Multi-Objective Model Predictive Control

Abdelillah Otmane Cherif, Dmitry Balandin

***Abstract*— Multi-objective optimization design recently has attracted great attention of the researchers in solving engineering problems that have conflicting objectives. Although several control specifications which are often irreconcilable can be considered in the single objective function, choosing the appropriate weighting functions are another challenge faced by control designers. In this paper, a new Model Predictive Control scheme based on the multi-objective optimization is proposed in which at each sampling time, the MPC control action is chosen automatically among the set of Pareto optimal solutions based on the Nash Bargaining Solution from Game Theory. This method is independent of the system type. It is applied on the nonlinear systems along with TP transformation to design multi-objective MPC. As a result, LMIs and convex optimization techniques can be utilized to provide an on-line solution for the multi-objective MPC design. The proposed method is executed on a complex nonlinear system. It is shown through the examples that the proposed method can execute approvingly compared to other methods in the literature of the control systems.**

***Keywords*— Linear programming (LP), Model predictive control (MPC), real-time control, Nash bargaining, Uncertain multi-objective optimization.**

I. Introduction

Recently, many control designers have been working on design methods which satisfy multiple design specifications called multi-objective control design [15]. Multi-objective optimization which is also called multi-criteria optimization or vector optimization has been defined as the finding of the decision variables vector satisfying constraints to give optimal values to all objective functions. One of the methods of solving a multi-objective optimization problem is to reduce the optimization dimension.

This method is based on using weighted quadratic sum of the objective functions rather than solving them simultaneously. Since conflict exists between them, choosing appropriate weighting factor in this method is inherently difficult and could be regarded as a subjective design concept. Moreover, the trade-offs existed between some objectives cannot be explored and it would be.

Therefore, impossible to choose an appropriate optimum design reflecting the compromise of the designer's choice concerning the absolute values of objective functions. Therefore, this problem can be formulated as a multi-objective optimization problem so that the trade-offs between objectives can be found consequently. The results of multi-objective optimization is a set of optimal solutions which is called Pareto Frontier [7].

In many practical optimization problems, only one among solutions belonging to the set of Pareto should be selected as the final solution. For example, in MPC design, one control action can be selected for the current sampling time. García et al. used evolutionary algorithms to find Pareto frontier and a fuzzy inference system as an expert decision maker to select the best solution of the Pareto set.

This method would be computationally intractable, because all the Pareto solutions should be obtained at each time step and then a trade-off point is selected using decision maker.

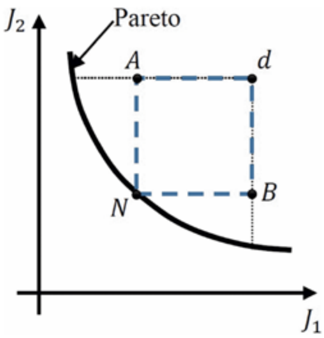


Figure 1: The Nash bargaining solution

One of the useful methods to find the best solution is based on the Bargaining concepts which has its origin in two papers by Nash [7]. Base on the proposed method by Nash, called Nash Bargaining Solution, the trade-off point is the point of Pareto feasible set, S, at which the product of utility gains from Threat point is maximal [7]. Threat or disagreement point, $d = (d_1, d_2)$, is a point in which players (objectives) can expect to receive other better outcomes than the one which becomes effective when they do not cooperate or negotiations break down. According to the Nash bargaining solution, the trade-off point is

$$N(\zeta, d) = \arg\max_{J\in\zeta} \prod_{i=1}^{N} (J_i - d_i)\,,\ for\ J \in \zeta\ \ with\ J \leq d,$$

where J_i is the objective function, and $J \leq d$ means J dominates d. Figure 1 illustrates the concept of Nash bargaining solution geometrically. The Nash bargaining solution is the point on the edge of S and a part of the Pareto frontier which yields the largest rectangle $(N,\ A,\ B,\ d)$ [7]. This point can be obtained by a few Pareto points; therefore, this method can be beneficial to MPC design. We use this method for the proposed MPC design. The methodology of the controller design using Nash Bargaining Solution is given in this paper.

