

Optimizing Model Base Predictive Control for Combustion Boiler Process at High Model Uncertainty



This work is licensed under a Creative Commons Attribution 4.0 International License

H. M. Osman*

Department of Chemical Engineering,
King Khalid University,
P.O.394, Abha, Saudi Arabia

doi: 10.15255/CABEQ.2016.867

Original scientific paper
Received: March 16, 2016
Accepted: July 21, 2017

This paper proposes a multi-objective evolutionary algorithm for optimizing model base predictive control (MBPC) tuning parameters applied to the boiling process. The multi-objective evolutionary algorithms are able to incorporate many objective functions that can simultaneously meet robust stability and performance that can satisfy control design objective functions. These promising techniques are successfully implemented to stabilise MBPC at the implications of different levels of model uncertainties.

The Pareto optimum technique is able to overcome the problem of trapping the standard genetic algorithms (SGAs) in the local optimum when using the LQ as the objective functions at the price of high model uncertainty. Introducing robust stability and performance objective functions has successfully improved the search procedure for MBPC tuning variables at high model uncertainty.

Key words:

model base predictive control, multi-objective evolutionary algorithm, control tuning

Introduction

MBPC is a control strategy based on the explicit use of a process model to predict the process outputs over a long period. When no process model mismatches are present, there are sufficient conditions to guarantee stability and feasibility of the controller. If the process model is uncertain, it is compulsory to consider the bounded uncertainties in the controller synthesis in order to guarantee robust stability, which is very difficult or impossible to be analytically calculated^{1,2}. Therefore, it was necessary to find an alternative solution to stabilise MBPC without conducting a difficult mathematical analysis. Genetic algorithms are promising in finding optimum tuning parameters for MBPC to meet robust control design. This type of advance control system is widely used in industry. Qin and Badgwell³ reported that the majority of MBPC applications (67 %) are in the area of refining, the original application field of MBPC. A significant proportion of the remainder can be found in the petrochemicals and chemicals industries. The main difference among MBPC strategies is in the kind of model used for the process modelling and the formulation of the cost-function that needs to be minimised.

One of the most popular MBPCs is Dynamic Matrix Control (DMC), during the last three decades

large number of DMC control design, analysis, and tuning methods have been proposed, which started with a systematic tuning procedure that presented by Cooper *et al.* when the model is identical to the process only^{4a,4b}.

An optimal tuning approach based on particle swarm optimisation, in which the Morari resiliency index and the condition number were applied as the performance measure, is proposed by Chu *et al.*⁵ Also, the tuning parameters were computed by searching for an optimal bandwidth that gave a trade-off between robustness and nominal performance⁶.

More recently, the tuning problem has been formulated as a two-degree-of-freedom (2-DOF) optimisation problem by Júnior, Martins and Kalid⁷. This framework is also employed to achieve better closed-loop responses, as measured by overshoots, settling times, and output oscillations with user-specified parametric uncertainties⁸.

Jorge *et al.*⁹ reviewed various techniques and available tuning guidelines for model predictive control. Nagrath *et al.*¹⁰ used default prediction horizon value of 10 for DMC controlling a state-space representation of a continuous-stirred tank reactor (CSTR) with excellent results.

The control horizon parameter affects how aggressive or conservative the control action is. Yamuna Rani and Unbehauen¹¹ suggested a default

*email: haitham.osman@gmail.com

value of 1 for the control horizon. Georgiou *et al.*¹² proposed setting the model horizon larger than the time required for the slowest open-loop process output response to reach 95 % of the steady state. Shridhar and Cooper¹³, and Wojsznis *et al.*¹⁴ proposed setting the model horizon equal to the final prediction horizon.

Rowe and Maciejowski¹⁵ used an LQG/LTR approach to tune an infinite horizon state-space MPC controller. Using that technique, the authors derived an expression for the output weights for minimum phase systems as the product of the transpose of the output matrix C multiplied by the output matrix. Rowe and Maciejowski¹⁵ provided another equation based on H_∞ shaping of the process model. This applies for nonstrictly proper cases. For strictly proper cases, the expression is the same as in the LQG approach. Both studies assume that the weight of change of inputs is 1.

Penalizing the rate of change yields a more robust controller but at the cost of the controller being more sluggish. Setting a small penalty or none whatsoever gives a more aggressive controller that is less robust. Hinde and Cooper¹⁶ used an approach to set the rate of change of inputs (suppression) weights based on the desired controller performance defined as a short rise time with 10–15 % overshoot.

The advantage of using an auto-tuning method is that the control engineer is not required to have a great amount of systems knowledge to initialize the tuning procedure. However, this comes at the price of computation time, since the engineer is now required to calculate two optimization procedures per time step rather than the one in an offline tuning strategy. However, modern computer systems have reduced computational time tremendously.

Future work in MBPC will focus on stability analysis, the development of data-driven techniques to perform the plant decomposition and feasibility of the computational feature¹⁷.

The objective of this paper was to find optimum tuning parameters for DMC for the boiling process to meet robust control design at the implication of different levels of model uncertainty.

