ (25)

It can be seen from Equations 1 to 15 that the *B* matrix is a function of the state and disturbance vectors. Therefore, a generalized nonlinear form of state space representation is more appropriate and will enable a nonlinear controller design to be used.

 $\dot{x}\left(t\right)=Ax\left(t\right)+B\left(x\left(t\right),d\left(t\right)\right)u\left(t\right)+Dd\left(t\right)$ (26)

Equation 27 shows the zone thermodynamics represented in state space form. The matrixes $A$, $B\left(x\left(t\right),d\left(t\right)\right)$, $C$ and $D$are fully described in section V and the appendix.

 $\left[\dot{\begin{matrix}T\_{a}\left(t\right)\\T\_{si}\left(t\right)\\T\_{se}\left(t\right)\\T\_{tm}\left(t\right)\\W\_{a}\left(t\right)\\W\_{ar}\left(t\right)\end{matrix}}\right]=\left[\begin{matrix}a\_{11}&a\_{12}&0&a\_{14}&0&0\\a\_{21}&a\_{22}&a\_{23}&0&0&0\\0&a\_{32}&a\_{33}&0&0&0\\a\_{41}&0&0&a\_{44}&0&0\\0&0&0&0&a\_{55}&0\\a\_{61}&a\_{62}&0&a\_{64}&a\_{65}&0\end{matrix}\right]\left[\begin{matrix}T\_{a}\left(t\right)\\T\_{si}\left(t\right)\\T\_{se}\left(t\right)\\T\_{tm}\left(t\right)\\W\_{a}\left(t\right)\\W\_{ar}\left(t\right)\end{matrix}\right]+\left[\begin{matrix}b\_{11}&b\_{12}\left(x,d\right)\\0&0\\0&0\\0&0\\0&b\_{52}\left(x,d\right)\\b\_{61}&b\_{62}\left(x,d\right)\end{matrix}\right]\left[\begin{matrix}\dot{Q}\_{h}\left(t\right)\\\dot{m}\_{mv}\left(t\right)\end{matrix}\right]+\left[\begin{matrix}d\_{11}&d\_{12}&0&0&d\_{15}\\0&0&0&0&0\\0&d\_{32}&0&0&0\\0&0&0&0&0\\0&0&d\_{53}&d\_{54}&0\\d\_{61}&d\_{62}&d\_{63}&d\_{64}&d\_{65}\end{matrix}\right]\left[\begin{matrix}\dot{Q}\_{free}\left(t\right)\\T\_{ex}\left(t\right)\\S\_{occ}\\W\_{ex}\\T\_{g}\end{matrix}\right]$ (27)

1. Tested Controllers

## Proportional Integral (PI) Controller

The Proportional Integral or PI controller is one of the most commonly used controller designs for HVAC systems. Occasionally a third, derivative term is added to the resulting in a PID controller. For the purposes of this study only the PI controller is used.

 The control algorithm for a PI controller is as follows

 $u\_{C}\left(t\right)=KIr\left(t\right)+K\_{P}e(t)$ (28)

 $e\left(t\right)=yc-y(t)$ (29)

 $\dot{r}(t)=K\_{I}e(t)$ (30)

The PI controller contains two tunable parameters; $KI$and $KP$. These parameters are adjusted until desired closed loop transient properties are obtained. The PI controller is a linear Single Input Single Output (SISO) design. This means that only one variable can be controlled at any one time, therefore, for the case of controlling temperature and humidity two completely separate PI controllers need to be designed – one for controlling each variable. If there is interaction between the controlled variables then this will be detrimental to the controller performance as interaction will manifest as a disturbance to affected controller.

Robust Inverse Dynamics Estimation (RIDE)

The basic principal of a nonlinear dynamical inverse (NDI) controller is that, by assuming that the actuator and sensor systems have reached a quasi-steady state condition then using an inverse of the system to be controlled explicitly in the controller design, the system’s dynamics will be cancelled out leaving the control system designer free to specify the desired closed loop dynamics. By cancelling out the dynamics any coupling between control channels, disturbances and nonlinear behavior which would otherwise cause control problems are removed(11).

