LAWRENCE
LIVERMORE
NATIONAL
LABORATORY

LLNL-TR-823797

Model Reference Adaptive Control (MRAC) for Additive Manufacturing

J. V. Candy

June 23, 2021

This work performed under the auspices of the U.S. Department of Energy by Lawrence Livermore National Laboratory under Contract DE-AC52-07NA27344.

Model Reference Adaptive Control (MRAC)

for

Additive Manufacturing (AM)

J. V. Candy (5/28/21)

I. INTRODUCTION

Model Reference Adaptive Control (MRAC) is based on the fundamental concept that the process under investigation is to be controlled to follow or "track" a reference system (model) characterized by a state/input/output model employing an adaptive optimization algorithm to adjust the controller parameters in real-time [1-8]. The generic structure of the MRAC is shown in Fig. 1 consisting of the following primary components: Reference model, Process (system) model, controller and the adaption algorithm. The basic structure of the controller is specified by a linear construct with the corresponding real-time adaption algorithms given by a gradient-type (so-called MIT rule) or based on stability theory (Lyapunov, hyperstability) [2, 3]. This approach to adaptive control is termed "direct", since the controller (parameters) are adjusted based on the component models/algorithm in contrast to the "indirect" approach that adjusts the process model parameters applying real-time system identification techniques [9-11].

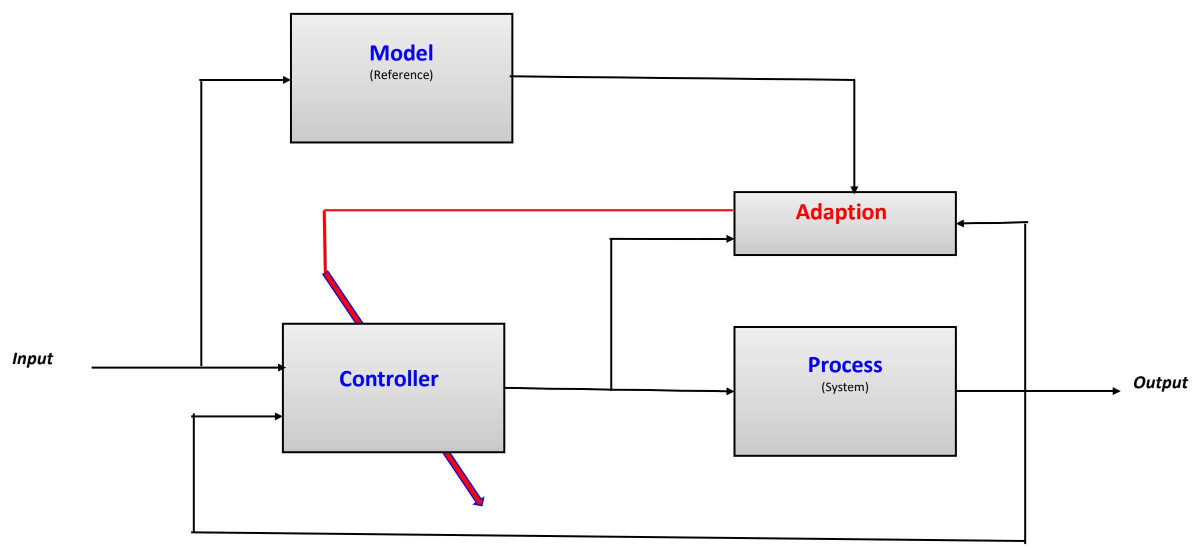

Figure 1. *Model Reference Adaptive Control (MRAC): Reference model, Process model, Controller and Adaption algorithm.*

The primary distinction between the MRAC and other methods is that the governing adaptive laws are incorporated and analyzed as part of entire control system rather than analyzing the criterion function itself. Next, we discuss the important components of the MRAC system and motivate its direct application [7, 8].

A. *Reference Model*

The reference model specifies the desired process behavior, is usually available in a parametric representation (e.g., transfer function/state-space models), and implemented in the control computer. The fact that its response must be precisely matched and therefore, it must be stable and minimum phase (poles/zeros in left-half plane) and be reasonably *representative* of the process under investigation.

B. *Controller*

The controller must satisfy some constraints in order to be an integral part of an MRAC system. It must: (1) satisfy the "perfect model matching" condition such that control parameters must exist to provide closed-loop response that matches that of the reference model; and (2) *direct* adaption implies a linear function of the control parameters, that is, a linear control law.

C. *Adaption*

The adaptive rules or laws have evolved from three basic approaches: (1) sensitivity models; (2) Lyapunov stability; and (3) Popov hyperstability—all attempting to guarantee a stable, closed-loop system. We will only consider the first two here.

Next, we develop some of the background information to develop various MRAC system designs.

II. BACKGROUND: *State-Space Systems*

The state-space provides a natural mechanism to incorporate measurements of a physical system together in a meaningful relationship [9, 12, 13]. It is a natural characterization of many systems incorporating the underlying phenomenology (physics), measurement system (measurements) and noise (uncertainties) by combining their interrelationships, mathematically, into a well-defined representation or model with inherent properties that can easily be analyzed to ensure a viable system.

