Proceedings of the
47th IEEE Conference on Decision and Control
Cancun, Mexico, Dec. 9-11, 2008

Performance Enhancement of Modular Control Systems Using μ Synthesis

Chris Vermillion, Jing Sun, and Ken Butts

Abstract— **For complex dynamic systems, a modular control design process is often employed, wherein the overall design is partitioned into smaller modules. The designers of each module only possess a model for a particular subset of the entire plant as well as closed loop *performance specifications* for the other module(s). In this paper, we will examine a common modular control strategy in which an outer loop controller computes a desired virtual control input and the inner loop computes real control inputs in order to achieve this desired virtual control input as closely as possible. The outer loop design is based on a specification for the inner loop, which may not always be achieved. We propose a *modular control error compensator* that is aimed at mitigating the performance degradation caused when the inner loop specifications are not achieved. We show that this compensator can be designed using μ synthesis and propose an iterative procedure to optimize performance based on two concrete worst-case metrics. The effectiveness of the proposed compensator is shown through an automotive example.**

I. INTRODUCTION

A common control strategy for many automotive, aerospace, and marine applications is shown in Fig. 1, where the control design is divided between outer and inner loop systems, with the ultimate objective of having the performance variable, $y \in \mathbb{R}$, track a given setpoint, $r \in \mathbb{R}$. The outer loop controller computes a desired overall force, moment, or generalized effect, $v_{des} \in \mathbb{R}$, whereas the inner loop controller, whose function is commonly referred to as *control allocation*, determines real control inputs, $u \in \mathbb{R}^p$ (for a p input system), for the actuators, in order to achieve this overall force or moment as closely as possible. v is commonly referred to as the *virtual control input* (and will be in this work), even though it does in fact represent a physical signal (such as a force or moment) and is only "virtual" in the sense that it is not at the immediate control of the actuators.

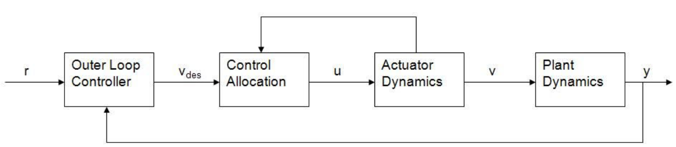

Fig. 1. Block Diagram of the Overall System Under a Modular Control Strategy

This work was supported by Toyota Engineering and Manufacturing in North America.

C. Vermillion and J. Sun are with the Electrical Engineering and Computer Science Department and Naval Architecture and Marine Engineering Department (respectively), University of Michigan, Ann Arbor, MI 48109, cvermill@umich.edu, jingsun@umich.edu

K. Butts is with Toyota Technical Center, Ann Arbor, MI 48105, ken.butts@tema.toyota.com

The control strategy of Fig. 1 is widely employed in industry due to its practicality in addressing the plant dynamics and actuator dynamics through separate controllers, particularly when different vendors are responsible for different components. Examples include [1]-[3] for automotive systems, [4]-[7] for marine applications, and [8]-[12] for flight control.

While the modular control strategy shown in Fig. 1 is widely used in practice, the fact that two controllers are being designed in parallel leads to several issues regarding the overall system performance when the inner and outer loops are combined. In order to facilitate the parallel design of the inner and outer loops, the designer of the inner loop is typically given a set of control specifications to be satisfied, and the outer loop design is carried out under the assumption that these specifications are met. In reality, the inner loop, with its controller in place, may not meet the specifications for various reasons, such as model uncertainties, controller simplification, and overly aggressive specifications. This mismatch between the actual and ideal characteristics will lead to a performance deterioration of the overall system. In this work, we propose a control strategy aimed at recovering the performance of the system when the inner loop controller fails to meet the specification exactly. The goal is to compensate for the mismatch between specifications and actual performance without altering the performance of the system when the mismatch vanishes.

The proposed control structure is shown in Fig. 2, where a *modular control error compensator* (MCEC) is used to recover performance in the event that the inner loop specification is not attained. Here, F represents the inner loop specification, which reflects the ideal closed loop behavior of the inner loop. C, P, and C_v represent the outer loop controller, plant, and MCEC, respectively. We assume that the inner and outer loop controllers have been designed from the outset and that the inner loop controller results in inner closed loop dynamics given by T, where T may not be equal to F. The proposed MCEC design structure exploits the signal $\tilde{v}$, which represents the difference between the virtual control input, v, and its nominal value, v_{des}^f (the output of F), to minimize the performance degradation of the overall system. While the controller structure proposed in Fig. 2 is applicable to both linear and nonlinear systems, the scope of this paper is limited to systems in which the blocks in Fig. 2 are linear.

