12th IFAC International Symposium on
Advanced Control of Chemical Processes
July 14-17, 2024. Toronto, Canada

Improved Gain Conditioning for Linear Model Predictive Control

Mouna Y. Harb, Stephen D. Sanborn*, Andrew J. Thake**, and Kimberley B. McAuley**

** Department of Chemical Engineering, Queen's University, Kingston, Canada (e-mail: kim.mcauley@queensu.ca)*

*** Imperial Oil, Calgary, Canada (e-mail: andrew.j.thake@esso.ca)*

Abstract: One challenge when using linear model predictive control (MPC) is that model mismatch and ill-conditioned gain matrices can lead to undesirable aggressive controller behavior. To address this issue, we propose improvements to an existing offline method for gain-matrix conditioning. The proposed algorithm identifies problematic manipulated variables (MVs) with correlated effects on controlled variables (CVs) and solves a constrained linear least-squares optimization problem to adjust the problematic gains. Additionally, the proposed algorithm prevents the optimizer from switching the signs of some gains and allows control practitioners to specify trusted key gains that should be held constant. We also extend the method to condition gain submatrices in scenarios where some of the CVs may temporarily be eliminated from the control problem. To illustrate the effectiveness of the proposed algorithm, we present a case study involving industrial fluidized catalytic cracking.

Keywords: Model predictive control, Gain matrix, Ill-conditioned, Gain conditioning.

1. INTRODUCTION

Linear model predictive control (MPC) is widely used in multivariable advanced process control applications (Darby & Nikolaou, 2012; Qin & Badgwell, 2003). MPC uses gain matrices to describe steady-state influences of manipulated variables (MVs) on controlled variables (CVs). These gain matrices may be ill-conditioned if some MVs have highly correlated effects on the CVs. When there are some nearly, but not perfectly, collinear columns in the gain matrix, the MPC may act as if there are more degrees of freedom available than there really are. The MPC may attempt to exploit these apparent degrees of freedom, resulting in undesirable control performance if the actual plant behavior does not perfectly match the model (Marlin, 2015; Seborg et al., 2016).

Several methods are available to prevent issues due to ill-conditioned gain matrices, such as Input Move Suppression (IMS), Singular Value Thresholding (SVT) methods and Relative Gain Array (RGA) methods (Hall et al., 2010; Zheng et al., 2007; Zheng et al., 2017). Recently, Sanborn et al. (2023) proposed an orthogonalization-based method to identify and adjust nearly collinear columns in gain matrices, thereby avoiding aggressive MPC behaviour when there is a plant-model mismatch. Their approach detects problematic MVs with little independent influence on the CVs and adjusts the gains of the problematic MVs (Sanborn et al., 2023).

We propose improvements to Sanborn's algorithm and develop an approach for scenarios where the MPC considers only a subset of CVs of the original gain matrix. Section 2 provides background on Sanborn's algorithm. Section 3 presents the proposed improvements and an approach for conditioning subsets of the full gain matrix. In Section 4, an industrial fluidized catalytic cracking case study is used to test the proposed algorithms. Finally, conclusions are provided in Section 5.

2. BACKGROUND

In Sanborn's algorithm, the MVs are ranked from most influential to least influential based on their steady-state effects on the CVs. During the ranking process, the algorithm identifies problematic MVs that have highly correlated influences with higher-ranked MVs. A MV is considered problematic if less than 5% of its overall influence is independent of the influence of higher-ranked MVs on the importance list (Sanborn et al., 2023).

Control practitioners have the flexibility to adjust this 5% cut-off value based on their confidence in the accuracy of the identified gains. For example, a smaller cut-off value of 3% could be used if they highly trust the gains identified from plant experiments. The ranking algorithm uses a scaled version $\boldsymbol{K_s}$ of the steady-state gain matrix, in which columns correspond to MVs and rows correspond to CVs. The i,j^{th} element of $\boldsymbol{K_s}$ is:

$$K_{s\,ij} = \frac{\Delta CV_i}{\Delta MV_j}\frac{s_j^u}{s_i^y} \quad (1)$$

where scaling factor s_j^u is the typical size of an important change in the j^{th} MV and scaling factor s_i^y is the typical size of an important change in the i^{th} CV. These scaling factors are selected by control practitioners based on their process experience. This scaling leads to gains that are dimensionless and easily comparable (Sanborn et al., 2023). Sanborn's algorithm for ranking and identifying problematic MVs is the same as the proposed algorithm in Table 1, except for step 8. We modify step 8 by introducing a lower cut-off value of 0.0001%. This cut-off helps to identify MV columns that are perfect linear combinations of other higher-ranked MVs. The benefits of using this lower cut-off are described in detail later in the Proposed Methodologies and the Case Study.