We employ a simple structural system to motivate the state-space modeling concepts concentrating on a generic single channel mechanical system incorporating an accelerometer sensor measurement model. Consider a simple mechanical representation that can be characterized by a set of linear, constant coefficient, ordinary differential equations based on the corresponding mass-damper-spring (MCK) "parameters" governing its behavior, that is,

$$\overbrace{m}^{\text{mass}} \underbrace{\ddot{d}(t)}_{\text{acceleration}} + \overbrace{c}^{\text{damper}} \underbrace{\dot{d}(t)}_{\text{velocity}} + \overbrace{k}^{\text{spring}} \underbrace{d(t)}_{\text{displacement}} = \underbrace{F_a(t)}_{\text{excitation}} \qquad \text{[Mechanical System]} \qquad (1)$$

To convert this linear time-invariant (LTI) system of equations into the state-space, define the state-vector in terms of the displacement d and velocity $\dot{d}$, then it follows that

$$\mathbf{x(t)} := \begin{bmatrix} x_1(t) \\ x_2(t) \end{bmatrix} = \begin{bmatrix} d(t) \\ \dot{d}(t) \end{bmatrix}$$

Rewriting the 2nd-order differential equation of Eq. 1 and converting to the equivalent set of 1st-order state equations is obtained by solving for the highest derivative and using this definition of state vector to obtain a set of two 1st-order state equations as

$$\begin{aligned} \dot{x}_1(t) &= x_2(t) \\ \dot{x}_2(t) &= -\frac{c}{m} x_2(t) - \frac{k}{m} x_1(t) + F_a(t) \end{aligned} \qquad \text{[State Equations]} \qquad (2)$$

Placing these relations into *vector-matrix* form gives the well-known *state equations* as

$$\underbrace{\frac{d}{dt}\begin{bmatrix} x_1(t) \\ x_2(t) \end{bmatrix}}_{\dot{\mathbf{x}}(t)} = \underbrace{\frac{d}{dt}\begin{bmatrix} x(t) \\ \dot{x}(t) \end{bmatrix}}_{\dot{\mathbf{x}}(t)} = \underbrace{\begin{bmatrix} 0 & 1 \\ -\frac{k}{m} & -\frac{c}{m} \end{bmatrix}}_{A} \underbrace{\begin{bmatrix} x(t) \\ \dot{x}(t) \end{bmatrix}}_{\mathbf{x}(t)} + \underbrace{\begin{bmatrix} 0 \\ 1/m \end{bmatrix}}_{B} \underbrace{F_a(t)}_{\mathbf{u}(t)} \qquad \text{[State-Space]} \qquad (3)$$

More compactly, we have

$$\dot{\mathbf{x}}(t) = A\mathbf{x}(t) + B\mathbf{u}(t) \qquad \text{[State-Space]} \qquad (4)$$

The underlying physical phenomenology is captured by these state equations and in this case---the mechanical system.

Next a measurement system model can be developed and incorporated into the overall system model. If a sensor is developed to measure the acceleration directly, it can be modeled by including a scale factor conversion, say C_a, to give

$$y(t) = C_a \ddot{x}(t) = \begin{bmatrix} 0 & C_a \end{bmatrix} \frac{d}{dt} \mathbf{x}(t) = \begin{bmatrix} 0 & C_a \end{bmatrix} \left(A\mathbf{x}(t) + Bu(t) \right) \tag{5}$$

by substituting Eq. 3. Performing the indicated multiplications provides the underlying measurement model

$$y(t) = [0 \quad C_a] \underbrace{\begin{bmatrix} 0 & 1 \\ -\frac{k}{m} & -\frac{c}{m} \end{bmatrix}}_{A} \underbrace{\begin{bmatrix} x(t) \\ \dot{x}(t) \end{bmatrix}}_{\mathbf{x}(t)} + [0 \quad C_a] \underbrace{\begin{bmatrix} 0 \\ 1 \end{bmatrix}}_{B} \underbrace{F_a(t)}_{u(t)} = \underbrace{\begin{bmatrix} -\frac{k}{m} C_a & -\frac{c}{m} C_a \end{bmatrix}}_{C} \underbrace{\begin{bmatrix} x(t) \\ \dot{x}(t) \end{bmatrix}}_{\mathbf{x}(t)} + \underbrace{C_a}_{D} \underbrace{F_a(t)}_{u(t)} \tag{6}$$

or more succinctly

$$y(t) = C\mathbf{x}(t) + Du(t) \qquad \text{[Measurement System]} \tag{7}$$

This completes the discussion of mechanical systems in state-space form that will be incorporated into subsequent discussions.

III. BACKGROUND: *Model-Reference Control Systems*

MRAC is a control technique developed to follow or “track” a preferred model (reference) response and instantaneously make parametric adjustments (adaption) to alter (control) the underlying process (system) producing the “desired” output. Consider a more detailed look into the essentials of a MRAC system as depicted in Fig. 2.

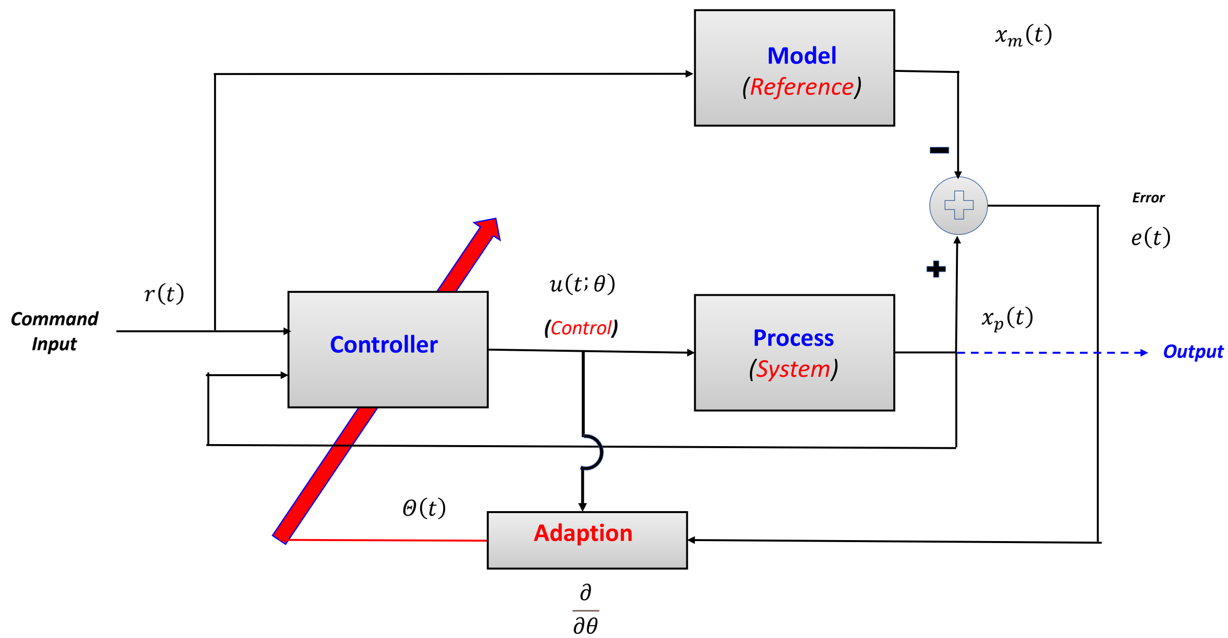

Figure 2. *Model Reference Adaptive Control (MRAC): Reference model, Process model, Controller and Adaption algorithm.*

