**Enhancing Human Operators Trust in MPC Controllers Through an Explainable AI Methodology**

Anurag Pathak*a* , Babji Srinivasan*b,c* , Rajagopalan Srinivasan*a,c*

*a Department of Chemical Engineering, Indian Institute of Technology Madras, Chennai, 600036, India*

*b Department of Applied Mechanics and Biomedical Engineering, Indian Institute of Technology Madras, Chennai, 600036, India*

*c American Express Lab for Data Analytics, Risk and Technology, Indian Institute of Technology Madras, Chennai, 600036, India*

[raj@iitm.ac.in](mailto:raj@iitm.ac.in) , [babji.srinivasan@iitm.ac.in](mailto:babji.srinivasan@iitm.ac.in)

Abstract  
Model predictive control (MPC) utilizes a system model to obtain optimal control sequences and takes action that improves plant efficacy. However, the successful deployment of MPC in industrial plants requires that plant operators trust them. In complex, real-life, tightly integrated processes, the MPC’s inherent complexity and the lack of interpretability of its actions, especially for a non-expert, can lead to a loss of trust, even when the MPC operates as intended. We seek to address this gap in this work. Specifically, we introduce the concepts of explainable AI (XAI) to understand MPCs actions in real-time. Specifically, we use the well-known SHapley Additive exPlanations (SHAP) to provide to the operator with the subset of variables that contribute to a particular MPC action. The effectiveness of the proposed method is demonstrated on a nonlinear Continuous Stirred Tank Reactor (CSTR). We show that the results from SHAP align with the operator’s understanding of the process causality and thus aid the operator in interpreting MPC’s actions.

Keywords: Model Predictive Control, Explainability, SHAP, Machine learning, Attributions

* 1. Introduction

Process industries are composed of many interconnected processes with multiple interacting units, and it is necessary to maintain control over all these units to optimize production and increase plant efficiency. To achieve this, Model Predictive Control (MPC) is employed to provide superior control for multivariable systems. MPC operates as a control paradigm by utilizing a prediction model of the controlled system. It generates optimized control sequences, taking into account both system constraints and interdependencies among process variables. Consequently, it is widely recognized as the industry standard for managing large-scale processes, as it often takes actions that improve plant performance beyond what a skilled and experienced operator can achieve.  
However, a key challenge for MPC is the interaction between the operator and the optimization techniques. The evolution of advanced automation has led to a transformation in the role of operators, shifting towards monitoring and supervision, thereby placing them out of the loop with the system and making situation awareness challenging. The complex algorithmic framework of MPC may not synchronize with the familiar, intuitive single-loop control mechanism operators typically use (Lindscheid et al.,2016). This makes MPC a black box model for the operator, which lacks interpretability. Therefore, monitoring, diagnosing, and controlling MPC is a challenging task for the operator, and there are numerous instances where plant operators may deactivate MPC when faced with difficulties in understanding its predictions, resulting in diminished trust in the system’s control actions. The significance of human factors is often overlooked or disregarded and plays a pivotal role in the effective functioning of MPC solutions (Forbes et al., 2015). To tackle these trust and comprehension challenges, we propose a methodology that offers explainability of the MPC actions to the operators, thus bridging the gap between human intuition and automated systems.

Explainability refers to insights behind the output of a model. When model explanations are available through that we can establish an intuitive relationship between the input and output. Various studies have been done regarding explainability in the context of process monitoring applications. Bhakte et al. (2022) applied an Integrated Gradient (IG) approach to interpret the fault detection results obtained from a deep neural network. IG is a gradient-based attribution method based on the Shapely value concept from game theory. The effectiveness of any XAI method in a chemical process relies on the ability to establish a meaningful correlation between attributions and the chemical process. If such a correlation can be established, it enhances model interpretability. Bhakte et al. (2023) utilized an XAI method that provides domain-specific explanations by providing important variables with a supervised variable range that causes faults in the system. They also explain why it is preferable to have local explanations (explains the result for each individual sample) than a global explanation (explains the entire AI model) in process monitoring applications. XAI methods can be classified as transparent models and post-hoc techniques. Transparent models are self-interpretable, whereas post-hoc techniques generate explanations by extracting relevant information from a complex, already trained model. Post hoc explainability can be provided in various ways, and among all of those, feature relevance explanations are widely used because sensor data (tabular) is predominant in the chemical industry.

In this paper, we demonstrate how XAI methods can be used to provide insights into MPC and enhance human operator trust in its control action. Section 2 provides a comprehensive explanation of the proposed methodology. Section 3 presents the outcomes of this methodology applied to a basic chemical process. Lastly, the concluding section 4 concisely summarizes the observed outcomes and suggests potential future research directions.   
  