This is somewhat impractical as the full dynamics of the system, including any disturbances, need to be known. A more robust and practical approach to NDI controller design can be achieved by using the Robust Inverse Dynamics (RIDE) controller design method(12). RIDE uses an estimate of the system inverse (for which less information about the system is required) inside of a traditional feedback control structure. The feedback controller assigns closed loop dynamics and removes uncertainty from the estimate(13).

If the plants dynamics including a quasi-steady state models for sensor and actuators are expressed as in equation 26then the control to set $\dot{y}(t)=0$, ueq(t), is given by

 $u\_{eq}\left(t\right)=-CB\left(x,d\right)^{-1}\left(CAx\left(t\right)+CDd\left(t\right)\right)$ (31)

This nonlinear dynamic inverse, known as the equivalent control, removes any interaction between controlled channels, nonlinear plant dynamics and disturbances. However, this control could be impossible to realise in practice as it requires full knowledge of the plant dynamics as well as full state and disturbance feedback. By rearranging Equations 21 and 26 and substituting into 30 the following is produced

 $\hat{u\_{eq}}(t)=-\left(CB\left(x,d\right)\right)^{-1}\dot{y}(t)+u(t)$ (32)

This form of the equivalent control is much more practical is it only requires knowledge of the $B$ matrix at the expense of extra feedback.Perfect feedback is not possible, thus this is an estimate of the equivalent control described in Equation 30.

$\hat{u\_{eq}}$ can then be implemented in a feedback control structure which will remove uncertainty in the estimate and assign closed loop dynamics.

 $u\_{C}\left(t\right)=r\left(t\right)-K\_{P}w\left(t\right)+\hat{u\_{eq}}$(t) (33)

 $\dot{r}(t)=K\_{I}e(t)$ (34)

**Figure 1. RIDE control structure.**



When implementing the RIDE controller with a near perfect estimate of the equivalent control the closed loop dynamics are reduced to the following transfer function

 $G\_{r(s)}=\frac{K\_{I}CB\_{trim}}{s^{2}+\left(K\_{P}CB\_{trim}\right)s+K\_{I}CB\_{trim}}$ (35)

Where $CB\_{trim}$is the nonlinear $CB(x,d)$ matrix trimmed at an operating point about which a small perturbation in the state vector, *x*, will take place. In the case this is slowly varying relating to *x(t)* then it is the value of CB at point x and *d.* Comparing this to a transfer function representing generalized second order dynamics it can be noted that the gains *KI* and *KP* can be calculated for a designed system natural frequency and damping ratio.

 $G\_{(S)}=\frac{ω\_{s}^{2}}{s^{2}+\left(2ζ\_{s}ω\_{s}\right)s+ω\_{s}^{2}}$ (36)

 $K\_{P}=g[CB\_{trim}]^{-1}$ (37)

 $K\_{I}=p[CB\_{trim}]^{-1}$ (38)

 $g=2ζ\_{s}ω\_{s}$ (39)

 $p=ω\_{s}^{2}$ (40)

An important proviso as well as *CB* being slowly varying is that the transmission zeros of the system using the feedback vector should have all eigen values in the left half of the complex plane. If this is the case, then the slow modes of the NDI closed loop poles will coincide with the locations of the transmission zeros as well as at the locations specified by the second order dynamics in the characteristic equation of X, ensuring that the system is stable for all values of *g* and *p*.

Variable Transient Response

The gain, *p*, of a control system dictates how fast the controlled output will be able to reach its setpoint. A larger gain will result in a more responsive controller that is able to reach a setpoint in a faster time. However, the gain cannot be increased without limit - if the gain is too large then undesirable overshoot of the setpoint or oscillatory behavior can develop resulting from actuator inertia and sensor measurement delays. The limit at which this occurs is greatly influenced by parasitic dynamics e.g. the time constants of the actuators and sensors or any lags within the feedback system. Large time delays may mean that a responsive controller is not possible without incurring severe oscillatory behavior. Thus, a controller that is designed to be responsive may not be robust when implemented due to the presence of sensor lags and large actuator or sensor time constants. In order to reduce the controller’s sensitivity to unmodelled fast dynamics it is necessary to reduce the controller’s gain, but unfortunately this could reduce the controller’s responsiveness. If the unmodelled dynamics are slow enough compared to the closed loop bandwidth then the gain may have to be set to a value which would result in an unacceptable speed of response. Therefore, reducing sensitivity to parasitic dynamics comes at the expense of speed of response. However, this penalty can be overcome if different response characteristics are prioritized at different points during the controllers operation.