Overview of standard genetic algorithm for tuning MBPC parameters

The basic concept of MBPC is that the future control action is evaluated by minimising a quadratic objective function:

$$J = \sum_{i=N_1}^{N_2} (y(t+i) - r(t+i))^2 + \lambda \sum_{i=0}^{Nu} (\Delta u(t+i-1))^2 \quad (1)$$

in which the error between the predicted output and the reference signal is reduced and balanced against

a term related to the rate of change of the control action. At every sampling interval the control signal in the case of model reference predictive control is calculated by minimising the objective function by searching for the control changes between time 0 and $Nu-1$. The variable $y(t+i)$ represents i steps ahead prediction of the output signal, $r(t+i)$ the set point at the time $t+i$, and $u(t+i-1)$ is the control signal. The values N_1 and N_2 are the minimum and the maximum of the prediction horizons, respectively (hence, $Np = N_2 - N_1$), and Nu denotes the control horizon, D is the difference operator $(1-z^{-1})$, parameters λ_1, λ_2 represent the control weightings. Equation 1 is the GPC quadratic cost function. This cost function is solved by using the Diophantine identity¹⁸ to obtain the prediction equation, and the control law as follows:

$$\Delta u = (G^T G + \lambda I)^{-1} G^T (r - f) \quad (2)$$

where u is the control single output, G is a square matrix, contains the step-response coefficients and is of dimension $(N_2 - N_1 + 1)$ by Nu , r is the setpoint, f is the response, I is the weighting matrix.

Standard Genetic Algorithms (SGAs) are stochastic search algorithms inspired by the principles of natural selection and genetics. The fundamental mode of operation is that a population of candidate solutions or individuals compete with each other for survival. Using a measure of fitness, stronger individuals are given a greater chance of contributing to the production of new individuals than weaker ones. The flowchart of a simple genetic algorithm is presented by Goldberg¹⁹. Genetic algorithms offer the potential to tune automatically the MBPC parameters. One of the important reasons for using SGAs to optimize MBPC tuning parameters is that the optimization problem often becomes non-convex in the presence of MBPC constraints and/or a nonlinear process, which makes the conventional optimization algorithm converge to a local optimum, resulting in poor tuning of MBPC and a degradation of performance. The problem of optimizing the MBPC tuning parameters is nonlinear and it is difficult to solve by analytical solution for the case of optimal control design methods (e.g. using LQC, H2 and H control design). Indeed, rarely is there an analytical solution to the exact problem the designer wishes to solve. The analytical solution becomes impossible, especially in the case of increasing process plant complexity and in the presence of constraints and disturbances.

The tuning of unconstrained SISO DMC is challenging because of the number of adjustable parameters that affect closed-loop performance. These include the following: a finite prediction horizon, Np ; a control horizon, Nu ; a control weighting, and a sample time t , in Control, Automation and Sys-

tems (ICCAS), 2014 14th International Conference. The first problem that needs to be addressed is the selection of an appropriate set of tuning parameters from among those available for DMC. Practical limitations often restrict the availability of sample time, t as a tuning parameter. The model horizon is also not an appropriate tuning parameter, since truncation of the model horizon misrepresents the effect of past moves in the predicted output, and leads to unpredictable closed-loop performance²⁰.

Therefore, the tuning parameter variables used in this paper are the two control weighting λ_1, λ_2 and the control horizon Nu .

The following is the general specification of SGAs applied to optimize MBPC prediction horizon and the control weightings tuning parameters.

Population size: all runs are initialized with similar random population of 100 members to ensure stability of the algorithm and fare comparison.

Using the random number generator is one of the important components of evolutionary algorithms to ensure population diversity. The random number generator uses a seed (i.e. similar seeds produce a similar set of numbers) to select the first number and then updates it for the second, and so forth. It is necessary to initialize the GA with similar population at each run to guarantee fair comparison between the optimization problems. Fitzpatrick *et al.*²¹ showed that, in general, increasing the population size while decreasing the samples per trial appears to improve the performance of genetic algorithms. However, this effect is moderated by the requirement that the algorithm performs a sufficient number of iterations to adequately explore the search space. This latter requirement places a limit on the population size when the genetic algorithm overhead is relatively high in comparison to the sampling cost.

Goldberg¹⁹ mentions that “At small population sizes the GA makes many errors of decision, and the quality of convergence is largely left to chance or to the serial fix-up of flawed results through mutation or other serial injection of diversity. At large population sizes, GAs can reliably discriminate between good and bad building blocks, and parallel processing and recombination of building blocks leads to the quick solution of even difficult deceptive problems.”

Chromosome encoding: The control decision variables are encoded using the binary system; the prediction horizon, and the control weighting are encoded as 16 bit and combined which leads to a 32-bit chromosome length.

Fitness assignment: After the objective function is calculated, the fitness function is scaled in

the range, then the population is ranked related to its fitness value.

Genetic operators: The single point crossover with probability of $P_c = 0.95$ is chosen as it proved to work in most difficult optimization problems, the single point mutation with $P_m = 0.01$ is used, and the population size is 100 individuals. The objective function is to reduce the MSE (Mean of Square Error) which proved to be a successful performance index for tuning PID²².

The combustion boiler process model

The combustion system of a coal-fired power plant boiler is shown in Figure 1. The main objectives of the combustion control system are to keep steam pressure stable and respond to the load changes rapidly, in order to achieve optimum combustion efficiency and keep furnace negative pressure stable.