After ranking the MVs and determining which ones are problematic, Sanborn's gain-conditioning algorithm fits each problematic column as a linear combination of the higher-ranked non-problematic columns. A constrained linear least-squares optimization problem is solved to ensure that zero-valued gains in the problematic column remain at zero after conditioning. In Sanborn's algorithm, conditioned gains are not allowed to decrease in magnitude from their original values to ensure the robustness of the MPC.

3. PROPOSED METHODOLOGIES

The proposed improvements to Sanborn's algorithm are outlined in Tables 1, 2 and 3. Table 1 summarizes the updated algorithm proposed for ranking MVs and for identifying problematic gains in $\boldsymbol{K}_s$. Table 2 provides an updated algorithm for adjusting the problematic gains. Table 3 provides a new algorithm where submatrices of $\boldsymbol{K}_s$ are considered to prevent conditioning problems when the MPC disregards some of the CVs (e.g., due to sensor failure).

Table 1. Updated orthogonalization-based deflation algorithm for ranking MVs (Sanborn et al., 2023).

1. Calculate the magnitude (Euclidean norm) of each column of the scaled gain matrix, $\boldsymbol{K}_s$.
2. Select the column with the largest magnitude. This column corresponds to the most influential MV.
3. Put the selected column into matrix $\boldsymbol{X}_k$. When one MV has been selected, k=1, and the matrix will contain only one column. When subsequent MVs are selected, $\boldsymbol{X}_k$ will contain additional columns.
4. Calculate $\widehat{\boldsymbol{K}}_{sk}$, the least-squares prediction of the scaled gain matrix, using the information in $\boldsymbol{X}_k$:
$$\widehat{\boldsymbol{K}}_{sk} = \boldsymbol{X}_k\left(\boldsymbol{X}_k^T\boldsymbol{X}_k\right)^{-1}\boldsymbol{X}_k^T\boldsymbol{K}_s \tag{1.1}$$
5. Calculate residual matrix $\boldsymbol{R}_k$:
$$\boldsymbol{R}_k = \boldsymbol{K}_s - \widehat{\boldsymbol{K}}_{sk} \tag{1.2}$$
6. Calculate the magnitudes of each column of $\boldsymbol{R}_k$. The column with the largest magnitude corresponds to the next-most-influential MV. Check if the column in $\boldsymbol{R}_k$ corresponding to this MV has a magnitude that is at least 5% of the corresponding column magnitude in $\boldsymbol{K}_s$.
7. If the residual column magnitude from step 6 (or step 8) meets the 5% criterion, augment matrix $\boldsymbol{X}_k$ by including the new column. This augmented matrix is $\boldsymbol{X}_{k+1}$.
8. If the residual magnitude considered in step 7 is between 0.0001% and 5% of the corresponding column magnitude in $\boldsymbol{K}_s$, put the corresponding MV on the list of problematic MVs requiring conditioning. If unranked MVs remain, then select the MV corresponding to the next-largest column in $\boldsymbol{R}_k$ as the next-most-influential MV and go to step 7.
9. Advance iteration counter k by 1 and repeat steps 4-8 until all the MVs are ranked, or singularity problems are encountered when inverting $\boldsymbol{X}_k^T\boldsymbol{X}_k$.

Steps 1 to 7 in Table 1 are identical to Sanborn's algorithm. In step 8, a proposed lower cut-off value of 0.0001% is introduced to prevent previously conditioned columns from being re-identified as problematic when submatrices are considered for conditioning using the algorithm in Table 3.

In Table 2, we propose three improvements to Sanborn's gain conditioning algorithm: *i)* a new constraint in the linear least-squares optimization problem to prevent gains from switching signs during conditioning; *ii)* a new constraint that can be used to hold key gains constant during conditioning; and *iii)* an updated constraint to permit conditioned gains to be smaller than their original values by a practitioner-specified factor. These three improvements are implemented in step 1 of the proposed gain conditioning algorithm shown in Table 2.