Mathematically, MRAC is characterized by the following "generic" relations:

- REFERENCE model to provide the *desired* performance governed by the state equations:

$$\begin{aligned} \dot{x}_m(t) &= A_m x_m(t) + B_m r(t) \\ y_m(t) &= C_m x_m(t) \end{aligned} \tag{8}$$

- PROCESS or the *system* to be controlled governed by the state equations:

$$\begin{aligned} \dot{x}_p(t) &= A_p x_p(t) + B_p u(t;\theta) \\ y_p(t) &= C_p x_p(t) \end{aligned} \tag{9}$$

- CONTROLLER to provide the *control* governed by parametric equation:

$$u(t;\theta) = \theta_x x_p(t) - \theta_r r(t) \tag{10}$$

- ERROR to provide the *residuals* to be adjusted by the control parameters:

$$e(t;\theta) = x_p(t;\theta) - x_m(t) \tag{11}$$

- ADAPTION algorithm to instantaneously adjust the parametric (error) equations:

$$\frac{d\theta}{dt} = -\gamma e(t;\theta) \times \frac{\partial e(t;\theta)}{\partial \theta} \qquad \text{[MIT Rule]}$$

or (12)

$$\frac{d\theta}{dt} = -\gamma \Theta(t) \times e(t;\theta) \qquad \text{[LYAPUNOV Rule]}$$

IV. BACKGROUND: *Model-Reference Adaptive Control System Theory*

In this section we discuss two of the primary adaptive laws or rules that govern the design of MRAC systems: (1) Sensitivity approach or MIT Rule; and (2) Lyapunov stability rule.

A. Adaptive Algorithm: Sensitivity Model (MIT Rule) [7, 8]

The sensitivity approach evolves by selecting a quadratic performance criterion and minimizing the tracking error over a period *T*, that is, define the integral criterion as

$$J(t+T) = \tfrac{1}{2}\int_{t}^{t+T} e^2(\tau;\theta)d\tau \qquad \text{for} \qquad e(t;\theta) = x_p(t) - x_m(t) \tag{13}$$

with x_p, x_m the respective process and reference model states with the unknown parameters $\{\theta_i\}$

constant over *T*. The parameters are updated based on decreasing the criterion, such that,

$$\theta(t+T) = \theta(t) - \gamma\frac{\partial J}{\partial \theta} = \theta(t) - \gamma \int_{t}^{t+T} \underbrace{e(\tau;\theta)\frac{\partial e(\tau;\theta)}{\partial \theta}d\tau}_{\frac{\partial J}{\partial \theta}} \tag{14}$$

and therefore,

$$\frac{\theta(t+T)-\theta(t)}{T} = -\frac{\gamma}{T}\int_{t}^{t+T} e(\tau;\theta)\frac{\partial x_p(\tau;\theta)}{\partial \theta}d\tau \qquad \text{for} \qquad \frac{\partial e(\tau;\theta)}{\partial \theta} = \frac{\partial x_p(\tau;\theta)}{\partial \theta}$$

As $T \to 0$ in the limit, we have that the instantaneous change in the parameters is given by

$$\frac{d\theta(t)}{dt} = -\gamma \times e(t;\theta) \times \underbrace{\frac{\partial x_p(t;\theta)}{\partial \theta}}_{\text{Sensitivity Derivative}} \tag{8}$$

The sensitivity derivative represents state sensitivity to changes in the parameters; however, using this approach usually called a gradient technique (MIT rule) cannot guarantee stability of the algorithm. Next, we consider an approach that does, the Lyapunov stability algorithm.

B. *Adaptive Algorithm: Lyapunov Stability Algorithm* [7, 8]

Lyapunov's direct method is a well-known method [2, 3, 7] that can be applied to ensure the overall stability an MRAC system and is based on establishing the existence of a function that possesses a prescribed set of properties (positive definite, a negative definite derivative, etc. see [1] for details). Typically, the Lyapunov function,

$$V(e_x, e_\theta) := e_x^T P e_x + e_\theta^T \Gamma^{-1} e_\theta \quad \text{where } \mathrm{e}_x(t;\theta) = x_p(t;\theta) - x_m(t); \text{ and } \mathrm{e}_\theta(t) = \theta(t) - \theta_o(t) \qquad (9)$$

Where both $P, \Gamma > 0$ must be positive definite for V to be viable and $\dot{V}(e_x, e_\theta) < 0$ to guarantee *uniform asymptotic stability* [3, 7]. In general, $\dot{V}$ has the negative definite form if *Q>0*, that is,

$\dot{V} = -e^T Q e + f(\theta)$ terms

and when the parametric terms are set to zero, then the *Malkin theorem* applies for *A*, a stable system matrix, such that *P, Q>0* can *always* be found satisfying

$$AP + PA^T = -Q \qquad \text{[Lyapunov Equation]} \qquad (10)$$

The representation of the Lyapunov MRAC processor design follows the generic adaptive form given by

$$\frac{d\theta(t)}{dt} = -\gamma \times \varepsilon(t) \times \xi(t) \quad \text{for} \quad \varepsilon(t) \Rightarrow f(e(t;\theta)) \quad \text{and} \quad \xi(t) \Rightarrow f(x_p(t;\theta))$$

(11)

where ε is a function of the usual state or output error *e* and ξ is a function of the state/output or their filtered versions. Note that *all* of these techniques incorporate the *product* of the error and state/output. For the gain, γ a constant this form is considered a gradient adaption rule. Also, it should be noted that the usual implementation *normalizes* the Lyapunov adaption rule as

$$\frac{d\theta(t)}{dt} = -\Gamma \times \left(\frac{\varepsilon(t)\xi(t)}{\underbrace{1+\xi^T(t)\xi(t)}_{\text{Normalization}}} \right) \tag{12}$$

where the one ensures not zero-division and the second term constrains the speed of adaption (see [7, 8] for more details. This completes the discussion of the Lyapunov stability approach. Next, we consider a simple first-order system example to illustrate these approaches.