In this paper, the performance deterioration is defined by the difference between the true closed loop transfer function from r to y and the "ideal" transfer function that is attained

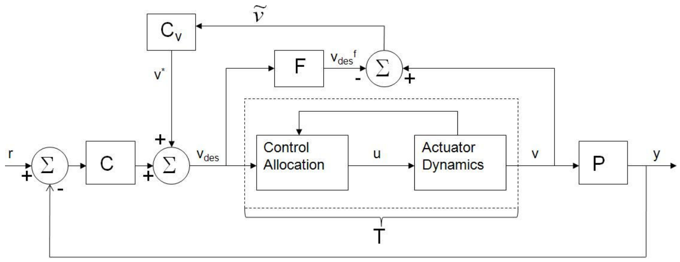

Fig. 2. Block Diagram of the Overall System Under a Modular Control Strategy

when $T = F$. Furthermore, we assume that there exists a bound on the uncertainties that create the difference between T and F. One approach to designing robust controllers in the presence of bounded uncertainties is to choose a compensator that minimizes the performance deterioration of the system under a worst-case uncertainty scenario. One can use the μ synthesis approach, which is detailed in [17]-[18], and employed in numerous application examples, such as [13]-[16], to design C in Fig. 2 to optimize a worst case performance metric. This design strategy leads to robust performance in a min-max sense but leads to an important tradeoff between nominal performance and performance in the presence of uncertainty.

Rather than redesigning C to compensate for uncertainty, we propose to leave C fixed and use μ synthesis to design an additional compensator, C_v, to address the performance deterioration. The structure shown in Fig. 2 has two key advantages:

1) It retains the modularity of the inner and outer loops in the sense that it allows the two controllers to be designed in parallel without knowledge of each other;
2) When inner loop specifications are met ($T = F$), the nominal performance will be recovered automatically, since $\tilde{v}$ will go to 0.

In this paper, we show that by recasting the optimization problem for the MCEC design, we can formulate a μ synthesis problem to optimize specific worst-case performance metrics. The paper is organized as follows. In Section II, we quantify the metric to be used to define the worst-case performance for the system of Fig. 2, then set up a μ synthesis problem in Section III such that this worst case performance can be optimized through the design of C_v. In Section IV, we present design and analysis results on an automotive thermal management system.

II. PROBLEM FORMULATION AND DESIGN OBJECTIVES

The purpose of MCEC is to compensate for a mismatch between the achieved and specified inner loop performance. In order to formulate the design problem, we assume a multiplicative uncertainty structure for the inner closed loop, which will be used to describe and place a bound on this performance mismatch:

$$T = (1 + \Delta W)F, \tag{1}$$

$$\|\Delta\|_\infty \leq \Delta_{max} \tag{2}$$

where Δ represents bounded inner closed loop uncertainty, and W is a known stable transfer function that captures information known about the uncertainty (typically W is a high-pass filter for a multiplicative uncertainty structure).

Remark 2.1: Note that the uncertainty is given for the inner *closed* loop, and consequently the characteristics of the uncertainty set, such as the bound, will, in general, be affected by the inner loop control law as well as uncertainties in the system dynamics. The appropriate choice of inner loop control law and specification (F) is an important engineering judgment that will have an effect on the closed inner loop uncertainty.

Remark 2.2: The choice of uncertainty structure (additive, multiplicative, feedback, etc.) should be chosen to be most representative for the application at hand. In this paper, a multiplicative uncertainty structure is employed. However, the procedures employed in this paper are applicable to any other uncertainty structures in which Δ is interconnected with the remainder of the system dynamics.

In order to quantify the performance degradation, we define an error variable, $e = y - y^r$, where y^r is the output of the system when $T = F$ and will be referred to as the *nominal* output of the system. The transfer function from r to e is given by:

$$G^{er} = \frac{PTC}{1 + PTC + C_v(F - T)} - \frac{PFC}{1 + PFC}. \tag{3}$$

The performance metric for the design and analysis of C_v will be the worst case gain, $\gamma^{wc}(C_v)$, from r to e, which is given by:

$$\gamma^{wc}(C_v) = \max_{\Delta:\|\Delta\|_\infty \leq \Delta_{max}} \|G^{er}\|_\infty. \tag{4}$$

The objective of C_v is to make y track y^r as closely as possible in spite of the inner loop uncertainty. In defining this as our objective, we have assumed that the closed loop behavior of the system with the inner loop specified by F represents the ideal behavior. Depending on the a priori knowledge about the uncertainties and other design constraints, we can formulate two synthesis problems that relate to this objective:

S1 Given that $\|\Delta\|_\infty < 1$, find a stabilizing C_v that minimizes γ^{wc}.