When the error is large, i.e the output is far away from the setpoint, the priority is to reach the setpoint in an acceptable time. During this period responsivity can be prioritized over robustness as this period should be brief and any oscillatory behavior will be short lived. When the output approaches the setpoint the reverse is true as the priority is robust and stable performance. What is being described by these statements is a transient change in system damping inversely proportional to the error and a directly proportional change in system natural frequency. If this can be achieved then a fast response time to the setpoint can be achieved whilst simultaneously using a lower gain and speed of response at the setpoint to reduces sensitivity of the performance to unmodelled fast dynamics.

 Relationships between natural frequency and error, and damping and error can be proposed that meet these requirements, where *X* and *Y* are tuneable coefficient matrixes and *K* is a function of the plant.

 $ω\_{VTR} target \left(e\left(t\right)\right)=K\_{ω}\left(Xe\left(t\right)^{2}+Y\right)$ (41)

 $ζ\_{VTR} target \left(e\left(t\right)\right)=K\_{ζ}\left(Xe\left(t\right)^{2}+Y\right)^{-1}$ (42)

 $X=\left[\begin{matrix}X\_{i}&0&0&0\\0&X\_{i+1}&0&0\\0&0&\ddots &0\\0&0&0&X\_{n}\end{matrix}\right]$ (43)

 $Y=\left[\begin{matrix}Y\_{i}&0&0&0\\0&Y\_{i+1}&0&0\\0&0&\ddots &0\\0&0&0&Y\_{n}\end{matrix}\right]$ (44)



**Figure 2. Proposed transient changes in natural frequency and damping with error.**

Considering the RIDE control system, this relationship can be achieved if a nonlinear gain matrix is placed before the integrator in the forward path of the outer loop, as shown in Figure 2. The regulator is now

 $\dot{r}\left(t\right)=N\left(e\left(t\right)\right)K\_{I}e\left(t\right)$ (45)

and the total feedforward gain is given by

 $K\_{ff}\left(e\left(t\right)\right)=N\left(e\left(t\right)\right)K\_{I}$ (46)



**Figure 3. Placement of the VTR gain within the RIDE control structure**

Since the equivalent control allows assignable second order closed loop dynamics the natural frequency and damping ratio are given by the following relations where *CB* trim is a trimmed value of *CB*:

 $ω\_{VTR}\left(e\left(t\right)\right)=\sqrt{CB\_{trim}K\_{I}N\left(e\left(t\right)\right)}$ (47)

 $ζ\_{VTR}\left(e\left(t\right)\right)=K\_{P}CB\_{trim}\left(2\sqrt{K\_{I}CB\_{trim}N\left(e\left(t\right)\right)}\right)^{-1}$ (48)

 $N\left(e\left(t\right)\right)=\left(Xe\left(t\right)^{2}+Y\right)^{2}$ (49)

It can be seen from these expressions that if the VTR controller is tuned for a given setpoint then, keeping the same controller parameters, if the setpoint is raised the response will be underdamped and, if lowered, the response will be overdamped. This can be corrected either by finding a compromise by tuning *X* and *Y* so that the system is not over-damped for small setpoints or under-damped for large setpoints, or by simply scheduling *X* and *Y* with setpoint.

In summary, the following controller design procedure is suggested:

1. Design RIDE control system to achieve desired closed loop natural frequency and damping with gains *KI* and *KP*. If the desired closed loop dynamics are not obtainable due to overshoot or oscillatory dynamics a VTR design is required.
2. Retune the RIDE controller with a reduced integral gain until oscillations or overshoot are no longer present, this can be known as $KI\_{crit}$.
3. Set the VTR tuning parameter so that $Y=\sqrt{\frac{KIcrit}{KI}}$. This will result in the steady state VTR integral gain being equal to $KI\_{crit}$.
4. Tune the value of *X* so that the speed of response is increased to the desired level.
5. Actuator Saturation

 Actuator saturation can cause extreme performance degradation of the control system if its effects have not been considered in the controller design(14). If the actuator saturates and the control output is allowed to build then the actuator can remain on its limit for longer than would be necessary to achieve the desired closed loop performance. This can lead to excessive amounts of energy being used. Therefore, a method of limiting the control signal so that it does not exceed the actuator limits is required. Simply placing a static limit after the controller output would not be effective as this would simply replace the actuator as the saturation element. A dynamic method of limiting the control output is therefore necessary. The RIDE control algorithm can be modified to include a Variable Structure Control (VSC) design that will dynamically prevent the controller output from exceeding specified limits(6).