II. Problem formulation

Manuscript received March 13, 2022.
Abdelillah OTMANE CHERIF, Post-Graduate Student, Lobachevsky State University of Nizhny Novgorod, Nizhny Novgorod 603105, Russia, Email: abdouiitmm@gmail.com

Dmitry Balandin, Doctor of Physics and Mathematics, Lobachevsky State University of Nizhny Novgorod, Nizhny Novgorod 603105, Russia, Email: dmitriy.balandin@itmm.unn.ru

Consider the following discrete-time LPV system of the form:

$$x_{k+1} = A\,x_k + B\,u_k$$
$$y_k = C\,x_k + D\,u_k \qquad (1)$$

where $x_k \in \mathbb{R}^n$ is the state vector, $u_k \in \mathbb{R}^m$ is the control input and $y_k \in \mathbb{R}^n$ is the measured output. Also, the input and output are subjected to the following constraints:

$$-\mathrm{u}_{max} \le \mathrm{u}_k \le \mathrm{u}_{max}$$
$$-y_{max} \le y_k \le y_{max} \qquad (2)$$

The time-varying system matrix is defined as follows:

$$S = \begin{pmatrix} A & B \\ C & D \end{pmatrix} \in \mathbb{R}^{(n+p)\times(n+m)} \qquad (3)$$

This matrix belongs to the convex hall - $S := \{\sum_{i=1}^{L} w_i S_i \;;\; w_i \;\ge\; 0,\; \sum_{i=1}^{L} w_i = 1\}$ (4), including the convex combination of (L) LTI models.

Using the LPV model, the control signal is derived by minimizing an upper bound of the worst-case infinite horizon quadratic cost at sampling time k

$$J^k = \max_{p\in\Omega} \sum_{i=0}^{\infty} x_{k+i|k}^T Q x_{k+i|k} + u_{k+i|k}^T R u_{k+i|k} \qquad (5)$$

where $Q > 0$ and $R > 0$ are weighting matrices which should be designed by the designer to make a trade-off between the response performance, and control input cost. Therefore, the performance of the control system depends on these matrices. In the following sections, a method will be proposed in which the designer does not need to specify these matrices.

III. Multi-objective MPC design

Our goal is to find a method which helps the designer to solve the multi-objective MPC problems without choosing the weighting matrices. In order to obtain the control signal at each sampling time, the following optimization problem should be solved at each sampling time

$$\underset{\gamma_k, X_k, Y_k}{\mathrm{minmize}}\; \gamma_k$$

subject to

$$\begin{bmatrix} 1 & x_{k|k}^T \\ x_{k|k} & X_k \end{bmatrix} \ge 0$$

$$\begin{bmatrix} X_k & * & * & * \\ A_j X_k + B_j Y_k & X_k & * & * \\ X_k & 0 & \gamma Q^{-1} & * \\ Y_k & 0 & 0 & \gamma R^{-1} \end{bmatrix} \ge 0 \qquad (6)$$

$$\begin{bmatrix} u_{max}^2 \boldsymbol{I} & Y_k \\ Y_k^T & X_k \end{bmatrix} \ge 0$$

$$\begin{bmatrix} y_{max}^2 \boldsymbol{I} & C(A_j X_k + B_j Y_k) \\ (A_j X_k + B_j Y_k)^T C^T & X_k \end{bmatrix} \ge 0$$

where, $j = 1, 2, \dots, L$ (L is the number vertices). And, u_{max} and y_{max} are the upper bound for input and output, respectively.

The LMI control design method in (6) can be extended to the multi-objective optimization problem. Since, the performance of the control system depends on the weighting matrices, the following method is proposed to tune these matrices based on the Nash bargaining solution at each sampling time.

According to the Pareto front properties, since the objective functions in (4) are convex, the cost function J can be defined as a linear combination of those convex objective functions by specifying the weighting matrices, Q and R as follows

$$Q = diag(\hat{\alpha}), \quad \hat{\alpha} = [\alpha_1, \alpha_2, \dots \dots, \alpha_n] \qquad (7)$$
$$R = diag(\bar{\alpha}), \quad \bar{\alpha} = [\alpha_{n+1}, \alpha_{n+2}, \dots \dots, \alpha_{n+m}] \qquad (8)$$

where $\boldsymbol{diag}$ means the diagonal matrix and $\alpha = [\hat{\alpha}, \bar{\alpha}] \in \mathcal{A}$ is the tuning parameter vector obtained by Nash bargaining solution at each sampling time. Therefore, the designer does not need to design the weighting matrices, since they are automatically tuned at each sampling time.