S2 Given $\gamma^{th} > 0$ (γ^{th} is a performance threshold representing tolerable performance degradation), find C_v that maximizes $\|\Delta\|_\infty$ such that the closed loop system remains stable and $\gamma^{wc} < \gamma^{th}$.

Remark 2.3: Each synthesis objective appeals to a slightly different control problem. Objective S1 aims for maximum performance recovery when given a bound on the closed inner loop uncertainty. On the other hand, objective S2 maximizes allowable uncertainty for a given performance threshold. It will be shown that the same design tool can be used to achieve both synthesis objectives.

Remark 2.4: Worst case performance may be more generically defined by:

$$\gamma^{wc}(C_v) = \max_{\Delta:\|\Delta\|_\infty \leq \Delta_{max}} \|W^g G^{er}\|_\infty. \quad (5)$$

where W^g is a weighting function that penalizes performance degradation at particular frequencies, thereby allowing more design flexibility.

III. MCEC DESIGN USING μ SYNTHESIS

For the purposes of designing C_v to achieve our synthesis objectives, S1 and S2, the block diagram of Fig. 2 can be cast in the form of the block diagram given by Fig. 3, where $\bar{P}$ is a 3 x 3 transfer function matrix containing all of the system components besides C_v and Δ (θ is a fictitious signal that represents the output of the uncertainty block, Δ). One can show through block diagram manipulations that $\bar{P}$ is given by:

$$\bar{P} = \begin{pmatrix} \frac{V_{des}(s)}{\theta(s)} & \frac{V_{des}(s)}{R(s)} & \frac{V_{des}(s)}{V^*(s)} \\ \frac{E(s)}{\theta(s)} & \frac{E(s)}{R(s)} & \frac{E(s)}{V^*(s)} \\ \frac{\tilde{V}(s)}{\theta(s)} & \frac{\tilde{V}(s)}{R(s)} & \frac{\tilde{V}(s)}{V^*(s)} \end{pmatrix} = \begin{pmatrix} \frac{CPFW}{1+PFC} & \frac{C}{1+PFC} & \frac{1}{1+PFC} \\ \frac{PFW}{1+PFC} & \frac{-1}{1+PFC} & \frac{PF}{1+PFC} \\ FW & 0 & 0 \end{pmatrix}. \quad (6)$$

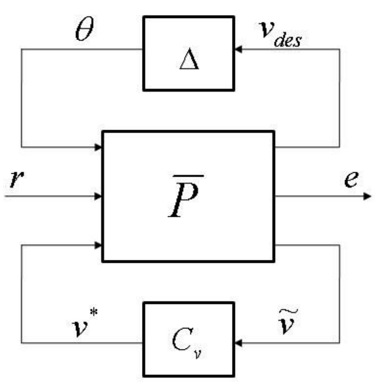

Fig. 3. Generic Control Problem Formulation

Given this system representation, we seek either to minimize the worst case gain, γ^{wc} (S1), or to maximize the size of uncertainty under which $\gamma^{wc} < \gamma^{th}$ (S2). Both of these objectives lend themselves to the μ synthesis design tool, given the following proposition [17]:

Proposition 3.1: (Interpretation of μ) The structured singular value is the smallest scalar value μ such that the system of Fig. 3 is stable for all $\Delta : \|\Delta\|_\infty < \frac{1}{\mu}$. Furthermore, there exists a transfer function $\Delta : \|\Delta\|_\infty = \frac{1}{\mu}$, which results in an unstable closed loop system.

□

Given Proposition 3.1, μ characterizes the allowable levels of uncertainty for closed loop stability. μ synthesis is the process of designing a controller to minimize μ for a particular closed loop system. The process of μ synthesis has been facilitated and streamlined through the development of numerical tools such as the MATLAB Robust Control Toolbox that is used for the design example in this work.