**2. Methodology**In this study, we have proposed an approach to understand MPC’s actions using the XAI technique. To implement this XAI technique on machine learning-based models, an MPC controller is initially designed and simulated for a series of changes in the input variables. Its output is recorded, and the data accumulated is then used to train various machine learning models like- XGBoost regressor, Random Forest regressor, and Neural Network (NN) Model. Empirical validation on a distinct dataset substantiates the efficacy of these models in approximating the MPC’s predictions. Given a trained machine learning model, denoted as *fx*, the interpretation of its predictions can be addressed through a cohesive framework known as SHAP (SHapley Additive exPlanations) (Lundberg ,2017). SHAP provides explanations in the form of feature importance, calculated as the Shapley value from coalition game theory. In this context, features are conceptualized as players in a game, capable of being either present or absent for a specific prediction. This study uses KernelSHAP, a model agnostic approximation technique, to calculate Shapley values for feature attribution.

Firstly, all the possible combinations of coalitions are generated, and these coalitions have entries in the form of 1’s and 0’s. The length of each coalition is equal to the number of features. These coalitions are then superimposed on the input features, and the features masked with ‘1’ preserve their original values, while those masked with ‘0’ have their values replaced with random values from the corresponding features in the background data, thereby generating hypothetical samples. This process is iterated multiple times to minimize the bias the masking feature introduces, thereby generating numerous hypothetical samples for a particular coalition. These samples are evaluated on the black box model, and its output is averaged to get the coalition’s output values. Then, a linear model is fitted to the pairs of coalition-output values, revealing the influence of each feature’s presence or absence on the output.   
This study focuses on explaining machine learning models trained on MPC data that successfully replicate the actions taken by MPC, assuming single control input *uk* where *k* denotes a particular instance to be explained in the dataset. Next, we describe the procedure for calculating SHAP values for a specific input-output pair (*xk, uk)* as follows:   
1) Generating *‘m’* coalitions with the help of binary masking features zʹ*m* ϵ {0,1}, where is the number of input features, tocreate an entire coalition set M. 2)For each of the coalition zʹ*m*: Calculate the weight of every coalition with the help of the SHAP kernel

, ( 1 )

where denotes the number of present features (1’s) in zʹ*m.* To get the prediction of these coalition sets using *fx,* we pass it through a one-to-one mapping function *hx,* such  
 that *xk*,q where q is randomly sampled from the background data. The optimal control output of the mapped feature is calculated by passing it through the model *fx*: *uzʹm* *= f* (*hx* (zʹ*m*)). The above process is repeated *l* times, and the output is averaged to get the coalition’s output values.

( 2 )

3) Fitting linear model (·):

( 3 )

by minimizing the sum of squares loss (4) over the coalition set

| **Table 1: Parameters for the CSTR System** | | |
| --- | --- | --- |
| **Parameter** | **Value** | **Unit** |
| F | 1 | m3/h |
| V | 1 | m3 |
| R | 1.985875 | kcal/(kmol·K) |
| ΔH | -5,960 | kcal/kmol |
| E | 11,843 | kcal/kmol |
| k0 | 34,930,800 | 1/h |
| ρCp | 500 | kcal/(m3·K) |
| UA | 150 | kcal/(K·h) |

( 4 )

The solution of the above optimization formulation would be:   
 where , , ( 5 )

Implementing the proposed approach on an MPC-based machine learning model involves using background data as a baseline, which represents the normal operating state of the process to which MPC is applied. This ensures that when binary masking arrays are imposed on the input feature, those masked with ‘0’ have their values substituted from the baseline data, and the generated hypothetical samples would then represent the baseline. The model prediction for these samples aligns with the normal operating state since represent the expected value of all model predictions. Consequently, from (5), the attribution for that particular input-output pair would be zero.   
The SHAP values acquired through the above formulation signify that when the model prediction differs from the background data, the contribution of each variable to this deviation is indicated by their corresponding SHAP values. The higher these values, the higher their impact on the model prediction and vice versa. Applying this methodology to explain MPC’s actions would reveal how each input variable to the MPC model affects its output and its level of impact. The results of the implementation of the above methodology are shown in the subsequent section.   
  
**3.** **Case Study – Linear MPC for an Exothermic Chemical Reactor**

An adiabatic CSTR in which a single first-order exothermic and irreversible reaction A→B takes place in the vessel. The system is assumed to be perfectly mixed. Parameter values for the CSTR is shown in Table 1.As the reaction is highly exothermic, the excess heat generated should be removed from the CSTR to maintain the residual concentration. Therefore, the control objective is to maintain the residual concentration, C*A*, at its setpoint by adjusting the jacket coolant temperature, T*c*. Changes in the feed concentration, C*f,* and feed temperature T*f* causes disturbances in the CSTR system.For the above purpose, a linear MPC is designed using the MPC toolbox in

|  |  |
| --- | --- |
| (a) | (c) |
| (b) | (d) |
| Fig. 1. Plant Input: (a)Jacket Coolant Temperature, (b) Inlet feed Concentration, (c) Inlet feed Temperature, Plant Output: (d) Outlet Concentration | |

|  |
| --- |
| (a) |
| (b) |
| Fig. 2. MPC Output: (a)Jacket Coolant Temperature; (b) Attributions of MPC Input variables |