Table 2. Proposed algorithm for conditioning gains corresponding to the problematic j^{th} MV.

1. Solve the following optimization problem where $\boldsymbol{K}_{sj}$ is the problematic j^{th} column in the scaled gain matrix $\boldsymbol{K}_s$:
$$\min_{\beta_j}\left\|\boldsymbol{X}_k\,\boldsymbol{\beta}_j - \boldsymbol{K}_{sj}\right\|^2 \tag{2.1}$$
subject to:
$$\boldsymbol{X}_{ki}\,\boldsymbol{\beta}_j = 0 \quad \text{for all } i \text{ where } K_{s\,i,j} = 0 \tag{2.2}$$
$$K_{s\,i,j}\;\widehat{K}_{s\,i,j} \geq 0 \quad \text{for all } i \tag{2.3}$$
$$\boldsymbol{X}_{ki}\,\boldsymbol{\beta}_j = K^*_{s\,i,j} \quad \text{for all } i \tag{2.4}$$
$$\left|\widehat{K}_{s\,i,j}\right| \geq 0.95\left|K_{s\,i,j}\right| \quad \text{for all } i \tag{2.5}$$
In Equation (2.1), $\boldsymbol{X}_k$ is the matrix obtained from the algorithm in Table 1 after the first k MVs have been ranked. $\boldsymbol{X}_k$ contains the k columns from $\boldsymbol{K}_s$ corresponding to the non-problematic MVs that are higher ranked than the MV in the j^{th} column.

In Equations (2.2) and (2.4), $\boldsymbol{X}_{ki}$ is a row vector corresponding to the i^{th} row in $\boldsymbol{X}_k$. In Equations (2.2) to (2.5), $K_{s\,i,j}$ is the i^{th} gain of the j^{th} problematic MV and $\widehat{K}_{s\,i,j}$ is the i^{th} gain in the conditioned column $\widehat{\boldsymbol{K}}_{sj}$.
2. If the optimization problem in step 1 is feasible, use the least-squares parameter estimates $\widehat{\boldsymbol{\beta}}_j$ from step 1 to obtain a conditioned column $\widehat{\boldsymbol{K}}_{sj}$:
$$\widehat{\boldsymbol{K}}_{sj} = \boldsymbol{X}_k\widehat{\boldsymbol{\beta}}_j \tag{2.6}$$
and calculate the percent change of the conditioned gain $\widehat{K}_{s\,i,j}$ from the corresponding original gain $K_{s\,i,j}$ for each element in the j^{th} column.
$$\%\,change = 100\,\frac{\widehat{K}_{s\,i,j}-K_{s\,i,j}}{K_{s\,i,j}} \tag{2.7}$$
Flag the conditioned gain if $\%\,change > 20\%$ and ask the control engineer to decide whether to accept the conditioning or keep the original column of scaled gains.

Equations (2.1) and (2.2) remain the same as the corresponding equations in Sanborn's algorithm. Equation (2.3) is the new constraint used to preserve the signs of the gains after conditioning. The product on the left-hand side of Equation (2.3) is always positive when $K_{s\,i,j}$ and the corresponding conditioned gain $\widehat{K}_{s\,i,j}$ have the same sign. Although there may be considerable uncertainty about the value of a small steady-state gain, we believe that control practitioners are certain about the signs of the gains used in their MPC applications. Otherwise, they would opt to set a small uncertain gain to zero.

Sometimes, control practitioners may want to select important gains, which they believe to be accurate, and specify that they should remain fixed during conditioning. The new constraint in Equation (2.4) ensures that a well-known gain $K^*_{s\,i,j}$ (corresponding to the i^{th} CV and j^{th} MV) will be held constant when other gains are adjusted.

The new constraint in Equation (2.5) is proposed so that small reductions in magnitude between conditioned gains $\widehat{K}_{s\,i,j}$ and the original gains $K_{s\,i,j}$ are permitted. As recommended by our industrial sponsor, we set the minimum ratio at 0.95 in Equation (2.5) to ensure that the optimizer does not decrease the magnitude of any conditioned gain by more than 5% of its original value. This minimum ratio of 0.95 leads, on average, to smaller adjustments than the ratio of 1.0 used in Sanborn's gain conditioning algorithm.

