Simulation of a Linear Quadratic Regulator Controller for sCO_2 Cycle Using MARS-KS

Gi Hyeon Kim[a], Jeong Ik Lee[a*]
[a]*Dept. Nuclear & Quantum Eng., KAIST, 373-1, Guseong-dong, Yuseong-gu, Daejeon, 305-701, Republic of Korea*
[*]*Corresponding author: jeongiklee@kaist.ac.kr*

1. Introduction

The supercritical CO_2 (sCO_2) system is a power conversion system that utilizes CO_2 at the supercritical phase as a working fluid. As sCO_2 is compressed near the critical point, the pressure can be increased effectively with little work input. This allows an sCO_2 system to have smaller turbomachinery and higher efficiency than conventional steam Rankine or air Bryton cycles [1]. This small size with high-efficiency characteristics makes the sCO_2 system attractive for use as a power conversion system for a distributed power source [2].

The distributed power source is a concept of small and medium-sized power generation systems placed nearby consumers of electricity to increase system stability, lower initial expenditures, and cut down on the number of transmission infrastructures. To meet local energy needs, distributed power sources must generate sufficient electricity to fulfill the local needs. This load-following generation scheme can be accomplished by known control strategies, such as turbine and core bypass controls, inventory tank control, and compressor recirculation control. Problems with these control methods are that the conditions at the turbine outlet affect the compressor inlet depending on the precooler operating conditions.

In the case of a steam Rankine cycle, steam at the turbine outlet is cooled with a sufficient amount of water, and in an air Brayton cycle, outside air is injected, so the condition of the compressor inlet is unconnected with the physical condition of the turbine outlet. However, in an sCO_2 Brayton cycle, a precooler must remove an appropriate amount of heat to stabilize the compressor inlet condition.

A sCO_2's thermodynamic property that affects compression, such as compressibility factor and density, varies dramatically near the critical point. Meanwhile, the compressor inlet condition of an sCO_2 Brayton cycle is close to the critical point. Therefore, the compressor inlet conditions must be controlled to maintain compressor efficiency and stability [3]. Since this is primarily a matter of how much heat is removed by a precooler at the turbine outlet, it can be achieved by controlling the coolant flowrate to the precooler.

LQR is an abbreviation of Linear Quadratic Regulator and is a control method that minimizes the cost function for a given system. A regulator is designed to minimize the cost by representing the dynamics of the system linearly through the state space of the plant and expressing the cost as a quadratic equation. The control variables of this controller are calculated mathematically with the Riccati equation, which optimizes the cost function for arbitrary weights [4]. The reason why LQR control was chosen in this study is because of several advantages. First of all, the system controlled by LQR controller is always stable. In addition, unlike PID controller, when designing LQR controller, the number of variables to be tuned is reduced to one, which enables to design a more consistent controller. Finally, an optimal controller that does not require additional tuning is obtained by simply solving the LQ optimal control problem for arbitrary weights.

In this study, a precooler system is modeled using MARS-KS code and verified using experimental data obtained from the Autonomous Brayton Cycle test loop (ABC test loop) compressor surge control test conducted at KAIST [5]. The system identification of the precooler system was conducted using a step input signal to obtain a system transfer function. The linear quadratic regulator (LQR) based controller is designed from the obtained system transfer function and using state space. The designed LQR controller was implemented using MARS-KS code and evaluated by changing the precooler sCO_2 inlet condition.

2. Methods and Results

This section describes the design procedure and simulation results of LQR controllers. The following design procedure includes system modeling, system identification, controller design, and implementing the LQR controller using MARS-KS.

2.1 System Modeling Using MARS-KS

The ABC test loop was designed for the integrated experiment of a simple recuperated sCO_2 cycle. Baek modeled the entire ABC test loop using the MARS code and then verified it using the data of the compressor surge control test [5]. In this study, it is sufficient to have only the model of the precooler part among the entire ABC test loop system. To reduce calculation time, the precooler system of the ABC test loop was modeled using a node structure shown in Fig. 1, and the results are verified with the experimental data. The result is shown in Fig. 2.