<u>EXAMPLE</u>: Consider the following 1st-order system given by:

* Reference: $\dot{x}_m(t) = -2x_m(t) + 2r(t); \qquad y_m(t) = x_m(t); \qquad \mathrm{x}_m(0) = 0.0$
* Process: $\dot{x}_p(t;\theta) = -x_p(t;\theta) + 0.5u(t;\theta); \quad y_p(t) = x_p(t); \qquad \mathrm{x}_p(0) = 0.0$
* Controller: $\mathrm{u}(t;\theta) = \theta_x x_m(t) + \theta_r r(t)$
* Error: $\mathrm{e}(t;\theta) = x_p(t;\theta) - x_m(t)$
* Adaption: $\theta_x(t) = -2\gamma\left(x_p(t;\theta)\mathrm{e}(t;\theta)\right) \times x_p(t;\theta); \quad \theta_x(0) = 0$ [Lyapunov rule]

  $\theta_r(t) = -2\gamma\left(r(t)\mathrm{e}(t;\theta)\right) \times r(t); \qquad \theta_r(0) = 0$

or

* Adaption: $\theta_x(t+1) = \theta_x(t) - \Delta t \times \gamma \mathrm{e}(t;\theta) \dfrac{\partial e(t;\theta)}{\partial \theta_x}; \quad \theta_x(0) = 0.8$ [MIT rule]

  $$\theta_r(t+1) = \theta_r(t) - \Delta t \times \gamma \mathrm{e}(t;\theta) \frac{\partial e(t;\theta)}{\partial \theta_r}; \quad \theta_r(0) = 0.5$$

  $$\frac{\partial e(t+1;\theta)}{\partial \theta_x} = (1+2\Delta t)\frac{\partial e(t;\theta)}{\partial \theta_x} \underbrace{-2\,\Delta t \times x}_{A_m}{}_p(t;\theta)$$

  $$\frac{\partial e(t+1;\theta)}{\partial \theta_r} = (1+2\Delta t)\frac{\partial e(t;\theta)}{\partial \theta_r} - 2\Delta t \times r(t)$$

The data were simulated for this problem with an input a pulse train and the corresponding outputs. In Fig. 3, the “desired” output is shown (blue) with the “MRAC output” shown (red)---there exists some tracking error (black dots) which could be further minimized further by increasing the *gain* from 200 to 1000 yielding smaller errors on the order of $\sim 10^{-3}$. The control signal $u(t;\theta)$ is shown in green. More details of the controls are shown in Fig. 4 with the control output (blue) and its parameters θ_x and θ_r (green) below.

The gradient adaption algorithm was also applied to this data set with the results shown in Fig. 5. Here the desired and MRAC outputs (blue/red) track with some initial high frequency uncertainty response initially that evolves from the controller (green) and is reflected in the error plot (black) The error is on the same order as the Lyapunov adaption algorithm results. This completes the example problem illustrating the gradient (MIT rule) and stability (Lyapunov) approaches.

It is interesting to note that one of the properties of the MRAC controller is that it must perform "perfect control", that is, matching the controlled output to the desired signal *perfectly*. This constraint is satisfied by equating the desired model with that of the process model and solving for the control. For this 1st-order example, we have

$$\dot{x}_m(t) = -2x_m(t) + 2r(t) \quad \Leftrightarrow \quad \dot{x}_p(t;\theta) = -x_p(t;\theta) + 0.5u(t;\theta)$$