In step 2 of the algorithm in Table 2, updated values are computed for gains in the j^{th} column of $\boldsymbol{K}_s$. If the magnitude of the change in any calculated gain is more than 20% of the original gain, this gain is flagged so the control engineer can decide whether to keep the original gain or accept the conditioned value. The 20% threshold was suggested by our industrial sponsor and could be modified to suit the desires of other practitioners.

The new algorithm in Table 3 is proposed to ensure proper conditioning of gain submatrices when some of the variables are removed from the MPC optimization problem. Sometimes, MVs are removed when they are set manually by the operator or when valves or heaters are out of service. Also, CVs may be removed from the MPC problem by higher-level real-time optimization (RTO) or linear program (LP) applications (Elnawawi et al., 2022). CVs may also be removed when certain sensors are not available.

When MVs are temporarily eliminated from the control problem, the corresponding columns are eliminated from the gain matrix. Similarly, when CVs are eliminated, the corresponding rows are eliminated. Removing columns will not cause a well-conditioned gain matrix to become ill-conditioned (Sanborn et al., 2023). However, removing rows causes ill-conditioning when previously independent columns become (nearly) perfect linear combinations of other columns. The Case Study in this article provides several examples where removing CVs causes MVs to become problematic.

In the first step in the algorithm in Table 3, the gain matrix $\boldsymbol{K}_s$, which is often non-square (Elnawawi et al., 2022), is conditioned prior to removing any CVs. Also, the practitioner specifies r_{max}, which is the number of CVs that could reasonably be removed from the control problem before the plant operators would decide to shut off the MPC. In steps 2 to 4, CVs are removed from the gain matrix, starting with one CV at a time, then progressing to two, and so on, up to r_{max}. Submatrices with problematic MVs are identified in step 2. Problematic gains in $\boldsymbol{K}_s$ are replaced by conditioned gains in step 3. As the algorithm in Table 3 proceeds, it may encounter some columns that were previously conditioned and are now perfect linear combinations of other columns. The lower cut-off value of 0.0001%, which appears in step 8 in Table 1, prevents updated columns that are now perfect linear combinations of other columns from showing up repeatedly as problematic. Examples of this situation are highlighted in the Case Study.

Table 3. Proposed algorithm for selecting and conditioning subsets of the full gain matrix.

1. Condition the full gain matrix $\boldsymbol{K}_s$ with n_y rows and n_u columns using the updated MV ranking algorithm in Table 1 and the proposed gain conditioning algorithm in Table 2. Set the counter for the number of rows that will be removed at $r = 1$ and the counter for the corresponding submatrices for consideration to $s = 1$. Specify r_{max} as the maximum number of CVs that could be removed from the gain matrix.
2. Select the s^{th} submatrix from $\boldsymbol{K}_s$ by removing r rows. Use the ranking algorithm in Table 1, where $\boldsymbol{K}_s$ is replaced with the submatrix of interest, to determine if any MVs have become problematic due to removal of rows. If a problematic MV is detected, condition the gains in the corresponding submatrix using the algorithm in Table 2, where $\boldsymbol{K}_s$ is replaced with the submatrix of interest.
3. Replace each conditioned gain from the submatrix in step 2 in the full matrix $\boldsymbol{K}_s$. If $s = \binom{n_y}{r}$, continue to step 4; otherwise, increase s by 1 and return to step 2 to consider the next possible submatrix with r rows removed.
4. If $r < r_{max}$, increase r by 1. Set $s = 1$ and return to step 2.
5. Repeat the algorithm in steps 1 to 4, once more, with $\boldsymbol{K}_s$ replaced by the conditioned gain matrix $\widehat{\boldsymbol{K}}_s$ as a final check to confirm that no further conditioning is required. Then, stop and report the final conditioned gain matrix.