There are three control loops, including those for main steam pressure, excess air coefficient, and furnace negative pressure. The input variables are coal mass flow rate, supply air flow rate and draft gas flow rate, the output variables are the main steam pressure, excess air coefficient, and furnace negative pressure, respectively. Generally, the dynamic model of the boiler combustion system can be written as follows²³:

$$\begin{bmatrix} y_1 \\ y_2 \\ y_3 \end{bmatrix} = \begin{bmatrix} G_{11}(s) & 0 & 0 \\ G_{21}(s) & G_{22}(s) & 0 \\ 0 & G_{32}(s) & G_{33}(s) \end{bmatrix} \cdot \begin{bmatrix} u_1 \\ u_2 \\ u_3 \end{bmatrix} \quad (3)$$

where: y_1 , y_2 and y_3 are the main steam pressure (MPa), oxygen content of flue gas, and furnace negative pressure (Pa), respectively. u_1 , u_2 and u_3 are coal mass flow rate (kg s^{-1}), supply air flow rate ($\text{m}^3 \text{s}^{-1}$), and draft gas flow rate ($\text{m}^3 \text{s}^{-1}$), respectively. The main steam pressure change is mainly caused by the coal flow changes and the main steam flow changes. In normal operation conditions, the variations of main steam flow are very small. Especially in steady conditions, the main steam flow can be considered stable. Thus, the transfer function can be written as a first order plus time delay (FOPTD) process model, which represents the relation between the main steam pressure y_1 and the coal mass flow rate u_1 as follows:

$$Gp(s) = \frac{0.44^{-93.2s}}{39.1s + 1} \quad (4)$$

where: $K = 0.44$, $T = -93.2$ and $\tau = 39.1$ are the gain, time delay, and time constant, respectively.

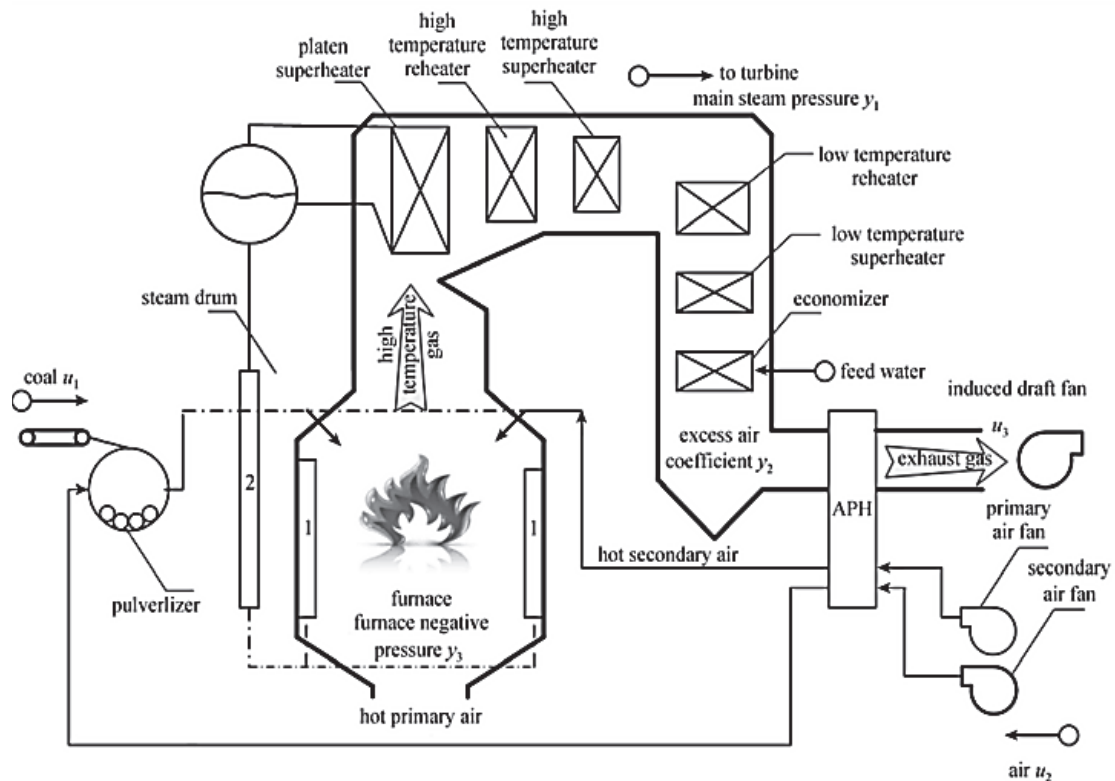


Fig. 1 – Power plant boiler combustion system²³

Tuning of the MBPC parameters at the implications of model mismatched

In industry, the specification of the process characteristics makes it difficult to obtain an accurate model of the detailed behaviour of the system. Generally, most chemical processes are nonlinear. Nonlinearity and complex dynamics make it impossible to model the real plant identically.

Much research has dealt with the MBPC when the model is identical to the real plant, and have ignored the uncertainty of the model. Neglecting model uncertainty leads to designing a controller which is too tight and likely to become unstable in the real operating environment. Although many researchers have addressed tuning of unconstrained and constrained MBPC for SISO and MIMO systems analytically, there are still no clear guidelines when the control model is not identical to the real plant. The stability and performance of MBPC in this case have proven to be a complicated issue and difficult to solve.