or

$$-2x_m(t) + 2r(t) \;=\; -x_p(t;\theta) + 0.5u(t;\theta)$$

and letting $x_p(t) \Rightarrow x_m(t)$

$$u(t;\theta) = 2 \times \left(2r(t) - x_p(t;\theta)\right)$$

providing the required control to match $x_m(t)$. ΔΔΔ

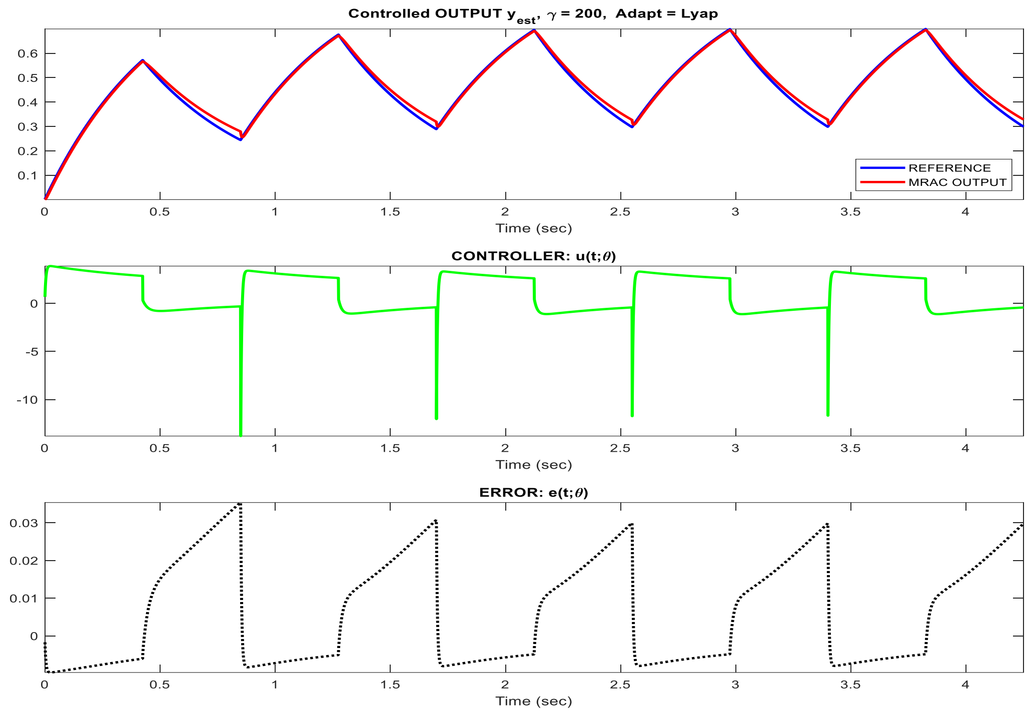

Figure 3. *1st-Order System Results: Lyapunov Adaption with Gain of 200: (a) Reference vs. MRAC Tracking. (b) Controller Output. (c) Tracking Error.*

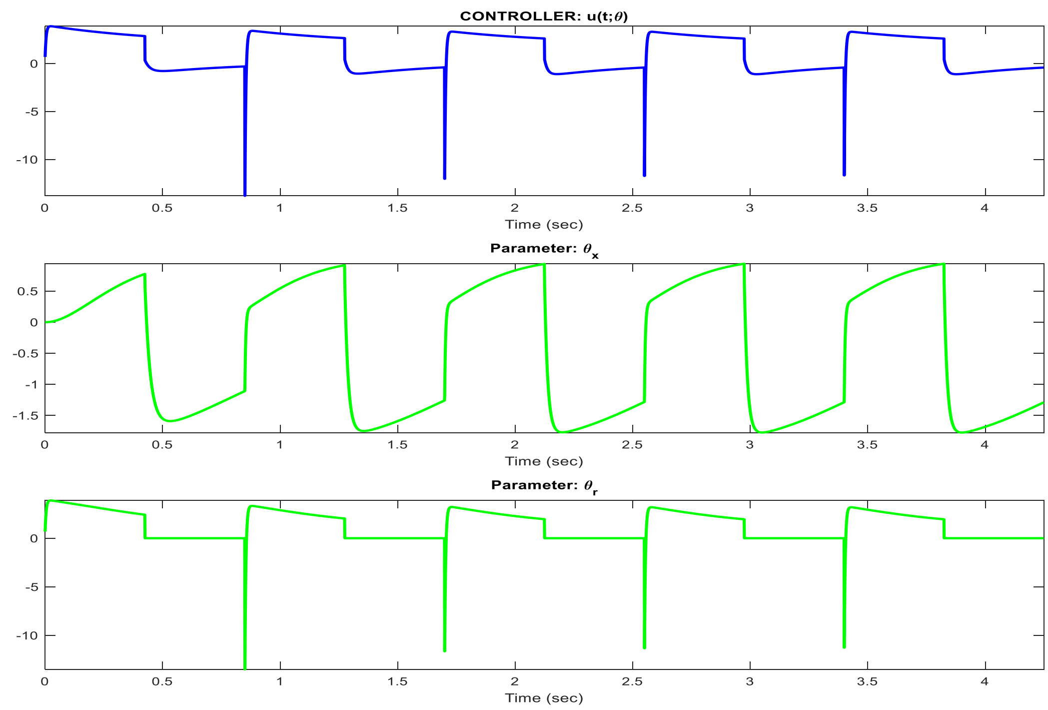

Figure 4. *1st-Order System Results: Lyapunov Adaption with Gain of 200. (a) Controller Output. (b) Control State Parameter. (c) Control Reference Parameter.*

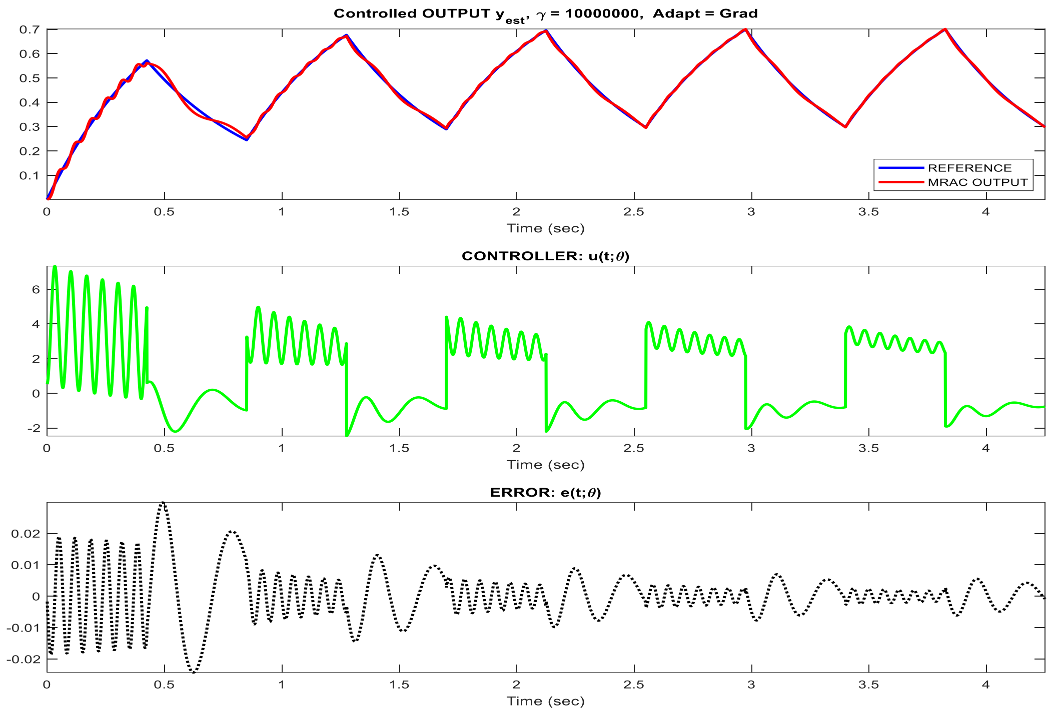

Figure 5. *1st-Order System Results: MIT Gradient Adaption with Gain of* 1×10^7 *: (a) Reference vs. MRAC Tracking. (b) Controller Output. (c) Tracking Error.*