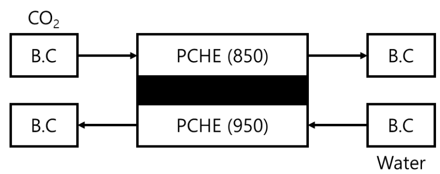

Fig. 1 Node structure of precooler system

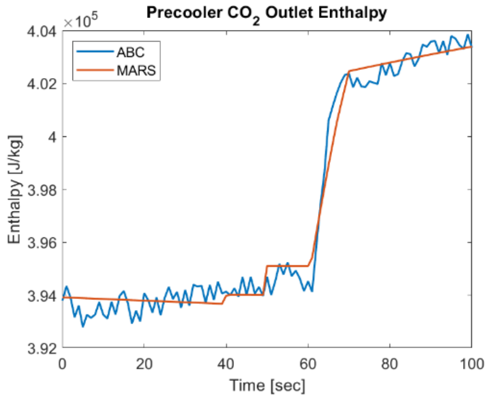

Fig. 2 Precooler CO2 outlet enthalpy comparison of ABC test loop experiment result and MARS-KS code result

2.2 System Identification

At the design point, CO_2 enters the precooler at 321.74K, 8.6MPa, and exits at 308.15K, 7.6MPa. Also, throughout the experiment, the intake water conditions were preserved at 298.15 K and 1 bar. The response of the system was simulated using the MARS code when the water flowrate was doubled while CO_2 side inlet condition was fixed to the on-design settings to determine the transfer function of a precooler system.

The process variable of the precooler system is CO_2 outlet enthalpy, while a control input is water flowrate. The original input and the output signal, $u_0(k)$ and $y_0(k)$ respectively, were normalized using Equation (1) to make the input signal u(k) a unit step input and the output signal y(k) to start at zero.

$$u(k) = \frac{u_0(k)}{u_{0,min}} - 1$$
$$y(k) = 1 - \frac{y_0(k)}{y_{0,max}} \quad \cdots (1)$$

Normalized system I/O data was measured via MARS code as shown in Fig. 3.

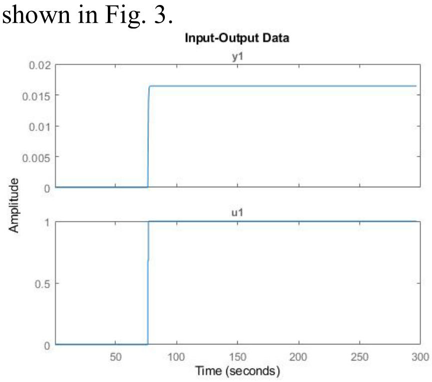

Fig. 3 System response for a unit step input

Using the least square method on data shown in Fig. 3, the transfer function $\tilde{G}(z)$ is approximated as shown in Equation (2).

$$\frac{\mathcal{Z}\{y(k)\}}{\mathcal{Z}\{u(k)\}} = \frac{Y(z)}{U(z)} = \tilde{G}(z)$$
$$= \frac{0.02004\,z + 0.001064}{z^2 + 0.3433\,z - 0.05896} \cdots (2)$$

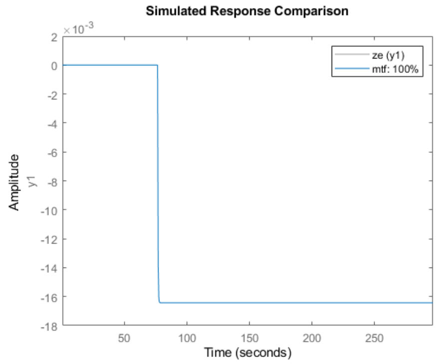

Fig. 4 System response comparison for a unit step input

Fig.4 shows system responses between the transfer function and the MARS simulation for a unit step input. It shows that it is possible to approximate the precooler system with the transfer function $\tilde{G}(z)$.

For the off-design condition, where the inlet CO_2 condition differs from the design point, the transfer function $\tilde{G}(z)$ obtained from the design point needs to be modified. The response of the system of off-design points calculated from MARS simulation and transfer function $\tilde{G}(z)$ without modification is shown in Fig. 5.