Without further modification to the block diagram of Fig. 3, μ does not have a clear connection with the performance goals encompassed by our synthesis objectives (S1 and S2). To consider performance, we augment the uncertainty structure of Fig. 3 such that the reference-error loop is wrapped around a fictitious uncertainty block, Δ_P, and a scalar gain block, k, as in Fig. 4. Under this configuration, μ synthesis considers an uncertainty structure given by:

$$\Delta_{aug} \triangleq \begin{pmatrix} \Delta & 0 \\ 0 & \Delta_P \end{pmatrix} \quad (7)$$

This leads to the following result [17], which follows from (7) and the small gain condition:

Proposition 3.2: (Robust performance interpretation of μ) Let μ be the structured singular value for the system shown in Fig. 4. The following two properties hold for μ:

1) The closed loop system is stable and $k\|G^{er}\|_\infty < \mu$ for all uncertainties satisfying $\|\Delta\|_\infty < \frac{1}{\mu}$.
2) There exists a perturbation $\Delta : \|\Delta\|_\infty = \frac{1}{\mu}$ for which the closed loop system is unstable or $k\|G^{er}\|_\infty = \mu$.

□

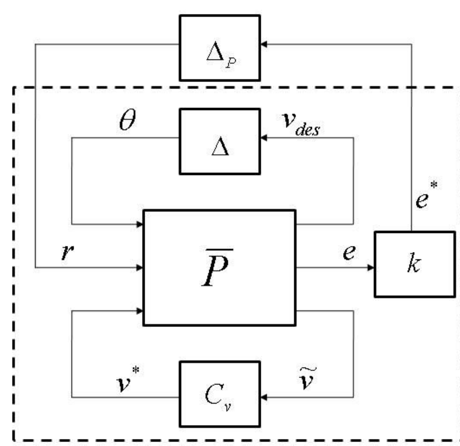

Fig. 4. μ-synthesis Performance Formulation

The adjustable parameter k weights the relative importance of robust performance vs. stability and is used here to provide a mechanism such that the synthesis objectives S1 and S2 can be achieved. The following propositions allow us to develop a design algorithm to achieve the synthesis objectives by iteratively adjusting k and carrying out μ synthesis to design C_v.

Proposition 3.3: (Synthesis Objective (S1)) Assume that S1 is feasible. Let $\mu^*(k)$ be the structured singular value of the system shown in Fig. 4 where C_v has been designed using μ synthesis, for a given constant k. Furthermore, let k_1^* be the maximum value of k such that $\mu^*(k) = 1$. Then k_1^* is guaranteed to exist, and if C_v is designed using μ synthesis, with $k = k_1^*$, then γ^{wc} is minimized and the closed loop is stable, subject to $\|\Delta\|_\infty < 1$.

□

Proof: *Existence:* From the feasibility assumption on S1, there exists a stabilizing C_v when $\|\Delta\|_\infty < 1$, which results in a finite γ^{wc}. Taking $k = \frac{1}{\gamma^{wc}}$, it follows from Proposition 3.2 that $\mu(C_v, k) = 1$, thereby proving existence.

When C_v is designed with $k = k_1^*$, the closed loop is stable and γ^{wc} is minimized, subject to $\|\Delta\|_\infty < 1$: Closed loop stability when $\mu^* = 1$ follows directly from property (1) of Proposition 3.2. To show that γ^{wc} is minimized, we first show by contradiction that $\gamma^{wc} = \frac{\mu^*}{k_1^*}$. Suppose that $\gamma^{wc} < \frac{\mu^*}{k_1^*}$ ($\gamma^{wc} > \frac{\mu^*}{k_1^*}$ violates Proposition 3.2, property (1), and need not be considered). Then there exists $k > k^*$ that also yields $\mu^*(k) = 1$, which contradicts the fact that k_1^* is the *maximum* value of k yielding $\mu^*(k) = 1$. Therefore, $\gamma^{wc} = \frac{\mu^*}{k_1^*} = \frac{1}{k_1^*}$. Because we are maximizing k, γ^{wc} is minimized for all C_v, k that yield $\mu(C_v, k) = \mu^*(k) = 1$, and because μ synthesis minimizes μ, γ^{wc} is indeed minimized for all C_v, k that yield $\mu(C_v, k) = 1$. Finally, we must show that γ^{wc} is minimized for all C_v, k that lead to closed loop stability for $\|\Delta\|_\infty < 1$. To do this, note from the existence proof that for all C_v that yield a stable closed loop when $\|\Delta\|_\infty < 1$, there exists a k that yields $\mu(C_v, k) = 1$. Therefore, by minimizing γ^{wc} over all C_v, k that yield $\mu(C_v, k) = 1$, we also minimize γ^{wc} over all C_v that yield a stable closed loop when $\|\Delta\|_\infty < 1$. ■

Proposition 3.4: (Synthesis Objective (S2)) Assume that S2 is feasible. Let $\mu^*(k)$ be the structured singular value of the system shown in Fig. 4 where C_v has been designed using μ synthesis, for a given constant k. Let k_2^* be the minimum value of k such that $\mu^*(k) = k\gamma^{th}$. k_2^* is guaranteed to exist, and if C_v is designed with $k = k_2^*$, then μ synthesis maximizes the value of $\|\Delta\|_\infty$ such that $\gamma^{wc} < \gamma^{th}$ and the closed loop system remains stable.