In order to find the Nash equilibrium point, first of all, the threat or disagreement point must be found. The threat point can be obtained by solving the optimization problem at $k = 0$ for $\alpha_s = 1,$ and $\alpha_i = 0,\; i \ne s,$ that is, the optimization problem is solved for each objective function separately. Then n different points will be obtained. Each of which has the best value for the corresponding objective function and they may have the worse value for other objective functions. Finally, one single point among them is obtained, which is dominated by other points, and it is considered as a threat or disagreement point. Now, the multi-objective MPC can be designed using Nash equilibrium point.

The design steps of the multi-objective MPC are given as follows:

Step 1: at $k = 0$

1. **for** $i = 1$ **to** L
2. $\alpha \leftarrow [0, 0, \dots, 0]$
3. $\alpha_i = 1;$
4. **find** $v_i = arg\; \min_{v} J_i$
5. **end**

Step 2: find threat point d

6. **for** $i = 1$ **to** L
7. $d_i \leftarrow max(J_i(v_1), J_i(v_2), \dots, J_i(v_L))$
8. **end**

Step 3: for $k = 1$ **to** *End* **do**

9. **set**: $\alpha^0 \leftarrow [\frac{1}{L}, \dots, \frac{1}{L}]$
10. **compute**: $v^* = arg \min_{v\in\bar{v}} \sum_{i=0}^{l} \alpha^0 J_i(v)$
11. **for** $i = 1$ **to** L **verify**
12. **if** $J_i(v^* \le\; d_i)$
13. calculate: $\tilde{\alpha}_j^N = \frac{\prod_{i\ne j}(d_i - J_i(v^*(\alpha^0)))}{\sum_{i=1}^{l} \prod_{h\ne i}(d_h - J_h(v^*(\alpha^0)))} \quad j = 1, \dots, L$
14. **else find** i_0 **which** $J_{i_0}(v^*) > d_{i_0}$
15. **update**: $\alpha_{i_0}^0 := \alpha_{i_0}^0 + 0.01$, $\alpha_i^0 := \alpha_i^0 - \frac{0.01}{l-1} \quad i \ne i_0$

return to 10
end
if $\left|\tilde{\alpha}_j^N - \alpha_i^0\right| < 0.01 \;\;, \; i = 1, \dots, L$
terminate
set: $\alpha_j^N = \tilde{\alpha}_j^N$
$k \leftarrow k + 1$
else $\alpha_i^0 \coloneqq 0.08\,\alpha_i^0 + 0.2\,\tilde{\alpha}_i^N$
return to 10
end

Note: numbers 0.01 and 0.8 in the above-mentioned design procedure are chosen arbitrary.

IV. Application on two Mass-Spring model

To illustrate the effectiveness of the proposed MPC method, the system consisting of a two mass-spring model with a time-varying nonlinear spring coefficient, is considered and shown in Figure 1.

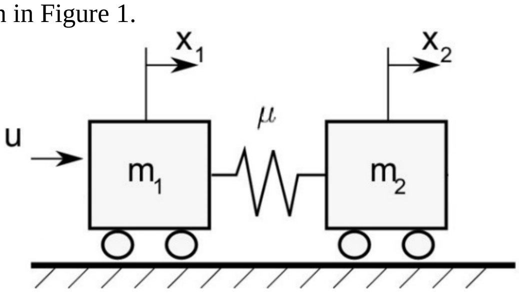


Figure 2: Two mass spring system

A. *Problem Formulation:*

The discrete-time state-space model of the two-mass-spring system (obtained from the continuous time model using a first-order Euler approximation with sampling time $(h = 0.1s)$ is given as follows:

$$x_{k+1} = \begin{pmatrix} 1 & 0 & 0.1 & 0 \\ 0 & 1 & 0 & 0.1 \\ -0.1\frac{\mu}{m_1} & 0.1\frac{\mu}{m_1} & 1 & 0 \\ 0.1\frac{\mu}{m_2} & -0.1\frac{\mu}{m_2} & 0 & 1 \end{pmatrix} x_k + \begin{pmatrix} 0 \\ 0 \\ \frac{0.1}{m_1} \\ 0 \end{pmatrix} u_k \qquad (9)$$

where m_1 and m_2 are two masses and μ is the spring constant. The state vector at each sampling time, x_k , includes the position of the masses $x_{1,k}$, $x_{2,k}$ and their velocities, $x_{3,k}$, $x_{4,k}$. In this example the masses are constant $m_1 = 1$ and $m_2 = 1$, while spring constant varies with time according to the following equation:

$$\mu_k = 5.25 - 4.75 \sin(0.5k) \qquad (10)$$

It can be seen that $\mu_k \in [0.5, 10]$. According to [6], the weighting functions w_k can be defined as $w_{1,k} = 1 - \frac{\mu_k - 0.5}{9.5}$ and $w_{2,k} = 1 - w_{1,k}$,k which satisfy convex hull condition in (4). For this system, two vertices based on the maximum and minimum values of the spring constant can be obtained as follows:

$$A_1 = \begin{pmatrix} 1 & 0 & 0.1 & 0 \\ 0 & 1 & 0 & 0.1 \\ -0{,}05 & 0{,}05 & 1 & 0 \\ 0{,}05 & -0{,}05 & 0 & 1 \end{pmatrix} \qquad (11)$$

$$A_2 = \begin{pmatrix} 1 & 0 & 0.1 & 0 \\ 0 & 1 & 0 & 0.1 \\ -1 & 1 & 1 & 0 \\ 1 & -1 & 0 & 1 \end{pmatrix} \qquad (12)$$

$$B_1 = B_2 = \begin{pmatrix} 0 \\ 0 \\ 0.1 \\ 0 \end{pmatrix} \qquad (13)$$

The objective of the control design is to steer the two masses from the initial condition $x_0 = [1, 1, 0, 0]^T$ to the origin. The control system must satisfy the input and output (positions of the masses) constraints, $\|u_k\| \le 0.05$ and $\|y_k\| \le 1$, respectively. Since, the constraints are considered only for the positions, the matrix $C = [1\,0\,0\,0\,;\,0\,1\,0\,0]$ is independent of w_k and is kept the same for both vertices.

B. *Simulation Results:*

Figure.3 illustrates the obtained simulation results. It is obvious that the system is asymptotically stable, and the states are steered to the origin efficiently. *Figure.4* shows the results from [4], [5], and [6]. It is noticed that the proposed MPC method steers the masses to the origin significantly faster compared with the other methods. According to the obtained results, the settling time is about $25\ sec$, while it is about $45\ sec$ in [6].

The control signal of the proposed MPC method is shown in *Figure.5a* Also, *Figure.5b* illustrates the control signal behavior of the methods in [4, 5, 6]. It can be seen that both input and output responses satisfied the considered constraints. Although the results show that the proposed MPC method has the greater control signal magnitude than the other methods do, it does not violate the constraint; therefore, the results are acceptable. It shows the advantage of the proposed method. At the beginning of the response, the state error is large. Therefore, using the Nash bargaining solution, the bigger states weights are chosen while the weight of the control input is small. This automatic tuning procedure is carried out within the simulation at each sampling interval to find the variable weighting matrices as $Q = \alpha_1 I_{4\times4}$, and $R = \alpha_2$, $([\alpha_1, \alpha_2] \in \mathcal{A})$. On the contrary, the other methods use the fixed weighting matrices as $Q = I_{4\times4}$, and $R = 1$.

Figure.6 shows the minimized upper bound on the worst-case cost function in (6). It clearly shows that the proposed method obtains incomparably smaller upper bound at each sampling time (*Figure.6a*) in comparison with the other methods (*Figure.6b*). As a result, the proposed controller is closer to the optimal solution that may be obtained for the unconstrained optimal control method (global minimal solution).

It is evident that applying the proposed MPC method on the considered LPV model achieve significant performance improvements compared with the methods presented in [4], [5], [6].

To generalize the application of the proposed method, in the following section a highly nonlinear system is considered and the proposed method is applied to that. Finally, the obtained results are compared with proposed method in

[7].