Standard Genetic Algorithms (SGAs) for tuning MBPC

The MBPC is used to control the boiling combustion model, the SGA parameters are the same as shown previously, the control objective function is to minimize the Mean of Square Error (MSE). Fig-

ures 2 and 3 show the simulation results; when the model is identical to the process plant, the SGA is able to tune the MBPC variables. However, at some values of tuning parameters it shows a slow response to achieve the set point at the end. When increasing the uncertainty of the model in the time delay to +0.07 %, it can be seen in Figure 3, that the SGA algorithm trapped in the local optimum resulted in poor tuning of the MBPC parameters and instability of the system at the price of model uncertainty, which is where the algorithm failed; the same initial random population was used to ensure same environment and all algorithms were terminated after the 10th generation. The computation time for the SGA lasted 1–2 seconds. Hence, SGA algorithm has proven the ability to tune MBPC when the model is identical to the plant but fails at model uncertainty.

The Bounded Input – Bounded Output (BIBO) stability is used for the MBPC, the state space description of the closed loop is obtained using MATLAB to determine the pole location. The closed loop (BIBO) stable if all the poles are inside the unit circle. However, the system could have a situation where is BIBO stable but not asymptotically stable.

The Pareto ranking approach

The previous case determined that SGA with MSE (Mean Square of the Error) single objective

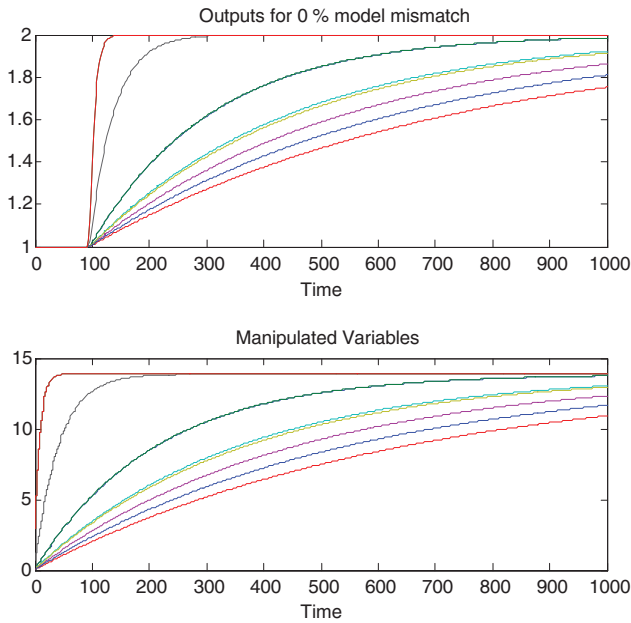


Fig. 2 – SGA tuning for MBPC at identical model

function is unable to tune MBPC when the control model is not identical to the real process.

Although, many methods are available for the stability analysis of linear, time invariant control systems including LQC, H_∞ , H_2 yield optimum stable control. However, for nonlinear systems and time varying systems, stability analysis may be extremely difficult or impossible, and the analysis of the stability of MBPC in the case of model/plant mismatch is very difficult or impossible, especially in the presence of constraints.

To overcome the drawbacks of SGAs combined with only MSE objective function, it is desirable to add a new function and combine a two objective functions in a single objective function, and then balance this function with weights.

The LQC design has proven a successful tool for stability of nonlinear control systems in analytical design. However, the use of SGAs with the LQC as objective function is a conventional approach to finding the minimum of the linear combination of variances of steady state and set point with the control output as follows objective function:

$$\min J = \lim_{k \rightarrow \infty} E[y_k^2 + \rho u_k^2] \quad (5)$$

The variance of the steady state actuator is the difference between the plant output and the set point, and it can be replaced by the MSE performance index to do the same work. The weighting of this objective function can provide the balance between the two functions and ensure that they have the same chance of being affected in the calculation

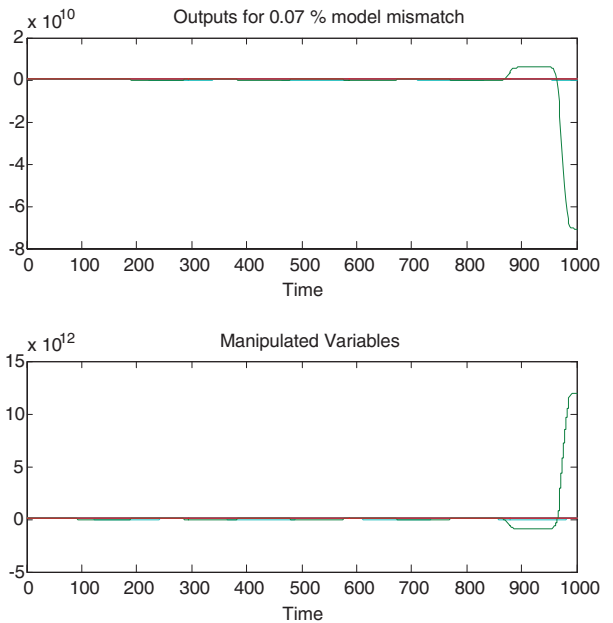


Fig. 3 – SGA tuning for MBPC when increasing model uncertainty to 0.07 %

of the SGA. Choosing the weighting variables can be the most difficult task. Initially, the weighting variable should provide the normalised to balance between the two objective functions. To find the value of the weighting variable ρ (sometimes called lambda) may need another optimisation routine in cascade, which could increase the computational time tremendously.