 Each channel of the regulator should be switched in accordance with the following commutation law

 $\dot{r}\_{i}=\left\{\begin{matrix} 0 if ε\_{ui}<0 or ε\_{li}>0 \\KI(yc\_{i}-w\_{i}) if ε\_{ui}>0 or ε\_{li}<0\end{matrix}\right.$ (50)

 $εu\_{i}=UL\_{i}-uc\_{i}$ (51)

 $εl\_{i}=LL\_{i}-uc\_{i}$ (52)

Providing that the steady state is reachable, i.e. there is enough actuator amplitude available to bring the output to a steady state, then the control limits will not be exceeded.

1. HVAC System Simulation

## Case Study

In this study an HVAC system providing heat and ventilation to a modern office space is simulated. The office space is a single zone with a floor area of 12m by 12m and a ceiling height of 4m. The external temperature and solar gains are for Glasgow, Scotland in February and as such the zone requires nearly constant heating during the office hours. Setpoints of 21°C zone air temperature and 50% relative humidity are demanded between 7am and 7pm, out of these hours the HVAC system is switched off. The control system varies the rate of heat output by the heater and the mass flow rate of the ventilation system in order to achieve this. Out of office hours the control HVAC system is switched off.

Large open spaces or poorly positioned sensors can lead to a sensor lag developing in the control system. This is occurs when there is a physical separation between the area to be controlled and the sensor measuring the controlled variable, resulting in a pure time delay in the closed loop system. This time delay can cause instability and may require the controller’s responsiveness to be reduced. The simulations will investigate each controller design’s sensitivity and robustness with varying degrees of sensor lag.

**Table 1: Case study parameters**

Ma = 691 kg Msi = 7000 kg Mse = 7000 kg Mtm = 8000 kg

As = 179.5 m2 Af = 144 m2  Ar = 144 m2 Aw = 51 m2 Atm = 138 m2

Uf = 0.2 W/m2K Ur = 0.13 W/m2K Uw = 1.5 W/m2K Utm = 2 W/m2K

k = 0.2 W/mK wt = 0.4 m he = 0.12 W/m2K hi = 0.11 W/m2K

LLQh = 0 W ULQh = 6000 W LLmv = 0 kg/s ULmv = 0.35 kg/s τh = 120 s τv = 60 s

τ = 0.003 seconds

ωa = 350 rad/s

*dr* = 0.7

Large motor deflection limit = 0.3 rad

Small motor deflection limit = 0.3 rad

Large motor rate limit = 17.5 rad/s

Small motor rate limit = 4.0 rad/s

τ = 0.003 seconds

ωa = 350 rad/s

*dr* = 0.7

Large motor deflection limit = 0.3 rad

Small motor deflection limit = 0.3 rad

Large motor rate limit = 17.5 rad/s

Small motor rate limit = 4.0 rad/s

τ = 0.003 seconds

ωa = 350 rad/s

*dr* = 0.7

Large motor deflection limit = 0.3 rad

Small motor deflection limit = 0.3 rad

Large motor rate limit = 17.5 rad/s

Small motor rate limit = 4.0 rad/s

Controller Design

Traditional Single Input Single Output Design (PID)

The PID controller design is single input single output (SISO), meaning that it can only control one variable at any one time. Therefore, it is necessary to design two individual control loops; one for temperature control and another for relative humidity control. The major drawback of this is that any coupling between the heating and ventilation systems is not accounted for explicitly in the controller designs and will consequently lead to performance degradation. The gains were tuned as to provide the best performance without serious instability occurring in the controller.

Common to both control loops is the need for some form of integrator anti-windup. Saturation of the actuators leads to the integrator term rapidly building (“winding up”) whilst the controller is not able to provide any more useful output. This leads to the control signal increasing in magnitude causing the actuator to remain “stuck” on its limit, often resulting in overshoot or limit cycles. The PI controller is implemented without any form of anti-windup compensation so that the effects of its absence can be observed.