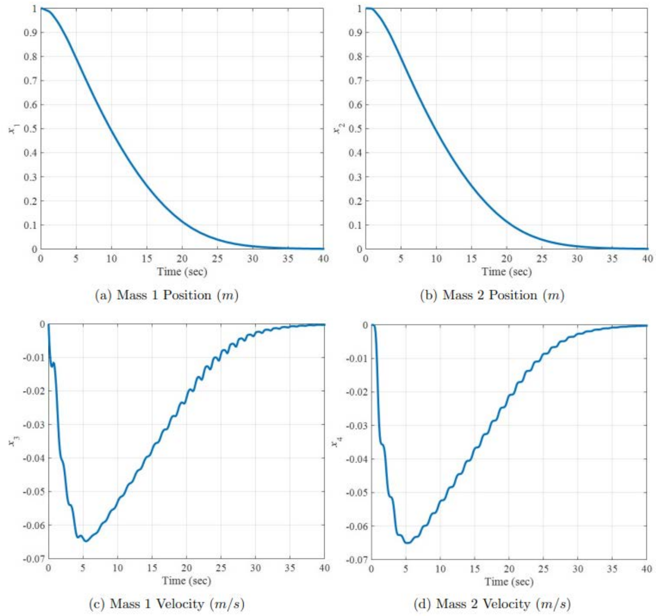


Figure 3: Time response of the state variables.

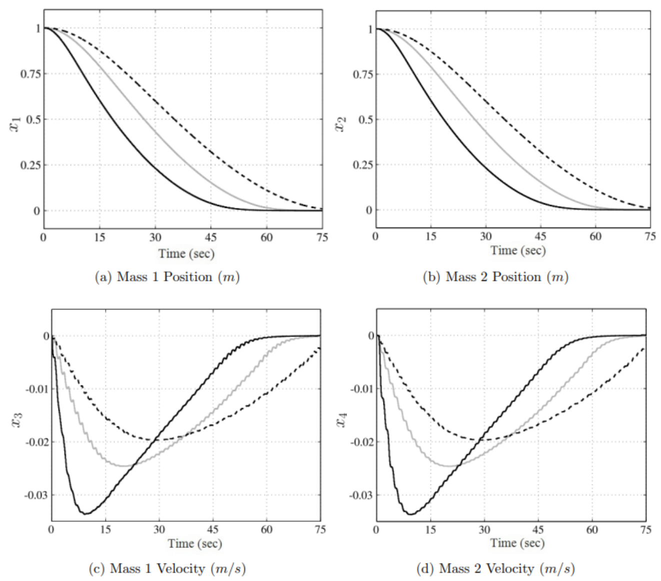


Figure 4: The state responses of the results from [4] (dashed line), [5] (gray line), and [6] (solid black line)