Non-dominant Sorting Genetic Algorithm (NSGA)

An alternative approach is to use the Pareto ranking technique to incorporate multi-objective functions. This way, there is no need to specify weighting to balance between MSE performance index and the control penalty. The non-dominant sorting genetic algorithm (NSGA-11) by^{24a, 24b} in Figure 4 is based on several layers of classification of the individuals. Before selection is performed, the population is ranked on the basis of domination (using Pareto ranking). All non-dominant individuals are classified into one category (with a dummy fitness value proportional to the population size). To maintain diversity of the population, these classified individuals are shared after all individuals have been compared with each other. To build a fair comparison, the previous specifications of GAs are used, population size of 100 individuals, single crossover of $P_c = 0.9$, and single mutation point of $P_m = 0.01$. The GA is terminated after 10 generations. Tables 1 and 2 show the results.

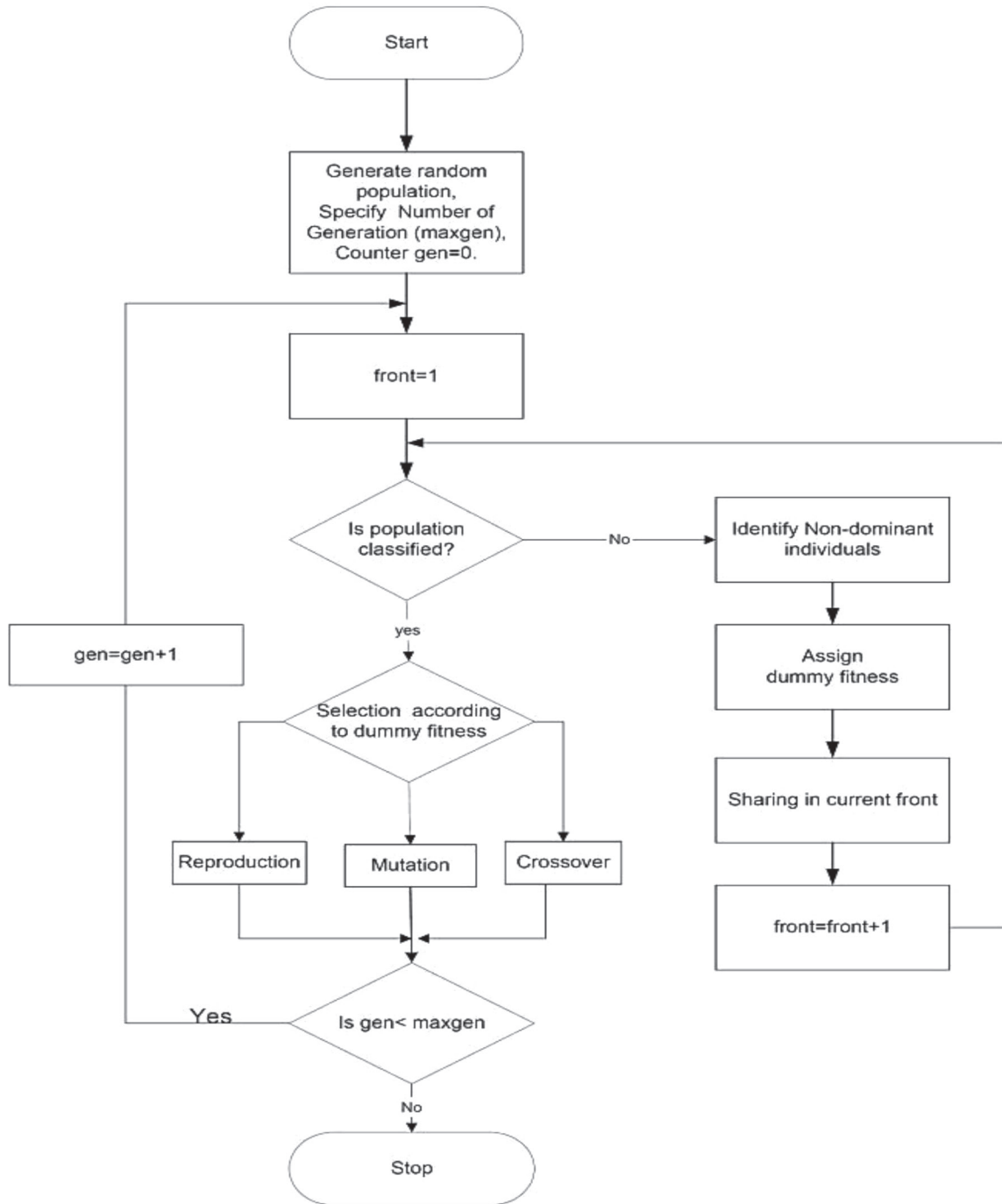


Fig. 4 – Flow diagram of the original NSGA^{24a,24b}

Implementing modified NSGA-II with constraints handling

In all experiments, the above GA configuration was used in addition to a sharing factor of 0.48, as well as elitism and a crowded comparison operator that keeps diversity of the solutions.

Unfortunately, there was no guarantee that the algorithm would find the global optima without

sticking in the local optima. Therefore, the NSGA-II was modified to test if it at least approached near global optima solutions, as follows: a different set of population members was used in each run, the number of iterations was increased, and the mutation rate in the evolutionary algorithm was increased. A small routine was added to ensure saving the best solution along running time of the algorithm, as follows:

```
{Store the best solution;
Obtain the neighbour solutions for the best solution;
Compare the best solution and best neighbour solution; }
```

The NSGA-II was also modified to handle constraints to avoid trapping in unstable solutions. The comparison study of using modified NSGA-II in Tables 1 and 2 at different model uncertainties revealed that NSGA-II was able to find a Pareto optimum front that covers a wider range of the search space.