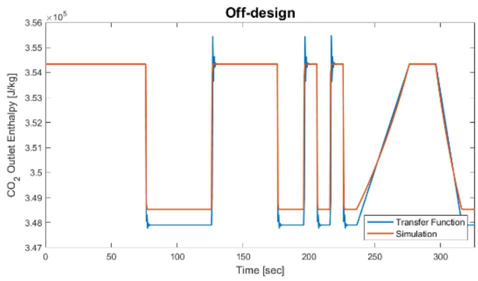

Fig. 5. Off-design performance approximation

The transfer function and MARS code show similar trends but amplitudes are different. Thus, transfer function $\tilde{G}(z)$ can approximate plant dynamics at the off-design condition by multiplying an appropriate scale parameter C_f as shown in Equation (3).

$$\tilde{G}(z) = C_f \frac{0.02004\,z + 0.001064}{z^2 + 0.3433\,z - 0.05896} \cdots (3)$$

Fig.6 shows the response of transfer function $\tilde{G}(z)$ with scale parameter C_f.

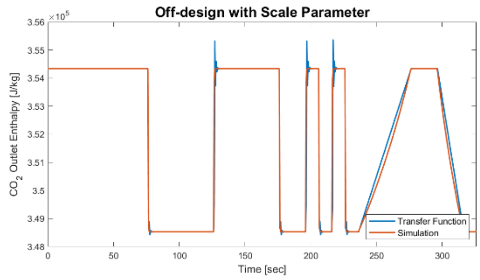

Fig. 6. Off-design performance approximation with scale parameter ($C_f = 1.283$)

2.3 Linear Quadratic Regulator Design

To design an LQR controller, the transfer function needs to be converted for state space. The state space realization of transfer function $\tilde{G}(z)$ is shown in Equation (4), using the observable canonical form.

$$\begin{cases} \boldsymbol{x}(k+1) = \boldsymbol{A}\boldsymbol{x}(k) + \boldsymbol{B}\boldsymbol{u}(k) \\ y(k) = \boldsymbol{C}\boldsymbol{x}(k) + Du(k) \end{cases} \cdots (4)$$

where,

$$\begin{cases} \boldsymbol{A} = \begin{bmatrix} -0.3433 & 1 \\ 0.05896 & 0 \end{bmatrix} \\ \boldsymbol{B} = \begin{bmatrix} 0.02004 \\ 0.001064 \end{bmatrix} C_f \\ \boldsymbol{C} = \begin{bmatrix} 1 & 0 \end{bmatrix} \\ D = 0 \end{cases}$$

It is well known that a full-state feedback controller gain that minimizes the cost function can be calculated by solving LQR equations. However, the state space of Equation (4) is just a model obtained from the simulated data. Thus, since Equation (4) does not represent the whole dynamics of an actual precooler, the controller gain calculated from Equation (4) cannot be directly utilized for controlling the whole system. Therefore, the LQR controller needs to operate for a given precooler by adding a discrete-time full-state observer as shown in Fig. 7.

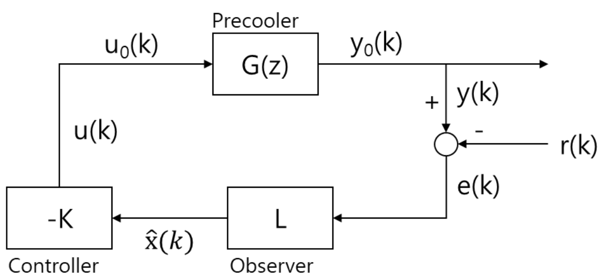

Fig. 7. LQR controller with full-state observer

For the state space equation in Equation (4), the full-state observer gains matrix **L**, setting the observer's eigenvalue as [0, 0] satisfied Equation (5) [6].

$$\det[\mathrm{A} - \mathrm{LC}] = 0 \cdots (5)$$

$$\boldsymbol{L} = \begin{bmatrix} -0.3433 \\ 0.05896 \end{bmatrix}$$

The optimal LQR controller gains **K** that minimizes the performance index can be calculated by solving the following Equation (6), which is known as the discrete-time algebraic Riccati equation (DARE) [7]. The R is an arbitrary positive definite matrix.

$$\begin{aligned} &\mathbf{K} = [\boldsymbol{R} + \mathbf{B}^{\boldsymbol{T}}\boldsymbol{P}\mathbf{B}]^{-1}\mathbf{B}^{\boldsymbol{T}}\boldsymbol{P}\boldsymbol{A} \\ &\boldsymbol{P} = \mathbf{A}^{\mathrm{T}}\boldsymbol{P}\boldsymbol{A} + \mathbf{Q} - \boldsymbol{A}^{\mathrm{T}}\boldsymbol{P}\mathbf{B}[\boldsymbol{R} + \mathbf{B}^{\boldsymbol{T}}\boldsymbol{P}\mathbf{B}]^{-1}\mathbf{B}^{\boldsymbol{T}}\boldsymbol{P}\boldsymbol{A} \cdots (6) \end{aligned}$$