 As the controller is switched off at the end of each working day it is necessary to ensure that the controller is properly initialized when it is switched back on. When the controller is switched on the output should be zero, therefore, the regulator term, $z(t)$, must be initialized so this that is the case. Thus, during initialization

 $z\left(t\right)=-\frac{KP}{KI}e\left(t\right)$ (54)

**Multivariable Inverse Dynamics Design (RIDE Controller)**

The first stage in the design of the RIDE control system is to establish the $CB$ matrix. Since the outputs to be controlled are $T\_{a}$ and $W\_{ar}$the$C$ matrix is

 $C=\left[\begin{matrix}1&0&0&0&0&0&0\\0&0&0&0&0&0&1\end{matrix}\right]$ (55)

Referring to the appendix the $B$ matrix is

 $B=\left[\begin{matrix}\frac{1}{M\_{a}C\_{a}}&-\frac{1}{M\_{a}}\left(T\_{a}\left(t\right)-T\_{ex}\left(t\right)\right)\\0&0\\0&0\\0&0\\0&0\\0&\frac{1}{M\_{a}}\left(W\_{ex}-W\_{a}\left(t\right)\right)\\-\frac{1.388}{M\_{a}C\_{a}}&\frac{1.388}{M\_{a}}\left(T\_{a}\left(t\right)-T\_{ex}\left(t\right)\right)+\frac{5000}{M\_{a}}\left(W\_{ex}-W\_{a}\left(t\right)\right)\end{matrix}\right]$ (56)

Therefore

 $CB=\left[\begin{matrix}\frac{1}{M\_{a}C\_{a}}&-\frac{1}{M\_{a}}\left(T\_{a}\left(t\right)-T\_{ex}\left(t\right)\right)\\-\frac{1.388}{M\_{a}C\_{a}}&\frac{1.388}{M\_{a}}\left(T\_{a}\left(t\right)-T\_{ex}\left(t\right)\right)+\frac{5000}{M\_{a}}\left(W\_{ex}-W\_{a}\left(t\right)\right)\end{matrix}\right]$ (57)

The equivalent control vector and controller gain matrixes can now be constructed from equation 57. This is a remarkable result as in order to implement a multivariable inverse dynamics controller *the only information about the structure required are the mass and specific heat capacity of the air*. The sensory requirements are simply the external and internal air temperatures, the internal relative humidity and internal and external absolute humidity, which are all practically obtainable. From this it can be seen that an advanced HVAC controller design can be relatively simple to implement and would only require a few extra sensors compared to a traditional PI design.

 The RIDE controller also needs to be initialized at startup, as described in the PI design. Therefore, upon starting the controller the regulator is set to the following:

 $z\left(t\right)=\left[KI\right]^{-1}\left(KPw\left(t\right)-u\_{eq}\left(t\right)\right)$ (58)

In order to prevent the controller output from exceeding the actuator limits the VSC design as described in section X is implemented with the regulator being switched as described by the logic in Equation 50. The controller limits *LL1, LL2, UL1*and *UL2* are set equal to the lower and upper heating power and ventilation mass flow rate limits respectively.

**VTR Controller De**s**ign**

The first stage in the VTR controller design is to select the parameter *Y* which will determine the steady state (when the setpoint is being tracked) gain of the controller. This parameter should be selected so that the gain is low enough not to cause any oscillations or unstable behavior caused by a high gain. A value of 0.55 was selected for *Y* to provide a good degree of robustness.

The second stage is to select the parameter *X* which will determine how the gain changes with error – the larger *X* the stronger the effect of the dynamic gain change. A value of 0.033 was used so that the speed of response was equivalent to that of the RIDE controller.

Simulation Results

The simulations were performed by numerically integrating at a time step of 8 seconds over the heating months. Results are presented over a 100 hour period from days 80 to 84. The first set of simulation results are with a small sensor lag of 2 minutes, comparing the closed loop performance of PI and RIDE controllers. The second set of results are with a large sensor lag of 7 minutes and compare the performance of PI, RIDE and RIDE/VTR.