4. CASE STUDY

4.1 Process description

The proposed algorithms are applied to the gain matrix for the industrial fluid catalytic cracking (FCC) process shown in Figure 1. FCC converts heavy hydrocarbons to higher-value, lighter hydrocarbons like gasoline and light olefins (Bai et al., 2019). In this case study, the FCC unit consists of a preheater, a reactor, a regenerator and a fractionator. The feed is heated in a gas-fired preheater before it enters the reactor, where it is mixed with a hot catalyst. Heavy hydrocarbons are vaporized and cracked into lighter hydrocarbon molecules in the reactor. The resulting vapour is separated from the spent catalyst at the top of the reactor. This light hydrocarbon stream flows to the fractionator, while the catalyst moves downward the reactor and enters the regenerator. Air is fed to the regenerator to burn off the coke layer that forms on the catalyst in the reactor. The catalyst is then fed back to the reactor to repeat the cycle. The fractionator separates the vapour into light-end products, naphtha, light-cycle gas oil and heavy-cycle gas oil. The MPC in this case study uses 8 MVs (highlighted in orange) to influence 13 CVs (highlighted in blue) as shown in Figure 1. The corresponding scaled gain matrix $\boldsymbol{K}_s$ is provided in Table 4. Details about the MVs, CVs, scaling factors and process dynamics are provided by Sanborn et al., 2023.

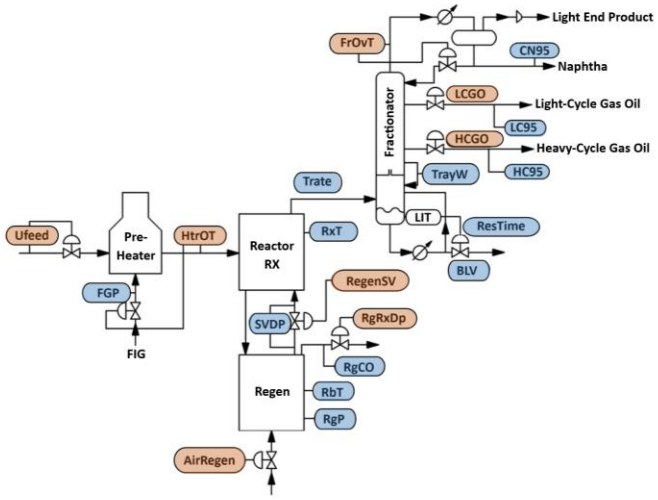

Figure 1. FCC process considered in the case study. MVs are highlighted in orange, and CVs are highlighted in blue. Adapted from Sanborn et al., 2023.

4.2 Implementation of proposed algorithms for gain conditioning

First, the full 13×8 scaled gain matrix $\boldsymbol{K}_s$ is analyzed to determine if it requires conditioning as specified in step 1 in Table 3. The full gain matrix is well-conditioned and does not contain problematic columns when all 13 CVs are considered by the MPC (Sanborn et al., 2023). Setting $r_{max} = 6$ in this case study indicates that the MPC would be shut off if more than six CVs are excluded from the control problem. In step 2 in Table 3, setting $r = 1$ produces 13 gain submatrices for consideration where one of the CVs is removed from $\boldsymbol{K}_s$. Using step 2 in Table 3, 4 of these 13 submatrices are identified as having problematic MVs (i.e., the submatrices with rows for LC95, CN95, FGP and RgP removed). Removing the row for LC95 from the gain matrix makes LCGO a problematic MV because the influence of LCGO becomes nearly collinear with the influence of HCGO. As shown in Figure 1 and Table 4, both LCGO and HCGO have influences on BLV, ResTime, TrayW and HC95, but LCGO has a special influence on LC95, whereas HCGO does not. When LC95 is removed, LCGO loses its special influence in the control problem. Notice that the column for LCGO with the gain for LC95 removed is nearly a perfect multiple (by a factor of 1.2) of the column for HCGO. The resulting conditioned gains for LCGO (obtained in step 2 in Table 3 using the algorithm in Table 2) are highlighted in green and yellow in Table 4 and are shown below the corresponding unconditioned gains. Notice that the three conditioned gains with green highlighting are close in value to the original scaled gains identified from plant experiments, but the gain highlighted in yellow (between LCGO and HC95) changes by 22.4%. As a result, this conditioned gain is flagged as prescribed in step 2 in Table 2, so the control engineer can decide whether this column should be conditioned or left in its original state. We assume that the control engineer decides to accept the proposed column of conditioned gains. The engineer may make this decision after judging that the relatively small gain between LCGO and HC95 could be quite inaccurate based on the limited step tests that were performed. As a result, the corresponding column is updated in $\boldsymbol{K}_s$.