□

Proof: *Existence:* From the feasibility assumption on S2, there exists a C_v that results in $\gamma^{wc} < \gamma^{th}$ for $\|\Delta\|_\infty < \Delta^*$, where Δ^* is the maximum tolerable uncertainty. Take $k = \frac{1}{\Delta^*\gamma^{th}}$, which will result in $\mu(C_v, k) = \frac{1}{\Delta^*}$. Therefore, we have $\mu(C_v, k) = k\gamma^{th}$, thereby proving existence.

When C_v is designed with $k_2 = k_2^*$, tolerable uncertainty is maximized: To show that tolerable uncertainty is maximized, we first show, by contradiction, that there exists $\Delta : \|\Delta\|_\infty = \frac{1}{\mu^*}$ that leads to closed loop instability. Suppose that there does not exist $\Delta : \|\Delta\|_\infty = \frac{1}{\mu^*}$ that leads to closed loop instability. Then there exists $k < k_2^*$ that yields $\mu^*(k) = k\gamma^{th}$, which contradicts the fact that k_2^* is the minimum value of k yielding $\mu^*(k) = k\gamma^{th}$. Therefore, there *does* exist $\Delta : \|\Delta\|_\infty = \frac{1}{\mu^*}$ that leads to closed loop instability. Since k_2^* is minimized, tolerable uncertainty is maximized for all C_v, k yielding $\mu(C_v, k) = \mu^*(k) = k\gamma^{th}$, and because μ synthesis minimizes μ, tolerable uncertainty is indeed maximized for all C_v, k yielding $\mu(C_v, k) = k\gamma^{th}$. Finally, it follows from the existence proof that there always exists a k that yields $\mu(C_v, k) = k\gamma^{th}$. Therefore, tolerable uncertainty is maximized over all C_v such that $\gamma^{wc} < \gamma^{th}$ and the closed loop system remains stable. ■

Using Propositions 3.3 and 3.4 and treating k as an adjustable parameter, we propose the following iterative algorithm to design the MCEC controller C_v to achieve objectives S1 and S2; for S1, take $\mu^o = 1$, for S2, take $\mu^o = k\gamma^{th}$:

1) a) Initialize k_{low} to any value that is known to satisfy $k_{low} < k_i^*$ ($i = 1$ or $i = 2$ depending on the objective). Take $k_{low} = 0$ if no other lower bound is known. Proceed to step (1b).
   b) Initialize k_{high} to any value that is known to satisfy $k_{high} > k_i^*$, and proceed to step (2). If no upper bound on k is known, make an initial guess, k_{init}, and carry out μ synthesis for $k = k_{init}$.
      i) If $\mu^*(k_{init}) - \mu^o > \epsilon$, take $k_{high} = k_{init}$ and continue with step (2).
      ii) If $\mu^*(k_{init}) - \mu^o < \epsilon$, increase k_{init} and repeat step (1b).
2) Carry out μ synthesis for $k = \frac{1}{2}(k_{high} + k_{low})$, which will return C_v and μ^*. Proceed to (3).
3) a) If $|\mu^* - \mu^o| < \epsilon$, move to (4) for S1 and (5) for S2.
   b) If $\mu^* - \mu^o < -\epsilon$, set $k_{low} = k$ and repeat (2).
   c) If $\mu^* - \mu^o > \epsilon$, set $k_{high} = k$ and repeat (2).
4) a) If $|k_{high} - k| < \epsilon_k$, where ϵ_k is a user-defined threshold, terminate the algorithm.
   b) If $|k_{high} - k| \geq \epsilon_k$, take $k_{low} = k$ and move to (4c).
   c) Take $k = \frac{1}{2}(k_{low} + k_{high})$, and carry out μ synthesis.
      i) If $|\mu^*(k) - \mu^o| \geq \epsilon$, set $k_{high} = k$ and repeat (4c) until $|k_{high} - k| < \epsilon_k$.
      ii) Otherwise, take $k_{low} = k$ and repeat (4c) until $|k_{high} - k| < \epsilon_k$.
5) a) If $|k - k_{low}| < \epsilon_k$, where ϵ_k is a user-defined threshold, terminate the algorithm.
   b) If $|k - k_{low}| \geq \epsilon_k$, take $k_{high} = k$ and move to (5c).
   c) Take $k = \frac{1}{2}(k_{low} + k_{high})$, and carry out μ synthesis.
      i) If $|\mu^*(k) - \mu^o| \geq \epsilon$, set $k_{low} = k$ and repeat (5c) until $|k - k_{low}| < \epsilon_k$.
      ii) Otherwise, take $k_{high} = k$ and repeat (5c) until $|k - k_{low}| < \epsilon_k$.