**Figure 4. Zone Air temperature and heating power for RIDE controller with 2 min. sensor lag**



**Figure 5. Zone Air temperature and heating power for PI controller with 2 min. sensor lag**



**Figure 6. Zone relative humidity and ventialtion flow rate for PI controller with 2 min. sensor lag**



**Figure 7. Zone relative humidity and ventialtion flow rate for RIDE controller with 2 min. sensor lag**

The first set of simulation results demonstrates a significant performance difference between the PI and RIDE controllers. The PI controller displays a consistent overshoot of the temperature setpoint, averaging 4 degrees. The heater reaches its power limit on each day with the control signal exceeding the limit each time. The lack of anti-windup protection in the PI controller is apparent. This causes the control signal to continue building even after the limit has been reached, resulting in the heater remaining on its limit for longer than is necessary, a temperature overshoot and needless energy usage. The tracking of the setpoint is reasonably good, however, some oscillations are observed. The RIDE controller displays no overshoot at all due to the control signal immediately returning to the limit once it has been exceeding and so no control signal buildup occurs. The consequence of this is that the heater remains on the limit for less time compared to the PI controller and thus using less energy during this period. This does not come at the expense of fast response performance as the RIDE and PI controllers reach the setpoint at the same time. Tracking of the setpoint is perfect as the inverse dynamics present in the RIDE algorithm are more effective than the PI controller at rejecting disturbances and coupling from the ventilation system. The humidity results for the controllers also demonstrate a difference in performance. The PI controller is unable to consistently track the setpoint of 50% relative humidity as there are steady state errors consistently present. This may be due to the fact that the PI controller is only single channel and so is unequipped to respond to the interactions between the ventilation and heating systems. Conversely, the RIDE controller is “aware” of the interactions due to its multivariable control law and is thus better able to track the humidity setpoint.



**Figure 8. Zone Air temperature and heating power for PI controller with 7 min. sensor lag**



**Figure 9. Zone Air temperature and heating power for RIDE controller with 7 min. sensor lag**



**Figure 10. Zone Air temperature and heating power for RIDE/VTR controller with 7 min. sensor lag**



**Figure 11. Zone relative humidity and ventialtion flow rate for PI controller with 7 min. sensor lag**



**Figure 12. Zone relative humidity and ventialtion flow rate for RIDE controller with 7 min. sensor lag**



**Figure 13. Zone relative humidity and ventialtion flow rate for RIDE/VTR controller with 7 min. sensor lag**

 The second sets of results are with a large sensor lag of 7 minutes. It can be seen that the performance difference between the PI and RIDE controllers is now much less significant as the addition of the lag has resulted in both controllers displaying considerable oscillations and overshoot. It is clear that any benefit gained by using an advanced control algorithm disappears if the sensory feedback is inaccurate or delayed and the controller is not robust enough to negate this.

 The simulation results of the HVAC system controlled by a RIDE with VTR algorithm are much improved compared to just the RIDE algorithm alone. It can be seen that the addition of VTR greatly improves the controller’s robustness whilst maintaining the speed of response to the setpoint. Any oscillatory behavior is restricted to a brief period during the rise to the setpoint and is of low amplitude. As the setpoint is approached the gain is decreased which results in much more robust and stable performance whilst tracking the setpoint.

 The VTR algorithm is particularly suited to an NDI based controller design due to the NDI controller not requiring a high gain at steady state in order to reject disturbances. Traditional designs use a high gain combined with feedback to reduce the sensitivity of the controller to disturbances and provide better tracking of the setpoint. NDI based designs use the inverse dynamics knowledge to effectively replace the need for a high gain meaning that the VTR design can be used without the lower steady state gain resulting in reduced tracking accuracy.