Results and discussions

Successful implementation of evolutionary algorithms for the boiler process revealed that SGA was able to tune the MBPC up to 0.07 % model mismatch, but was trapped in the local optimum at the price of increasing model uncertainty.

Figures 5a–5f show the ability of the NSGA-II to tune the MPC at additive model uncertainty from 15 % of the time delay up to 105 %, where it was trapped and unable to tune the MPC. Figures 6a–6b show that the NSGA-II negative model uncertainty from –15 % to only –60 %, which is the case when the tuning variables trapped in the local optimum resulted in oscillation and instability.

Further investigations of increasing the model uncertainty to +105 % by using NSGA-II in conjunction with LQC objective functions showed that NSGA-II was able to find a Pareto optimum front that covered a wider range of the search space.

This provided multiple solutions which help in the synthesis of process control at high model uncertainty, compared with other control conventional design methods that give only a single solution and are not able to deal with model uncertainty. The simulation was terminated after 38.48 ± 0.14 seconds and 10 generations in Pentium i5 4GB RAM.

Comparing with the literature²⁵ the original NSGA-II was trapped in the local optimum at only +90 % of model uncertainty, which resulted in instability and poor tuning of the control system.

Yuanhao *et al.*²⁶ applied sliding mode predictive control system (SMPC) to boiler main steam pressure control system under model mismatch conditions when a set point change was introduced at time $t = 0$, and compared the results with traditional PID control and Smith Predictor (SP), which showed that both outputs of PID and SP were close to critically damped, but the SMPC produced a faster response than that given by the SP. Compared to the NSGA-II, it was not only able to outperform the SMPC in terms of faster response, but was also able

to tune the MPC at very high model uncertainty, in addition to giving a wide range of optimum solutions.

It is very common to have Model to Plant Mismatch (MPM) more than 60 %. In practices it is unfeasible technically or economically, to develop detailed first principle models (e.g. regression). One of the important reasons for MPC’s success in industry has been the ability of engineers to construct the required models efficiently from plant tests. In current real industrial application, non-linear models are derived from the input-output data. However, it will unavoidably contain significant bias and variance. The uncertainties need to be quantified and used in the controller design and analysis. The theory for doing this is still in the developmental stages, even for linear systems. However, the need for systematic tools to deal with them is clear as insights and heuristics developed for linear controllers do not apply to non-linear controllers.

In general, impulse- and step-response test descriptions are only equivalent when there is no uncertainty. If there is uncertainty, they behave rather

```
– Initialize population and generate random population
P- size Nind
– Evaluate objective values
For i=1 to Gen
– Assign rank based on Pareto dominance using non-
dominated sorting strategy
– Determine crowding distance between points on each
front of solutions (same rank)
– Generate offspring population Q- size Nind
– Binary tournament selection
– Recombination
– Mutation
– Evaluate objective values for the offspring population Q
and stability criteria
– Combine parent population P and offspring population
Q- size: 2*Nind
– Assign rank based on Pareto dominance using non-
dominated sorting strategy
– Determine crowding distance between points on each
front of solutions (same rank).
– Select Nind solutions to propagate to the next generation
(1st: elitist -biased towards lower ranks- 2nd: crowding
distance - bias less crowded solutions)
End loop
```