Remark 3.1: As an alternative to the iterative procedure proposed here, a skew μ synthesis procedure may be employed, in which the size of one uncertainty block is held fixed while the other(s) are allowed to vary [19]-[20]. This framework requires additional mathematical tools to set up, but may result in reduced computational effort once implemented.

IV. ENGINE THERMAL MANAGEMENT APPLICATION

The proposed modular control design approach is now applied to an engine thermal management system, depicted in Fig. 5. The system is used to facilitate engine mapping and calibration on a dynamometer by providing tight regulation of coolant and oil temperatures at the outlet of an automotive engine. The physical system consists of two parallel loops (coolant and oil), each of which includes two actuators (a mixing valve and a heater), which are used to control the temperature of the fluid at the engine outlet. A detailed dynamic model for the system is given in [21], where it is also shown that the interaction between the coolant and oil loops is small, which will enable us to simplify our example here by considering only the coolant loop. The mixing valve and heater, which provide control inputs u_{mv} and u_{ht}, reside in a unit that is separated from the engine block itself and naturally forms the actuator subsystem, Σ_2. Consequently, the virtual control input is the engine inlet temperature, $T_{in,eng}$, and the engine block (including the fluid that passes through it) comprises the plant subsystem, Σ_1. The use of modular control for this application is well motivated by the fact that the actuator subsystem components are manufactured and assembled by a different company than the automotive company responsible for the engine mapping and calibration.

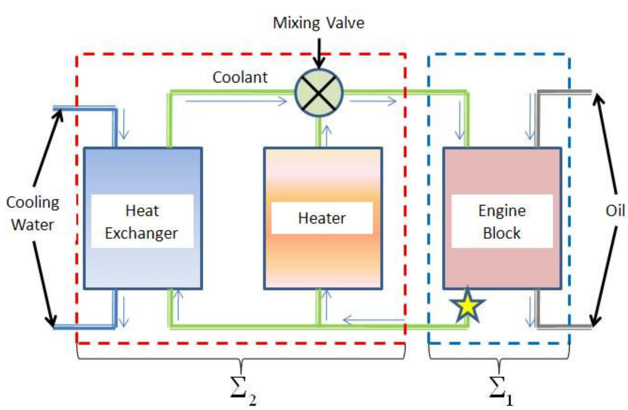

Fig. 5. Thermal Management System Diagram

Recent findings indicate the importance of modeling mixing valve temperature dynamics, which results in an inner loop transfer function of the form:

$$V(s) = A_0(s)\frac{1}{\tau_t s + 1}U_{mv}(s), \tag{8}$$

where $A_0(s)$ reflects the actuator dynamics in the absence of these temperature dynamics, and τ_t reflects the time constant associated with the mixing.

For this application, we will consider the problem of setpoint tracking at an operating condition of 3000 rpm, 100 N-m. We desire to achieve tracking of the desired virtual control input, $T_{in,eng}$, within a time constant of one second, leading to an inner loop performance specification,

$$F(s) = \frac{1}{s+1}. \tag{9}$$

The outer loop uses a PI controller that has been designed from the outset. With this outer loop controller in place, the simple control law, $u_{mv} = k_1 v_{des} - k_2 v$, $u_{ht} = u_{ht}^*$ [1] can be shown to result in an inner closed loop that tightly matches $F(s)$ in the absence of uncertainty. To account for the uncertainty in the closed inner loop, we introduce a multiplicative structured uncertainty, $T = (1 + \Delta W)F$, and for our example, the weighting function, W, is chosen as a high pass filter,

$$W(s) = \frac{Ks}{s+1}. \tag{10}$$

Here, K is a scalar used to represent the size of the uncertainty while maintaining $\|\Delta\|_\infty < 1$.