 The improved thermal comfort obtained when the VTR controller is used with a sensor lag is easily apparent. A case for reduced energy consumption when using the VTR algorithm can also be made. It can be seen from Figure 9 that the oscillations cause the air temperature to reach a minimum of 19oC during the operation of the controller. It can be supposed that the occupants would then raise the air temperature setpoint to meet the minimum air temperature. With the addition of VTR it would not be necessary to raise the temperature setpoint as oscillations are no longer present. The increase in energy usage caused by raising the setpoint can be estimated by running the simulation for an entire year and summing the power used by the heating element. Supposing that the minimum air temperature is defined as 20oC then for the RIDE controller the setpoint must be raised by 1oC to meet this requirement, whereas the RIDE/VTR controller does not require the setpoint to be raised. This results in the RIDE controller having a **17% increase in heating energy consumption** over the whole year compared to the RIDE/VTR controller. Supposing that the minimum air temperature is 21oC necessitates raising the setpoint by 2oC for the RIDE controller. This results in a **25% increase in energy consumption** compared to the VTR controller. This simple analysis demonstrates that even a small change in temperature setpoint can have a very signification impact on the energy consumption of the HVAC system. Therefore, being able to accurately and *robustly* track a specified setpoint is of great importance.

1. Conclusions

This paper has presented a controller design to enable high performance control of a HVAC system. The controller design used a new type of Nonlinear Dynamic Inverse (NDI) by combining RIDE and VTR techniques to enable simultaneous control of internal air temperature and relative humidity. The benefits of using the new NDI design compared to a traditional Proportional Integral PI based controller are significantly reduced interaction between the heating and ventilation systems as well as reduced sensitivity to disturbances. It was demonstrated that, in order to implement a NDI design, very little extra sensory information and knowledge of the building structure is required compared to a PI design. The only structure information needed is the mass and specific heat capacity of the airspace to be controlled and the extra sensory information is the external air temperature and humidity. A method for limiting the controller output so as not to overdrive the HVAC actuation system was presented. Overdriven actuation systems are essentially using energy that does not result in any performance gain and so it is important to prevent this to minimize the overall energy consumption of the HVAC system.

The robustness qualities of the NDI controller were improved by introducing a nonlinear, dynamic gain changing design known as Variable Transient Response (VTR). The addition of VTR significantly reduced the sensitivity of the controller to sensor lags by reducing the steady state gain whilst preserving the speed of response to the setpoint.

A simulation of a controlled HVAC system for a modern office space was undertaken to compare the performance of a PI controller with that of NDI and VTR designs. The simulation results demonstrate the performance and energy saving gains that could be made possible by using an advanced NDI controller design and the added robustness made possible by using VTR in conjunction with a NDI controller.

# References

x

|  |  |
| --- | --- |
| 1. | Tashtoush B, Molhim M, Al-Rousan M. Dynamic Model of an HVAC System for Control Analysis: Energy; 2005. |
| 2. | Bia Q, Caib W, Wangc Q, Hangc C, et al.. Advanced Controller Auto-Tuning and its Application in HVAC Systems: Control Engineering Practice; 2000. |
| 3. | Counsell JM, Khalid YA, Brindley J. Controllability of Buildings: A Multi-Input Multi-Output Stability Assessment Method for Buildings with Slow Acting Heating Systems: Simulation Modelling Practice and Theory; 2011. |
| 4. | Ding L, Bradshaw A, Taylor C. Design of Discrete Time RIDE Control Systems Glasgow, UK: Proceedings of the UKACC International Conference on Control; 2006. |
| 5. | Steer AJ. Application of NDI Flight Control to a Second Generation Supersonic Transport Aircraft: Computing and Control Engineering Journal; 2001. |
| 6. | Counsell JM, Brindley J, MacDonald M. Non-Linear Autopilot Design Using the Philosophy of Variable Transient Response Chicago: AIAA Guidance, Navigation, and Control Conference; 2009. |
| 7. | Gouda MM UCDS, Gouda MM, Underwood CP, Danaher S. Modelling the Robustness Properties of HVAC Plant Under Feedback Control: Building Services Engineering Services and Technology; 2003. |
| 8. | Murphy GB, Counsell JM. Symbolic Energy Estimation Model with Optimum Start Algorithm Leicester, UK: CIBSE Technical Symposium; 2011. |
| 9. | Rentel-Gomez C, Velez-Reyes M. Decoupled Control of Temperature and Relative Humidity using a Varaible-Air-Volume HVAC System and Non-interacting Control Mexico City: Proceedings of the 2001 IEEE International Control Conference on Control Applications; 2001. |
| 10. | Franklin GF, Powell JD, Emami-Naeini A. Feedback Control of Dynamic Systems: Prentice Hall; 2001. |
| 11. | Lane SH, Stengel RF. Flight Control Design Using Non-Linear Inverse Dynamics: Automatica; 1988. |
| 12. | Muir E, Bradshaw A. Control Law Design for a Thrust Vectoring Fighter Aircraft using Robust Inverse Dynamics Estimation (RIDE): IMechE; 1996. |
| 13. | Counsell JM, Bradshaw A. Design of Autopilots for High performance Missiles: ImechE; 1992. |
| 14. | Tarbouriech S, Turner M. Anti-Windup Design: An Overview of Some Recent Advances and Open Problems: Control Theory and Applications; 2009. |