Fig. 5 – Modified NSGA-II with instability constraint Pseudo code

Table 1 – MBPC simulation and Pareto front under Positive Model Uncertainty

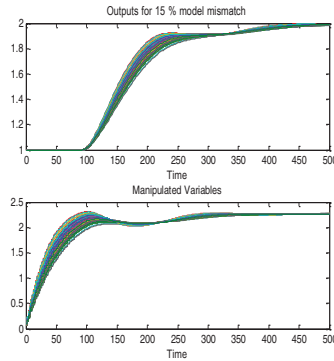
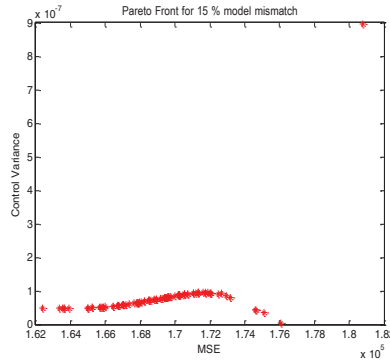
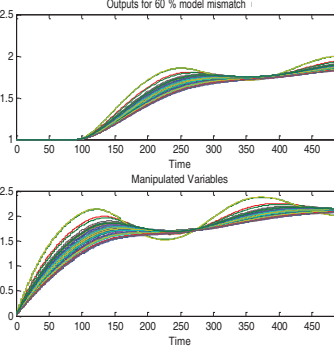
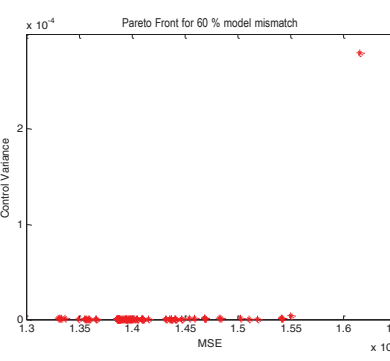
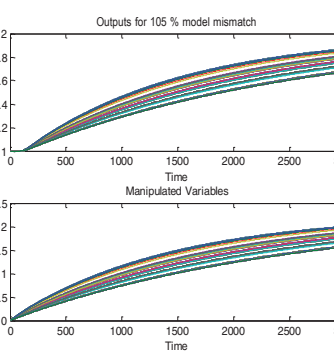
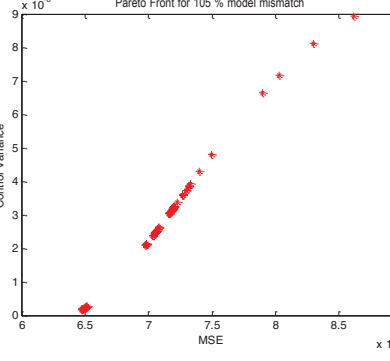
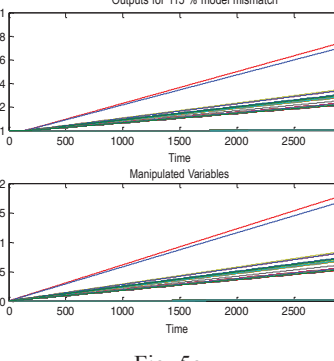
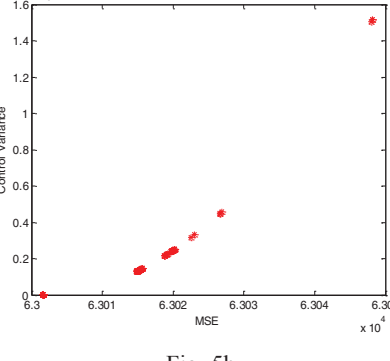
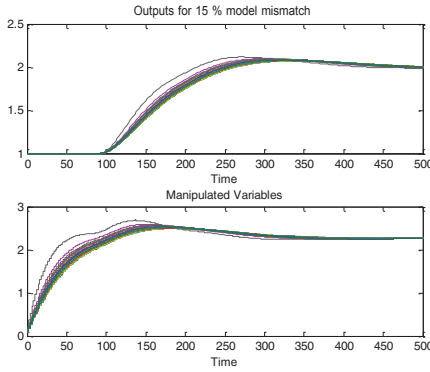
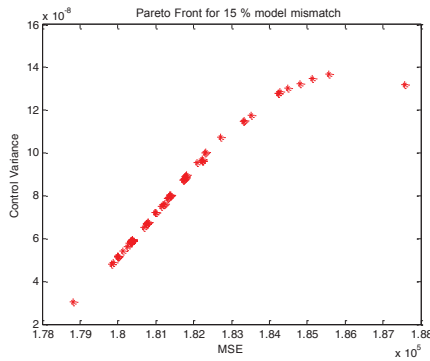
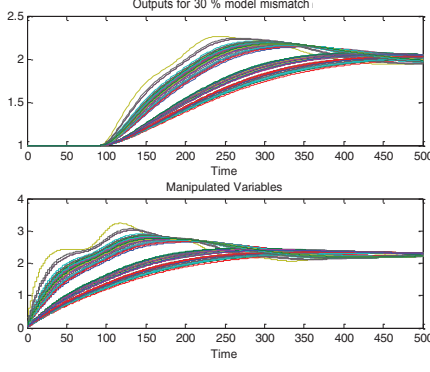
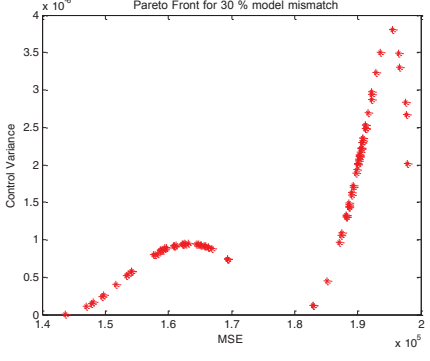
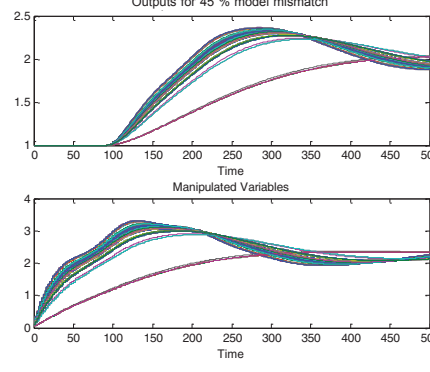
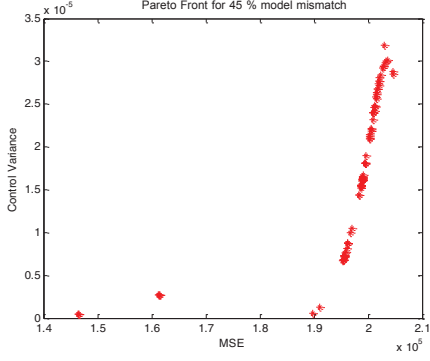
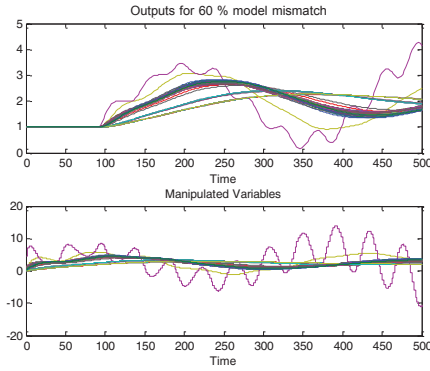
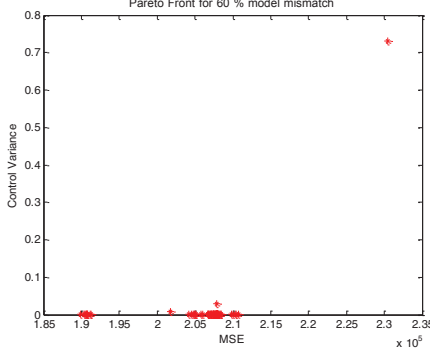
	MBPC simulation	Pareto front for 10 generations
Model mismatch +15 %	 <p>Fig. 5a</p>	 <p>Fig. 5b</p>
Model mismatch +60 %	 <p>Fig. 5c</p>	 <p>Fig. 5d</p>
Model mismatch +105 %	 <p>Fig. 5e</p>	 <p>Fig. 5f</p>
Model mismatch +115 %	 <p>Fig. 5g</p>	 <p>Fig. 5h</p>