CASE STUDY: Next, we consider the design of a MRAC for a simple MCK mechanical system characterized by *M=1 kg*, *C=0.01 N-sec/m*, *K=1 N/m*, *dt=3 msec*. The formulation of the MRAC is shown below where the reference model represents the “desired” response of the system while the process is the ideal MCK-system with a 5% parametric error ($\Delta\theta = \pm 0.05\bar{\theta}$ for $\bar{\theta}$ nominal) of each value. The simulation of the reference (desired) system for Lyapunov design [7, 8] is shown in Fig. 6 where the pulse input is shown along with the state responses (displacement/velocity) including the accelerometer measurement (noise-free). The design involves solving the algebraic Lyapunov equation followed by the parametric differential equations for the controller parameters that are adaptively adjusted at each time step. The results are illustrated in Fig. 7 where the controller outputs for each states are shown (red) along with the desired responses to be matched or tracked. It is clear from the figure that some bias (DC offset) is prevalent in the controlled displacement, while the velocity appears to track the reference quite well with the exception of some start-up and transition transient disturbances. This is confirmed by the accompanying

* Reference: $\ddot{x}_m(t) = -\left(\frac{c}{m}\right)\dot{x}_m(t) - \left(\frac{k}{m}\right)x_m(t) + \left(\frac{1}{m}\right)r(t)$

$$y_m(t) = \dot{x}_m(t); \qquad \mathrm{x}_m(0) = 0.0$$

* Process: $\ddot{x}_p(t;\theta) = -\left(\frac{c+\Delta c}{m+\Delta m}\right)\dot{x}_p(t;\theta) - \left(\frac{k+\Delta k}{m+\Delta m}\right)x_p(t;\theta) + \left(\frac{0}{m+\Delta m}\right)u(t;\theta)$

$$y_p(t) = \dot{x}_p(t); \qquad \mathrm{x}_p(0) = 0.0$$

* Controller: $u_c(t;\theta) = \theta_x^T x_p(t) + \theta_r r(t)$

* Error: $\mathrm{e}(t;\theta) = x_p(t;\theta) - x_m(t)$

* Adaption: $AP + PA^T = -Q; \quad Q = I$ [Lyapunov Equation]

$$\dot{\theta}_x(t) = -2\gamma\left(B_m^T P\mathrm{e}(t;\theta) \times x_p^T(t;\theta)\right); \qquad \theta_x(0) = 0$$ [Lyapunov rule]

7, $$\dot{\theta}_r(t) = -2\gamma\theta_r(t)B_m^T\left(\mathrm{e}(t;\theta)u_c(t;\theta)\right) \times \theta_r(t); \;\; \theta_r(0) = 0$$

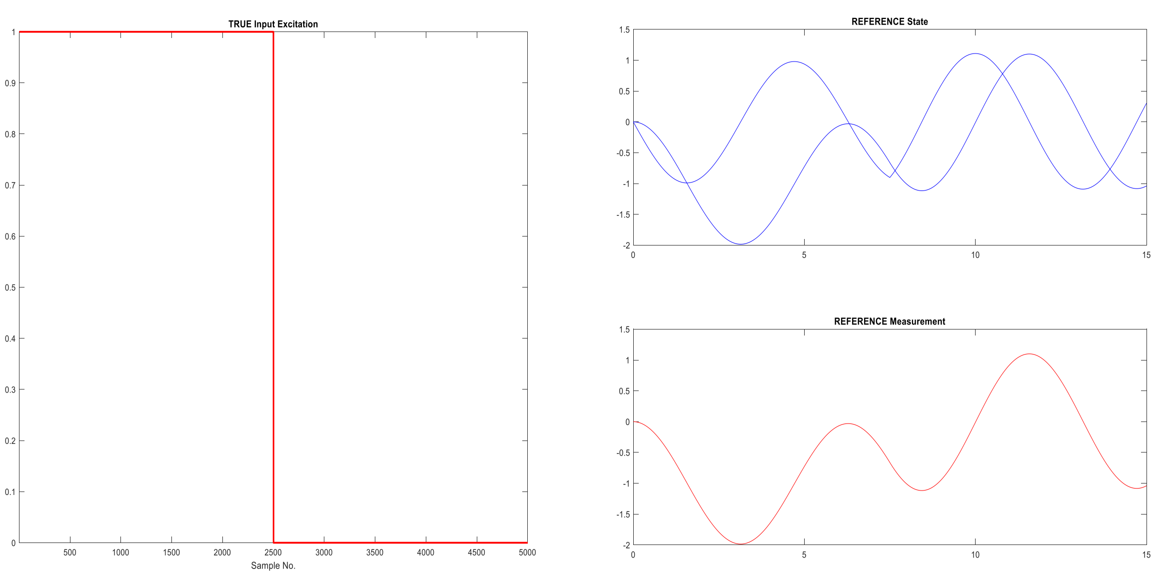

Figure 6. *MCK System Simulation: (a) Pulse Excitation. (b) Response: States and Accelerometer Measurement.*