x

1. Appendix

 $a\_{11}=\frac{1}{M\_{a}C\_{a}}\left(-U\_{f}A\_{f}-U\_{r}A\_{r}-U\_{w}A\_{w}-\dot{m}\_{nv}C\_{a}-U\_{tm}A\_{tm}-h\_{i}A\_{s}\right)$ (59)

 $a\_{12}=\frac{1}{M\_{a}C\_{a}}h\_{i}A\_{s}$ (60)

 $a\_{14}=\frac{1}{M\_{a}C\_{a}}U\_{tm}A\_{tm}$ (61)

 $a\_{21}=\frac{1}{M\_{si}C\_{si}}h\_{i}A\_{s}$ (62)

 $a\_{22}=\frac{1}{M\_{si}C\_{si}}\left(-h\_{i}A\_{s}-\frac{k}{wt}A\_{s}\right)$ (63)

 $a\_{23}=\frac{1}{M\_{si}C\_{si}}\frac{k}{wt}A\_{s}$ (64)

 $a\_{32}=\frac{1}{M\_{se}C\_{se}}\frac{k}{wt}A\_{s}$ (65)

 $a\_{33}=\frac{1}{M\_{se}C\_{se}}\left(-\frac{k}{wt}A\_{s}-h\_{e}A\_{s}\right)$ (66)

 $a\_{41}=\frac{1}{M\_{tm}C\_{tm}}U\_{tm}A\_{tm}$ (67)

 $a\_{44}=\frac{-1}{M\_{tm}C\_{tm}}U\_{tm}A\_{tm}$ (68)

 $a\_{55}=-\frac{1}{M\_{a}}\dot{m}\_{nv}$ (69)

 $a\_{61}=\frac{1.388}{M\_{a}C\_{a}}\left(U\_{f}A\_{f}+U\_{r}A\_{r}+U\_{w}A\_{w}+\dot{m}\_{nv}C\_{a}+U\_{tm}A\_{tm}+h\_{i}A\_{s}\right)$ (70)

 $a\_{62}=-\frac{1.388}{M\_{a}C\_{a}}h\_{i}A\_{s}$ (71)

 $a\_{64}=-\frac{1.388}{M\_{a}C\_{a}}U\_{tm}A\_{tm}$ (72)

 $a\_{65}=-\frac{5000}{M\_{a}}\dot{m}\_{nv}$ (73)

 $d\_{11}=\frac{1}{M\_{a}C\_{a}}$ (74)

 $d\_{12}=\frac{1}{M\_{a}C\_{a}}\left(U\_{r}A\_{r}+U\_{w}A\_{w}+\dot{m}\_{nv}C\_{a}\right)$ (75)

 $d\_{15}=\frac{1}{M\_{a}C\_{a}}\left(U\_{f}A\_{f}\right)$ (76)

 $d\_{32}=\frac{1}{M\_{se}C\_{se}}\left(h\_{e}A\_{s}\right)$ (77)

 $d\_{53}=\frac{1}{M\_{a}}$ (78)

 $d\_{54}=\frac{1}{M\_{a}}\dot{m}\_{nv}$ (79)

 $d\_{61}=-\frac{1.388}{M\_{a}C\_{a}}$ (81)

 $d\_{62}=-\frac{1.388}{M\_{a}C\_{a}}\left(U\_{r}A\_{r}+U\_{w}A\_{w}+\dot{m}\_{nv}C\_{a}\right)$ (82)

 $d\_{63}=\frac{5000}{M\_{a}}$ (83)

 $d\_{64}=\frac{5000}{M\_{a}}\dot{m}\_{nv}$ (84)

 $d\_{65}=-\frac{1.388}{M\_{a}C\_{a}}\left(U\_{f}A\_{f}\right)$ (85)