MATLAB by considering *Tc* as the manipulated variable, *Cf*, *Tf* as the measured disturbance, and the residual concentration *CA* as the measured output. This MPC is then simulated for numerous changes in the measured disturbance, and the output is recorded. The data collected is then utilized to train machine learning models, and the feature attribution is calculated by following the methodology mentioned in section (2). For the Explainability of the MPC Model, a single ramp change is provided in the *Cf* variable. This change affects other variables, *CA* and *Tc,* but not *Tf,* as they are independent of each other. The trends in plant input and output are illustrated in Fig.1, showcasing the impact of this change. Fig 2(a) shows the variation in the MPC model output *Tc,* and Fig.2(b) shows a feature attribution plot of all the input variables to the MPC model. This feature attribution plot is derived by applying the proposed methodology to an NN-based MPC model. Analyzing the plot shown in Fig.2(b) provides insights into the ongoing process. Firstly, it is evident that from time t=0 to t=3, the attribution of all the input variables is zero, indicating that variables are at their steady-state values and there is no change in the output *TC*. This ensures that operators can easily understand the logic behind the proposed attribution method. After t=3, attributions are observed in the variables *Cf* and *CA*, but *Tf* remains at zero attribution, implying that it has no impact on the output *Tc*. This observation aligns with the notion that changing the feed’s concentration will have no effect on the temperature of the feed. Also, this would assist the operator in identifying the variables that impact MPC output and which are the most significant ones. In feature attributions, positive SHAP values positively impact the model prediction, while those with negative values have a negative impact, and the magnitude measures how strong the impact is. Fig 2(b) shows that variable *Cf* has the highest negative attribution followed by a smaller attribution, *CA*, indicating that *Cf* has more impact on the output *Tc*. This interpretation remains valid, considering that *Cf* was the variable driving the change in the system state. After t=4.2, the attribution of variable *Cf* becomes constant and consistently negative (see Fig 2(b)), implying that its value has stabilized at a level different from its initial baseline, which can be observed in Fig 1(c). Post t=10, the attribution for variable *CA* returns to zero, signifying that the variable has reached its steady state value again. This suggests to the operator that in instances where attributions remain constant or zero, the corresponding variable values stabilize or attain a steady state, thereby indicating a consistent or negligible impact on the model output. Also, after t=4.2, variable *Cf* maintains a constant negative attribution; this suggests that the negative impact of *Cf* persists, and it needs to be counterbalanced by another variable, even after the system achieves a steady state. Fig.2(a) shows that the variable *Tc* does not return to its original position even after the disturbance is rejected, indicating that it is offsetting the effect of *Cf*. Hence, it would aid the operator in understanding that certain variables may not return to their initial state even after the controlled variable regains a steady state.

**4. Conclusions**   
In this study, we proposed a methodology to explain MPC’s actions by utilizing the XAI technique, SHAP. For this purpose, MPC input-output data is used to train different machine-learning models that imitate the strategy of an optimization-based MPC. Then, it is simulated for a ramp change in a measured disturbance signal, and feature attribution is calculated using KernelSHAP. The subsequent attribution analysis, conducted specifically for the test case, revealed that this approach has the potential to enhance operators’ understanding of the MPC’s actions. This analysis offers valuable insights into the variables influencing the MPC output as well as the role played by different variables and their significance. Additionally, this analysis deepens their understanding of the dynamics of the closed-loop process. For future research endeavors, the emphasis will be on extending the work by implementing SHAP on multivariate systems and the systems involving the application of nonlinear MPC.   
   
**5. References**   
C. Lindscheid, A. Bremer, D. Haßkerl, A. Tatulea-Codrean, S. Engell, 2016, A test environment to evaluate the integration of operators in nonlinear model-predictive control of chemical processes. IFAC Papers OnLine, 49, 32, 129–134   
Forbes, M.G., Patwardhan, R.S., Hamadah, H., Gopaluni, R.B., 2015. Model predictive control in industry: challenges and opportunities. IFAC-Papers OnLine, 48, 8, 531- 538  
Bhakte, A., Pakkiriswamy, V., Srinivasan, R., 2022, An Explainable Artificial Intelligence Based Approach for Interpretation of Fault Classification Results from Deep Neural Networks, Chemical Engineering Science, 250, 117373   
Bhakte, A., Chakane, M., Srinivasan, R., 2023, Alarm-based explanations of process monitoring results from deep neural networks, Computers & Chemical Engineering,179, 108442   
Lundberg, S. M., Lee, S.I., 2023, A unified approach to interpreting model predictions, Advances in neural information processing systems, Proceedings of the 31st International Conference on Neural Information Processing Systems, (pp. 4768–4777)