With Equation (4) and Equation (6), if the value of **B** at the design point is called $\mathbf{B_0}$ and the value of the controller gain **K** is called $\mathbf{K_0}$, then the relationship shown in Equation (7) holds for an off-design point with arbitrary C_f in the case of $\mathbf{R} = \mathbf{0}$.

$$\begin{aligned} \mathbf{B} &= C_f\mathbf{B_0} = C_f \begin{bmatrix} 0.02004 \\ 0.001064 \end{bmatrix} \\ \mathbf{K} &= [\boldsymbol{R} + \mathbf{B}^{\boldsymbol{T}}\boldsymbol{P}\mathbf{B}]^{-1}\mathbf{B}^{\boldsymbol{T}}\boldsymbol{P}\boldsymbol{A} = \left[C_f^2\mathbf{B}^{\boldsymbol{T}}\boldsymbol{P}\mathbf{B}\right]^{-1}C_f\mathbf{B}^{\boldsymbol{T}}\boldsymbol{P}\boldsymbol{A} \\ &= C_f^{-1}[\mathbf{B}^{\boldsymbol{T}}\boldsymbol{P}\mathbf{B}]^{-1}\mathbf{B}^{\boldsymbol{T}}\boldsymbol{P}\boldsymbol{A} = C_f^{-1}\mathbf{K_0} \\ &= C_f^{-1}[-17.128 \quad 49.899] \cdots (7) \end{aligned}$$

By using the observer gain matrix **L** from Equation (5), the LQR control gains **K** from Equations (6) and (7), and the state space equation shown in Equation (4), the observed state $\hat{\mathbf{x}}$ of Fig.5 with the full-state observer is calculated as shown in Equation (8).

$$\begin{aligned} &\hat{\boldsymbol{x}}(k+1) = (\boldsymbol{A} - \boldsymbol{LC} - \boldsymbol{BK})\hat{\boldsymbol{x}}(k) - \boldsymbol{L}e(k) \\ &= \left(\boldsymbol{A} - \boldsymbol{LC} - C_f\mathbf{B_0}C_f^{-1}\mathbf{K_0}\right)\hat{\boldsymbol{x}}(k) - \boldsymbol{L}e(k) \\ &= (\boldsymbol{A} - \boldsymbol{LC} - \mathbf{B_0}\mathbf{K_0})\hat{\boldsymbol{x}}(k) - \boldsymbol{L}e(k) \\ &= \begin{bmatrix} 0.3433 & 0 \\ 0.01822 & -0.05309 \end{bmatrix}\hat{\boldsymbol{x}}(k) - \begin{bmatrix} -0.3433 \\ 0.05896 \end{bmatrix} e(k) \quad (8) \end{aligned}$$

The full state feedback controller gains **K** and the observed state $\hat{\mathbf{x}}$ is calculated from Equation (3) to Equation (8). With the obtained **K** and $\hat{\mathbf{x}}$, the normalized control input u is calculated as $\mathrm{u} = -\mathbf{K}\hat{\mathbf{x}} = -C_f^{-1}\mathbf{K_0}\hat{\mathbf{x}}$. As $\mathbf{K_0}$ is known value and the state $\hat{\mathbf{x}}$ is obtained from MARS code calculation and using Equation (8), the scale factor C_f is the only term that needs to be pre-calculated before the control. The actual input of water flow to the precooler system is calculated by taking the normalized control input which is the inverse of Equation (1).

2.4 Simulating LQR Controller Using MARS-KS

A test scenario was selected to implement the LQR controller in the MARS code and evaluate its performance. The CO_2 temperature was changed from the design point of 321.74K to 331.74K while the pressure and flowrate of CO_2 at the inlet of precooler were kept constant at the design point. Three types of change in enthalpy were tested: ramp increase, step increase, and ramp decrease. The control goal of the LQR controller is to adjust water flowrate entering the precooler to match the enthalpy of CO_2 exiting the precooler to the design point.