The design of C_v for the thermal management application is based on the first synthesis objective (S1), in an effort to minimize the worst-case gain of the closed loop system and recover as much performance as possible, under the bound, $\|\Delta\|_\infty < 1$. For implementation of MCEC, the resulting 60^{th} order compensator from μ synthesis is approximated by a much lower second order compensator,

$$C_v(s) = -\frac{s+1}{s^2 + 8s + 15}. \tag{11}$$

The benefits of applying C_v, as designed using μ synthesis, are illustrated in Fig. 6, where the worst case gain is given both with and without C_v in place. Clearly, the use of MCEC leads to performance recovery, as is intended, over the entire range of uncertainties analyzed.

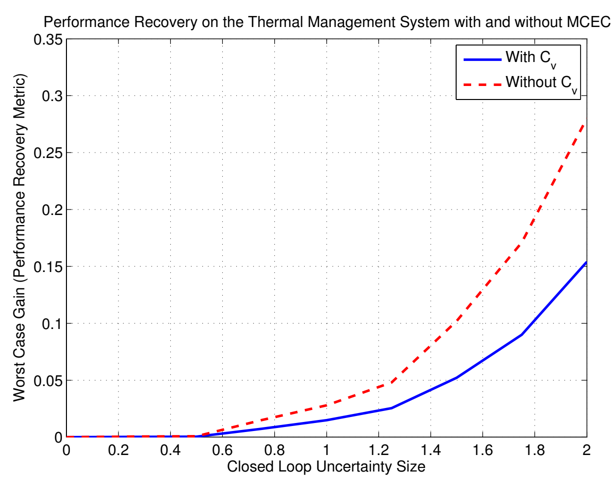

Fig. 6. Worst-case gain Comparison with and without MCEC

To see the benefit of MCEC through simulation results, consider a scenario in which the flow rates are not immediately affected by the mixing valve command, but rather are

[1] Here, u_{ht}^* is a constant heater input that is, in general, dependent on the operating condition of the system.

affected through a time constant, yielding actuator dynamics given by:

$$V(s) = A_0(s)\frac{1}{(\tau_t s+1)(\tau_f s+1)}U_{mv}(s), \quad (12)$$

where τ_f is the time constant associated with the flow dynamics.

Simulation results with and without the use of C_v are given in Fig. 7, where it is clear that the use of C_v drives the actual system performance closer to the nominal system performance. For the simulations in Fig. 7, we take $\tau_t = 5$ seconds and $\tau_f = 2$ seconds, which leads to uncertainty that falls within the bounds of $T = (1 + \Delta W)F$, with $K = 1.7$. We can see through simulation results that, even though the performance recovery as measured through γ^{wc} appears to be very good from Fig. 6 ($\gamma^{wc} = 0.17$ without C_v and $\gamma^{wc} = 0.08$ with C_v), there is still significant deviation from nominal performance in simulation, and the use of MCEC beings the actual system performance much closer to the specification.

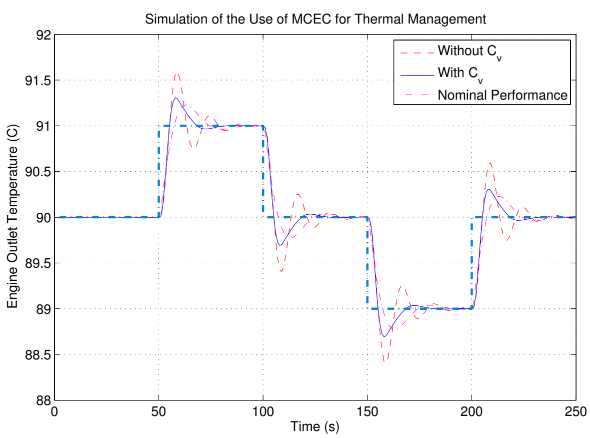

Fig. 7. Simulation on the Engine Thermal Management System

V. CONCLUSIONS AND FUTURE WORK

In this paper, we have proposed a modular control error compensator (MCEC) for recovering specified performance in modular control systems. The use of this compensator is beneficial for the integration of inner and outer loops when a modular control design is carried out. We have shown how μ synthesis may be used iteratively in order to achieve the design goals and have demonstrated the effectiveness of the proposed method on an engine thermal management system. Future work will include the development of deeper insights with regard to the tradeoffs between the inner loop performance specification, F, and the ability to achieve robust modular performance with the use of C_v.

VI. ACKNOWLEDGEMENT

The authors wish to acknowledge Professor Le Yi Wang of the Electrical and Computer Engineering Department at Wayne State University for discussions and suggestions on the problem formulation and control design.