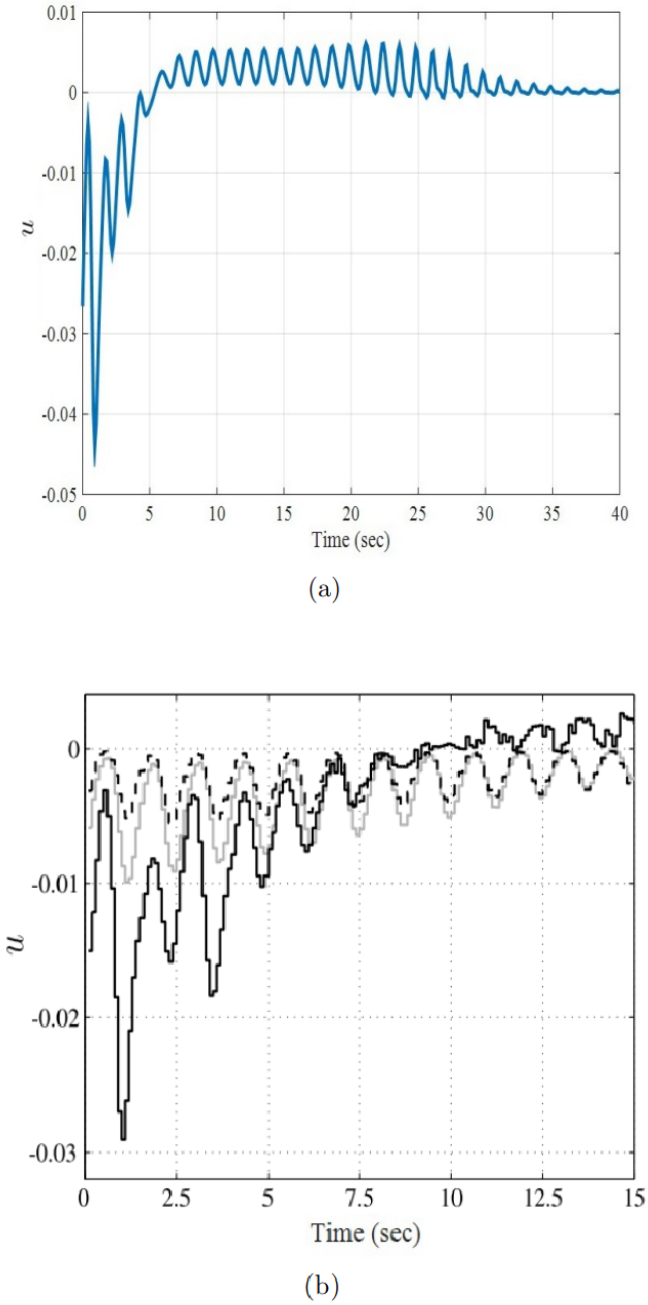
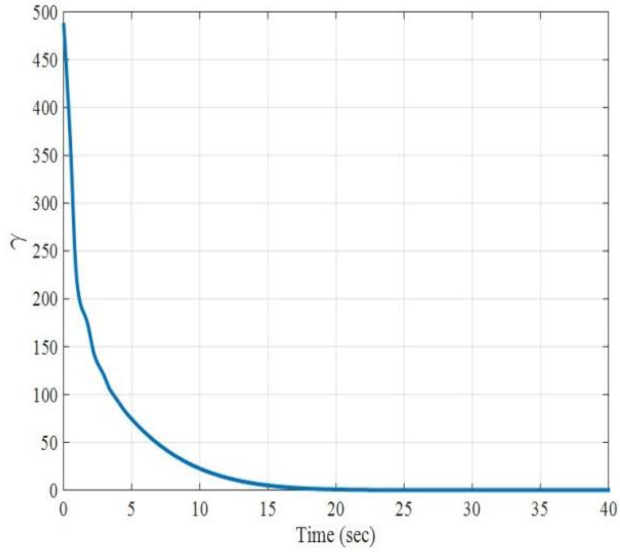
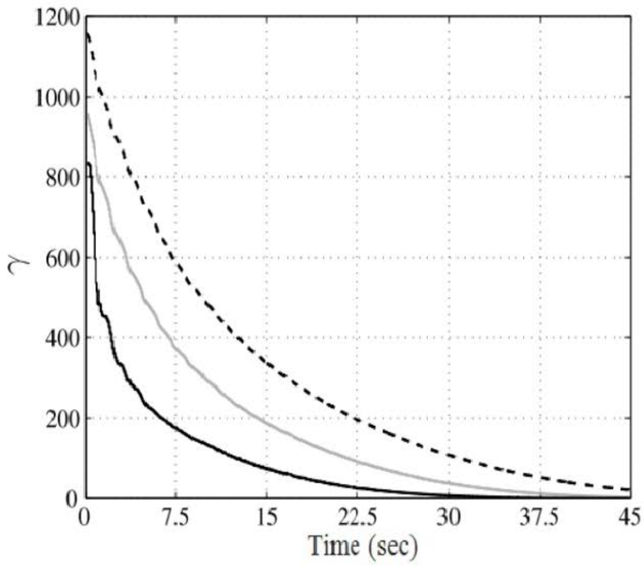


Figure 5: (a) Control signal of the proposed method, (b) [4] (dashed line), [5] (gray line), and [6] (solid black line).