It also makes sense that FrOvT is identified as problematic when CN95 is removed. Removing CN95 makes the influence of FrOvT nearly collinear with that of LCGO. When the row for CN95 is removed in Table 4, FrOvT loses its independent influence as the gain between FrOvT and CN95 is removed. The resulting conditioned gains for the FrOvT column are highlighted in green in Table 4 and are all close to their unconditioned values. As a result, the gain matrix $\boldsymbol{K}_s$ is updated without requiring any input from the control practitioner.

The third problematic MV identified is HtrOT, which becomes problematic when FGP is removed. HtrOT loses its special influence in the control problem when the gain between FGP and HtrOT is removed, making the columns for HtrOT and AirRegen nearly colinear. As shown in Table 4, the changes required to condition the gains for HtrOT are small.

When RgP is removed from the control problem, RgRxDp becomes problematic because its influence is nearly colinear with that of RegenSV. If we had used Sanborn's algorithm to condition the RgRxDp column in this situation, the optimizer would have switched the sign of the third gain (from 0.50637 to −0.10315). However, the proposed algorithm determines that the optimization problem is infeasible because of the constraint in Equation (2.3), so gains of RgRxDp are not adjusted using step 2 in Table 2.

Next, using step 4 in Table 3, we set $r = 2$, which results in $\binom{13}{2} = 78$ gain submatrices for consideration. HCGO is identified as a problematic MV when ResTime and HC95 are both removed from the control problem. The corresponding optimization problem in step 1 in Table 2 is infeasible due to the constraint in Equation (2.5), so the column of gains is not updated. Using Sanborn's algorithm, which does not include this constraint, would result in an undesirable outcome where a column of zeros replaces the gains for HCGO.

Table 4. Overall scaled 13×8 gain matrix. Conditioned gains are highlighted in green and yellow below the corresponding original identified gains. The magnitude of gains highlighted in yellow changed by more than 20% from their original identified values.

CV \ MV	UFeed	HtrOT	RegenSV	AirRegen	RgRxDp	HCGO	LCGO	FrOvT
Trate	0.60787							
RxT	-18.47826	10.50146 10.75785		14.62094				
SVDP			-0.12096		0.50637			
RgP					1.19021			
BLV	31.43840	-8.84327 -8.40111		-11.41790		-12.08133	-10.20630 -10.4768	-7.82886 -8.27101
ResTime	-6.48934	3.42219 3.607434		4.90284		1.69378	1.54614 1.468831	1.2050 1.159582
TrayW	9.85357	-1.66070 -1.60942		-2.18735		-5.11803	-4.47517 -4.43831	-3.68675 -3.50386
FGP	0.25669 0.243857	6.30792						
CN95								0.49316
LC95							1.55958	1.25899 1.231233
HC95						1.07989	0.76505 0.936467	0.68038 0.739302
RbT	-26.37746 -25.8819	16.09846 17.0238	-26.93788	23.13695	-23.48435 -22.9712			
RgCO	13.09083 14.00874	-8.16632 -8.25447	14.54263	-11.21861	11.45062 12.40118			

Next, AirRegen is identified as problematic when both BLV and TrayW are removed. The conditioning algorithm determines that the optimization problem in Table 2 is infeasible due to the constraint in Equation (2.5), so the corresponding column remains unchanged. RgRxDp is also identified as problematic due to the removal of both SVDP and RgP. This result makes physical sense because RgRxDp loses its independent influence on RgP. As a result, the remaining influence of RgRxDp becomes nearly colinear with that of RegenSV. The corresponding optimization problem is feasible, so the column for RgRxDp is updated as shown by the green entries in Table 4. Recall that when only one CV was removed (i.e., RgP), gains for RgRxDp were not updated due to the infeasibility of the optimization problem. Now, when two CVs (i.e., SVDP and RgP) are removed, the corresponding optimization problem becomes feasible, permitting changes in the gains between RxRgDp and two of the CVs (i.e., RbT and RgCO). Subsequently, RgRxDp is identified as problematic again when several other pairs of CVs are removed (e.g., RgCO and RgP), but further updating of RgRxDp does not occur due to infeasible optimization problems.