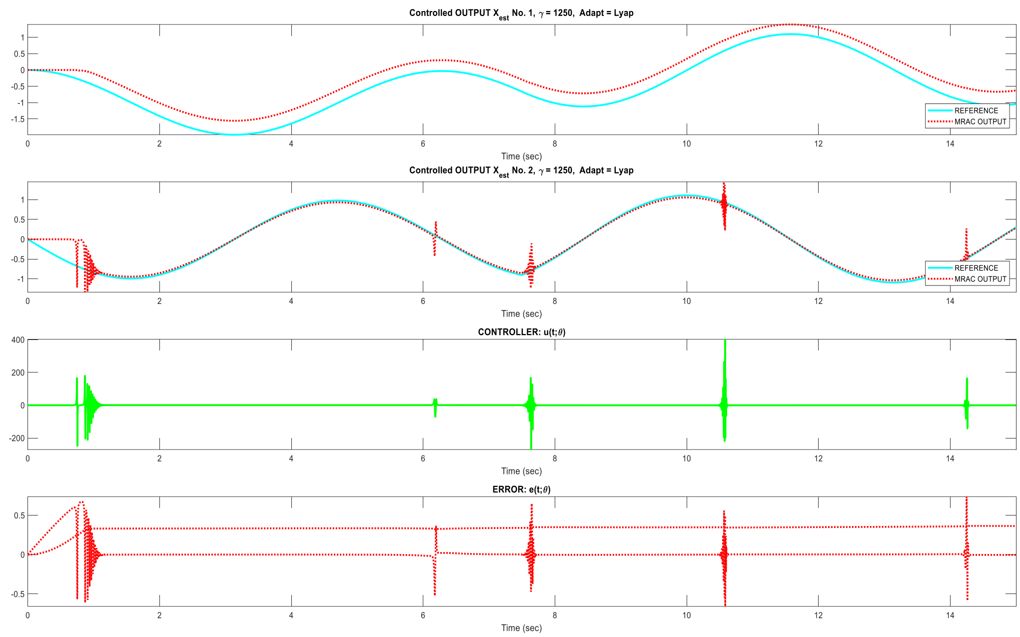

Figure 7. *MCK System Adaptive Control for Lyapunov Design: (a) State No. 1: Displacement (Reference (cyan) and Controlled (red)). (b) State No. 2: Velocity (Reference (cyan) and Controlled (red)). (c) Controller output with transient artifacts. (d) State Errors.*

controller response (transients) and the corresponding state error plots. The controller and controller parameter trajectories are shown in Fig. 8. It is clear from these plots that the transient disturbance occurs during the abrupt parametric changes. This could be caused by the 1st-difference approximation to the parametric differential equation solutions and might be improved using a numerical integration technique instead. This completes the case study. ΔΔΔ

Next, we consider the potential applicability to controlling the additive manufacturing (AM) process.

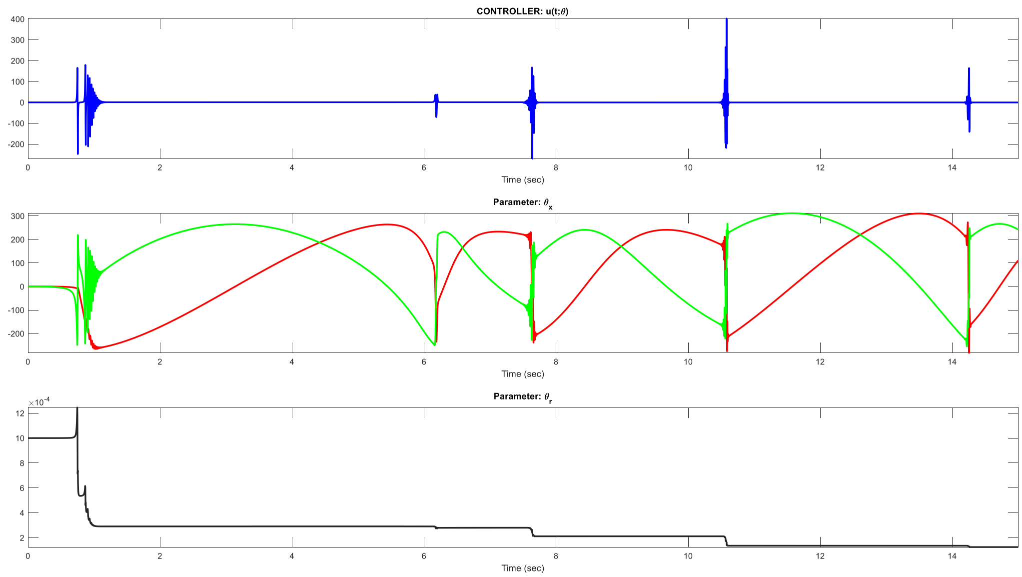

Figure 8. *MCK System Adaptive Control for Lyapunov Design: (a) Controller output with transient artifacts. (b) State Parameters: Displacement (red) and Velocity (green). (c) State Parameters: Reference (black).*

V. *Model-Reference Adaptive Control for Additive Manufacturing Application*

In this section we briefly discuss the *possibility* of applying these techniques to the AM problem. We show how the problem may be formulated within the MRAC framework for potential solution. As discussed previously, the following components are required to develop MRAC: reference model, process data (model), controller and an adaption algorithm. Clearly, the AM problem possesses a variety of components that might be applied to this construction problem.

For instance, consider the "welding" process inherent in AM does possess the components that can be used to construct a viable controller. The *weld path* available prior to the so-called "build" provides a reference trajectory or "desired" response of the component or part construction procedure. The process to be "controlled" is the powder material, while the controller is the laser welder, itself, that has its own inherent controls based on thermal material properties and temperature. The adaption algorithm follows the theory outlined in this report and could provide a viable solution. To see how this all might fit together in an MRAC scheme, consider Fig. 9 for this welding problem.