REFERENCES

[1] J. H. Plumlee, D. M. Bevly, A. S. Hodel, "Control of a Ground Vehicle Using Quadratic Programming Based Control Allocation Techniques," *Proceedings of the American Control Conference*, 2004.
[2] S. Chantranuwanthana, H. Peng, "Adaptive Robust Control for Active Suspensions," *Proceedings of the American Control Conference*, 1999.
[3] S. Chantranuwanthana, H. Peng, "Adaptive Robust Force Control for Active Suspensions," *International Journal of Adaptive Control and Signal Processing*, Vol. 18, 2004, pp. 83-102.
[4] K. Lindeguard, T. Fossen, "Fuel Efficient Rudder and Propeller Control Allocation for Marine Craft: Experiments with a Model Ship," *IEEE Transactions on Control Systems Technology*, Vol. 11, No. 6, 2003, pp. 850-862.
[5] T. Johansen, T. Fuglseth, P. Tondel, T. Fossen, "Optimal Constrained Control Allocation in Marine Surface Vessels with Rudders," *Proceedings of the IFAC Conference of Manufacturing and Control of Marine Vehicles*, 2003.
[6] T. Johansen, T. Fossen, S. Berge, "Constrained Nonlinear Control Allocation with Singularity Avoidance Using Sequential Quadratic Programming," *IEEE Transactions on Control Systems Technology*, Vol. 12, No. 1, 2004, pp. 211-216.
[7] O. J. Sordalen, "Optimal Thrust Allocation for Marine Vessels," *Control Engineering Practice*, Vol. 5, No. 9, 1997, pp. 1223-1231.
[8] O. Harkegard, "Resolving Actuator Redundancy - Control Allocation vs. Linear Quadratic Control," *Proceedings of the European Control Conference*, 2003.
[9] M. Bodson, "Evaluation of Optimization Methods for Control Allocation," *Journal of Guidance, Control, and Dynamics*, Vol. 25, No. 4, July-August 2002, pp. 703-711.
[10] Y. Luo, A. Serrani, S. Yurkovich, D. Doman, M. Oppenheimer, "Model Predictive Dynamic Control Allocation with Actuator Dynamics," *Proceedings of the American Control Conference*, 2004.
[11] Y. Luo, A. Serrani, S. Yurkovich, D. Doman, M. Oppenheimer, "Dynamic Control Allocation with Asymptotic Tracking of Time-Varying Control Input Commands," *Proceedings of the American Control Conference*, 2005.
[12] Y. Luo, A. Serrani, S. Yurkovich, M. Oppenheimer, D. Doman, "Model Predictive Dynamic Control Allocation Scheme for Reentry Vehicles," *Journal of Guidance, Control, and Dynamics*, Vol. 30, No. 1, 2007, pp. 100-113.
[13] B. Boulet, B. A. Francis, P. C. Hughes, T. Hong, "μ Synthesis for a Large Flexible Space Structure Experimental Testbed," *Journal of Guidance, Control, and Dynamics*, Vol. 24, 2001.
[14] M. Steinbuch, P. Van Gross, G. Schooststra, P. Wortelboer, O. Bosgra, "μ Synthesis of a Compact Disc Player," *International Journal of Robust and Nonlinear Control*, Vol. 8, 1998.
[15] J. Bibel, H. Stalford, "μ Synthesis Autopilot Design for a Flexible Missle," *Aerospace Sciences Meeting*, 1991.
[16] R. Adams, S. Banda, "Robust Flight Control Design Using Dynamic Inversion and Structured Singular Value Synthesis," *IEEE Transactions on Control Systems Technology*, Vol. 1, 1993, pp. 80-92.
[17] G. Balas, J. Doyle, K. Glover, A. Packard, R. Smith, *μ Synthesis and Analysis Toolbox*, The Math Works, 2001.
[18] K. Zhou, J. Doyle, *Essentials of Robust Control*, Prentice Hall, 1997.
[19] R. Holland, P. Young, and C. Zhu, "Development of a Skew μ Upper Bound," *Proceedings of the European Control Conference*, 2003.
[20] G. Ferreres, J. Biannic, J. Magni, "A Skew μ Toolbox for Robustness Analysis," *Proceedings of the IEEE Symposium on Computer Aided Control Systems Design*, 2004.
[21] C. Vermillion, J. Sun, K. Butts, A. Hall, "Modeling and Analysis of a Thermal Management System for Engine Calibration," *Proceedings of the IEEE Conference on Control Applications*, 2006.