Setting $r = 3$ yields $\binom{13}{3} = 286$ gain submatrices for consideration. FrOvT is identified as problematic when BLV, CN95 and ResTime are removed simultaneously, so the corresponding column of $\boldsymbol{K}_s$ is updated. FrOvT is then re-identified as problematic three more times when other triples of CVs are removed. Only minor changes of < 0.05% are made to gains when this re-identification occurs.

When $r = 4$, a total of 715 gain submatrices are obtained for consideration. No changes are made to the gains in any of these submatrices because any potentially problematic MVs that are identified lead to infeasible optimization problems. Similarly, when $r = 5$, no further conditioning occurs.

Finally, setting $r = 6$ results in conditioning of the gains for Ufeed when the following 6 CVs are removed together: Trate, RxT, SVDP, CN95, BLV and TrayW. When all these CVs are removed, the influence of Ufeed becomes nearly colinear with the influence of HtrOT. The updated gains are shown in green in Table 4. The final verification (i.e., step 5 in Table 3) is then performed to test whether any of the conditioning that was applied resulted in additional problematic MVs. No additional problematic columns were identified.

In total, the proposed algorithm made small adjustments to 20 out of the 43 non-zero gains, as summarized in Table 4. The largest percentage change is 22.4% for the gain between LCGO and HC95. The average absolute value for the percentage change in adjusted gains is 5.16%.

5. CONCLUSIONS

A recently developed gain-conditioning algorithm was improved so that it is now more reliable for use in industry. The proposed updated algorithm ensures that the signs of all gains are preserved during conditioning. It also allows the magnitudes of conditioned gains to be slightly smaller than the corresponding original gains. Using the proposed algorithm, control engineers can now opt to hold some key gains constant during conditioning. Most importantly, the revised algorithm is able to condition subsets of the full gain matrix, anticipating situations where CVs may be removed from the MPC problem. The FCC case study demonstrates that the proposed algorithm can readily be applied in practical MPC problems with non-square gain matrices. Future research involving the FCC case study will include dynamic MPC simulations to verify the effectiveness of the proposed algorithms.

REFERENCES

Bai, P., Etim, U.J., Yan, Z., Mintova, S., Zhang, Z., Zhong, Z. and Gao, X., 2019. Fluid catalytic cracking technology: current status and recent discoveries on catalyst contamination. *Catalysis Reviews, 61*(3), pp.333-405.

Darby, M.L. and Nikolaou, M., 2012. MPC: Current practice and challenges. *Control Engineering Practice*, *20*(4), pp.328-342.

Elnawawi, S., Siang, L.C., O'Connor, D.L. and Gopaluni, R.B., 2022. Interactive visualization for diagnosis of industrial Model Predictive Controllers with steady-state optimizers. *Control Engineering Practice*, *121*, p.105056.

Hall, R.S., Punuru, A.R., Peterson, T.J., Pottorf, T.S. and Vowell, L.E., ExxonMobil Research and Engineering Co, 2010. *Method for model gain matrix modification*. U.S. Patent 7,797,063.Marlin, T. E. (2015). Process Control, Designing Processes and Control Systems for Dynamic Performance.

Qin, S.J. and Badgwell, T.A., 2003. A survey of industrial model predictive control technology. *Control engineering practice*, *11*(7), pp.733-764.

Sanborn, S.D., Varshney, D. and McAuley, K.B., 2023. Orthogonalization-Based Gain Conditioning for Linear Model Predictive Control. *Industrial & Engineering Chemistry Research*, *62*(3), pp.1463-1479.Seborg, D. E., et al. (2016). Process dynamics and control: John Wiley & Sons.

Zheng, Q., Harmse, M.J., Rasmussen, K. and McIntyre, B., Aspen Technology Inc, 2007. *Methods and articles for detecting, verifying, and repairing collinearity in a model or subsets of a model*. U.S. Patent 7,231,264.

Zheng, Q.Q., Keenan, M.R. and Harmse, M.J., Aspen Technology Inc, 2017. *Online control calculation for models containing near colinearity and uncertainty*. U.S. Patent 9,715,221.