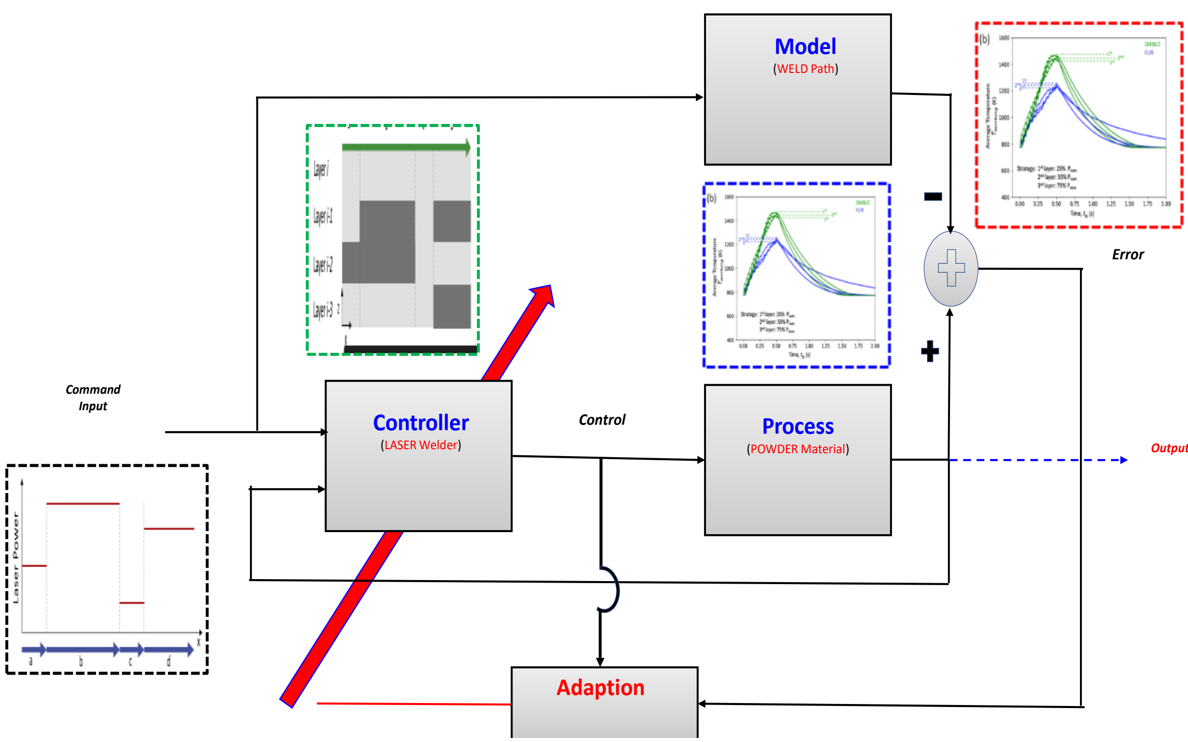

Figure 9. *Potential MRAC Solution to the Additive Manufacturing (AM) Problem: Command input trajectory, controller (laser welder), reference (desired weld path) process (powder) and adaption algorithm (temperature control).*

Based on the published papers and presentations of the LLNL AM group, it appears that all of the ingredients are available to attach this problem using MRAC [14, 15, 16] for the following reasons:

- REFERENCE model simulations (powder, microstructure, kinetics) are readily available.
- SENSORS are available (FLIR, photodiodes, acoustic emission microphones) that could potentially be applied on-line.
- CONTROLLERS exist in a variety of AM applications (laser welders, etc.).
- ADAPTION algorithms are available from the control literature that may be able to facilitate AM solutions.

The MRAC framework could potentially provide a viable solution to a multitude of AM-related problems of high interest.

VI. SUMMARY

It appears that the MRAC approach to control can possibly provide a potential solution to a variety of sub-processes within the AM problem. This approach enables the design capability for adaptive controller design, if a reference model is available along with measurement sensors capable of responding in near real-time. It is may be possible to perform on-line flaw detection during the construction process.

REFERENCES,

[1] K. Astrom and B. Wittenmark, *Computer Controlled Systems: Theory and Design*, Prentice-Hall, Englewood Cliffs, N. J., 1984.

[2] K. Astrom and B. Wittenmark, *Adaptive Control*, Addison-Wesley, Reading, MA., 1989.

[3] Y. Landau, *Adaptive Control: The Model Reference Approach*, Marcel Dekker, New York, N. Y., 1979.

[4] Y. Landau, "An extension of stability theorem applicable to adaptive control," *IEEE Trans. Auto. Control*, AC-25, pp. 593-611, 1981.

[5] Y. Landau and R. Lozano, "Unification of discrete-time explicit model reference adaptive control designs," *Automatica*, 17, pp. 814-817, 1980.

[6] K. Narenda and A. Annaswamy, *Stable Adaptive Systems*, Prentice-Hall, Englewood Cliffs, N. J., 1989.

[7] N. Hovakimyan and C. Chao, *L1 Adaptive Control Theory*, SIAM, Phila. PA, 2010.

[8] H. Butler, *Model Reference Adaptive Control: From Theory to Practice*, Prentice-Hall International, N.J., 1992.

[9] J. Candy, *Model-Based Signal Processing*, J. Wiley Inc., N.J., 2006.

[10] L. Ljung, *System Identification: Theory for the User*, 2^{nd}-Ed, Prentice-Hall N.J., 1999.

[11] J. Candy, *Model-Based Processing: An Applied Subspace Identification Approach*, J. Wiley Inc., N.J., 2019.

[12] P. DeRusso, R. Roy, C. Close and A. Desrochers, *State Variables for Engineers*, 2^{nd} Ed., J. Wiley-Interscience, 1998.

[13] J. D'Azzo, C. Houpis and S. Sheldon, *Linear Control System Analysis and Design with MATLAB*, 5^{th} Ed., M. Decker, N.Y., 2003.

[14] C. Druzgalski, A. Ashby, G. Guss, W. King, T. Roehling and M. Matthews, "Process optimization of complex geometries using feed forward control for laser powder bed fusion additive manufacturing," *Addit. Manuf.*, 34, pp. 1-11, 2020.

[15] J-B. Forien, N. Calta, P. DePond, G. Guss, T. Roehling and M. Matthews, "Detecting keyhole pore defects and monitoring process signatures during laser powder bed fusion: A correlation between in situ pyrometry and ex situ X-ray radiography," *Addit. Manuf.*, 35, pp. 1-10, 2020.

[16] K. Fisher, J. Candy, Gabe Guss and M. Matthews, "Evaluating acoustic emission signals as an in situ process monitoring technique for selective laser melting (SLM)," *LLNL Report*, LLNL-TB-706659, 2016.