(a)



(b)

Figure 6: Upper bound on the considered cost functions, (a) the proposed method, (b) [4] (dashed line), [5] (gray line), and [6] (solid black line

V. Conclusion

This paper presents a novel methodology to solve the problem of multi-objective model predictive control design. This method is proposed for linear parameter-varying systems. Multi-objective functions instead of single objective function are considered at each sampling time. This method leads to finding the trade-off between the objective functions. In order to solve the multi-objective optimization problem at each sampling time, the game theory and Nash bargaining solution are used to find the trade-off point. The Nash bargaining solution can find the trade-off point in game theories and can tune the weighting factors properly at each sampling time. The multi-objective optimization results are the solutions to a convex optimization problem based on linear matrix inequalities that are solved repeatedly at each sampling instant. The simulation results show the effectiveness of the proposed method that can be generally used for the control system design with more than one objective functions.

References

[1] Hernández Castellanos CI, Ober-Blöbaum S, Peitz S. *Explicit multiobjective modelpredictive control for nonlinear systems under uncertainty*. Int J Robust Nonlinear Control. 2020; 30:7593–7618. https://doi.org/10.1002/rnc.5197

[2] OTMANE CHERIF Abdelillah; BALANDIN Dmitry Vladimirovich. *Robust Model Predictive Control Design*. International scientific journal «Modern Information Technologies and IT-Education», [S.l.], v. 17, n. 4, dec. 2021. ISSN 2411-1473. Available at: <http://sitito.cs.msu.ru/index.php/SITITO/article/view/770>

[3] Y. Y. Baranyi, P. and P. V´arlaki, *Tensor product model transformation in polytopic model–based control*. Taylor & Francis Group, LLC, 2014.

[4] M. V. Kothare, V. Balakrishnan, and M. Morari, *Robust constrained model predictive control using linear matrix inequalities*, Automatica, vol. 32, no. 10, pp. 1361–1379, 1996.

[5] F. A. Cuzzola, J. C. Geromel, and M. Morari, *An improved approach for constrained robust model predictive control*, Automatica, vol. 38, no. 7, pp. 1183 – 1189, 2002.

[6] S. Yu, C. Bohm, H. Chen, and F. Allgower, *Stabilizing model predictive control for lpv systems subject to constraints with parameter-dependent control law*, in American Control Conference, 2009. ACC '09., June 2009, pp. 3118–3123.

[7] J. Engwerda, LQ Dynamic Optimization and Differential Games. Wiley, 2005.

[8] A. Bemporad and D. M. de la Pea, *Multiobjective model predictive control*, Automatica, vol. 45, no. 12, pp. 2823 – 2830, 2009.

[9] K. Binmore, A. Rubinstein, and A. Wolinsky, *The nash bargaining solution in economic modelling*, The RAND Journal of Economics, pp. 176–188, 1986.

[10] K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan, *A fast and elitist multiobjective genetic algorithm: Nsga-ii*, IEEE Transactions on Evolutionary Computation, vol. 6, no. 2, pp. 182 –197, apr 2002.

[11] P. Baranyi, *Tp model transformation as a way to lmi-based controller design*, Industrial Electronics, IEEE Transactions on, vol. 51, no. 2, pp. 387–400, 2004.

[12] L. Wang, Model predictive control system design and implementation using MATLAB R . Springer Science & Business Media, 2009.

[13] J. J. V. Garca, V. G. Garay, E. I. Gordo, F. A. Fano, and M. L. Sukia, *Intelligent multi-objective nonlinear model predictive control (imo-nmpc): Towards the on-line optimization of highly complex control problems*, Expert Systems with Applications, vol. 39, no. 7, pp. 6527 – 6540, 2012.

[14] Groetzner, P., Werner, R., 2021. *Multiobjective optimization under uncertainty: A multiobjective robust (relative) regret approach*. Eur. J. Oper. Res . doi: https://doi.org/10.1016/j.ejor.2021.03.068

[15] H. A. Hindi, B. Hassibi and S. P. Boyd, *Multiobjective* h_2/h_{infty} *optimal control via finite dimensional q-parametrization and linear matrix inequalities, American Control Conference 1998. Proceedings of the 1998*, vol. 5, pp. 3244-3249, 1998.

[16] A. Otmane Cherif, *Shimmy vibration control using robust model predictive control*, Int-Journal of Open Information Technologies, Vol 10, No 4, pp 24-30, 2022.