Equation (8) and scale factor C_f are the functions that need to be implemented in the MARS code. For Equation (8), matrix multiplication is needed. Although matrix calculation is not supported in the MARS code, if only the matrix multiplication of Equation (8) is solved in advance, the rest can be calculated for each step using a control card of the MARS code. To calculate the scale factor C_f, C_f was calculated for three points: 321.74K, 326.74K, and 331.74K. Polynomial interpolation of C_f at these three points predicted C_f with respect to inlet enthalpy in the entire range of scenarios.

Figure 8 shows the inlet enthalpy and the outlet enthalpy of precooler system to which the LQR controller was applied and simulated with MARS. As a result of controlling the enthalpy of the precooler CO_2 outlet by adjusting water flowrate to the precooler with LQR controller, it was confirmed that the control was possible in a scenario with a wide range and various types of input enthalpy change rates. As shown in Fig. 8, when the input was changed, the maximum error in the entire simulation was only 0.21%.

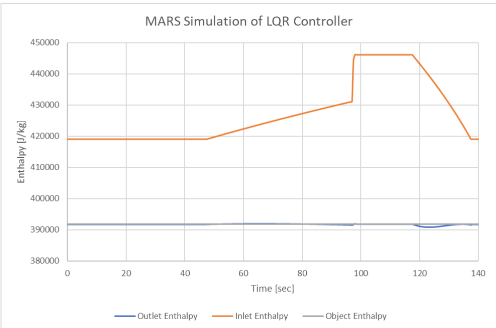

Fig. 8. CO_2 inlet enthalpy and outlet enthalpy of LQR-controlled precooler system

3. Conclusions

In this study, an LQR controller for the sCO_2 Brayton cycle precooler system was designed using data obtained from the ABC test loop. First, the MARS code was verified with experimental data, and the response of system to unit step input was simulated to obtain the transfer function. It was shown that this transfer function well predicted the open-loop response of precooler system under design conditions. In addition, it was shown that even when the precooler system is operating in off-design conditions, it can still be used by simply multiplying the transfer function obtained for the design condition with an appropriate scalar value. From this result, the LQR controller was designed to be used for the control of an entire range, not only for the design condition. Lastly, the LQR controller designed in the MARS code was implemented. By applying this to the sCO_2 precooler system and performing computational analysis with the MARS code, it was shown that the designed controller controlled well with an error of less than 0.2%.

This study suggests that state space-based controller design and simulation are possible using the MARS code for sCO_2 system. Further research works can be identified from the result. First of all, it is possible to design and simulate other state space-based controllers with MARS code as well as with LQR controller. Next, it is necessary to study whether the controller designed from the MARS code operates as expected in real operation. If the controller designed from the MARS code works well in real condition, using system analysis code to develop a controller can become more widely accepted methodology.

ACKNOWLEDGEMENT

This research was supported by the Challengeable Future Defense Technology Research and Development Program(No.912767601) of Agency for Defense Development in 2023.

REFERENCES

[1] Dostal, V., Driscoll, M. J., & Hejzlar, P. (2004). A supercritical carbon dioxide cycle for next generation nuclear reactors (Doctoral dissertation, Massachusetts Institute of Technology, Department of Nuclear Engineering).

[2] Ahn, Yoonhan, et al. "Review of supercritical CO2 power cycle technology and current status of research and development." Nuclear Engineering and Technology 47.6 (2015): 647-661.

[3] Neveu, J. D., Cich, S. D., Moore, J. J., & Mortzheim, J. (2021, June). Operation and Control of a Supercritical CO2 Compressor. In Turbo Expo: Power for Land, Sea, and Air (Vol. 85048, p. V010T30A017). American Society of Mechanical Engineers.

[4] Mehrmann, V. L. (Ed.). (1991). The autonomous linear quadratic control problem: theory and numerical solution. Berlin, Heidelberg: Springer Berlin Heidelberg.

[5] Baek, J. Y., Lee, J. J., & Lee, J. I (2022, May). System Modeling of KAIST S-CO2 ABC Test Loop. In KNS spring meeting.

[6] Luenberger, D. (1971). An introduction to observers. IEEE Transactions on automatic control, 16(6), 596-602.

[7] Pappas, T., Laub, A., & Sandell, N. (1980). On the numerical solution of the discrete-time algebraic Riccati equation. IEEE Transactions on Automatic Control, 25(4), 631-641.