Table 2 – MBPC simulation and Pareto front under Negative Model Uncertainty

	MBPC simulation	Pareto front for 10 generations
Model mismatch –15 %	 <p>Fig. 6a</p>	 <p>Fig. 6b</p>
Model mismatch –30 %	 <p>Fig. 6c</p>	 <p>Fig. 6d</p>
Model mismatch –45 %	 <p>Fig. 6e</p>	 <p>Fig. 6f</p>
Model mismatch –60 % (unstable)	 <p>Fig. 6g</p>	 <p>Fig. 6h</p>

differently²⁷. In order to arrive at a tight uncertainty description, both tests may have to be used simultaneously and further constraints may have to be imposed on the coefficient variations. In addition to the previously mentioned, Nafsun and Yusoff²⁷ showed that the gain mismatch has significant effect on MPC controller performance compared to time constant and time delay mismatch when testing MPM up to 70 %. For set-point tracking, MPC controller performance is bad in the presence of gain mismatch, but good in the presence of time constant and time delay mismatch.

Badwe *et al.*²⁸ showed that, in the time delay mismatch case, an underestimated (negative uncertainty) leads to oscillatory behaviour, and depending on the controller tuning, it may lead to instability. Overestimated (positive uncertainty) time delay in the model makes the controller conservative, leading to an over-damped response. They also applied MPM to Shell Control Problem (SCP) and the kerosene hydrofining unit, and noticed that the underestimate of the gain leads to a situation where the controller is aggressive as it computes large input moves in order to attain the specified set point target.

Further investigations of increasing the model uncertainty to more than +115.1 % by using modified NSGA-II in conjunction with LQC objective functions showed that NSGA-II was trapped in a local Pareto optimum and resulted in no tuning variables found for the MBPC at the price of increased model uncertainty. On the contrary, decreasing negative uncertainty in the delay time to –60 % resulted in instability of the system. This suggests the need to investigate modification of the control objective function to include H_2 , H_∞ and other control performance criteria.

Conclusions

SGA is able to successfully tune MBPC when the boiling process model is identical to the process plant, but fails at the price of increasing the model uncertainty to only +0.07 %, which resulted in poor control tuning and instability of the system, even though the objective function was the LQC.

The modified algorithm of NSGA-II separated the LQC objective functions and overcame the difficulties of specifying the weighting coefficients that balance between the objectives functions.

The comparison study of using modified NSGA-II at different model uncertainty revealed that NSGA-II was able to find a Pareto optimum front that covered a wider range of the search space. Further investigations of increasing the model uncertainty to +115 % and slight modification to NS-

GA-II in conjunction with LQC objective functions showed that modified NSGA-II was able to find a Pareto optimum front that covered a wider range of the search space. Compared to the literature²¹ the original NSGA-II was trapped in the local optimum at only +90 % of model uncertainty resulting in instability and poor tuning of the control system.

It can be seen from the simulation results that modified NSGA-II tuning algorithm applied to control system of the boiler main steam pressure to deal with the time delay was able to provide multiple solutions, which help in the synthesis of process control at high model uncertainty, compared with other control conventional design methods that give a single solution and are not able to deal with model uncertainty.

Further investigations of increasing the model uncertainty to +105 % by using modified NSGA-II in conjunction with LQC objective functions showed that modified NSGA was trapped in a local Pareto optimum and resulted in poor tuning and very slow response of MBPC at the price of increased model uncertainty. On the other hand, decreasing negative uncertainty in the delay time to only –60 % resulted in instability of the system.

A comparison with literature²⁵ revealed that the modified NSGA-II had the largest capacity to tune MBPC variables beyond the 90 % positive uncertainty.

A future investigation will be to modify the control objective functions to include H_2 , H_∞ and other control performance indices.

List of symbols and abbreviations

N_p	– Prediction horizon
NSGAII	– Non-dominated sorting genetic algorithm ii
N_u	– Control horizon
POF	– Pareto optimum front
SGA	– Standard Genetic Algorithm
$U(\cdot)$	– Process input signal or the control signal
$Y(\cdot)$	– Process output signal
W	– Control weighting
SISO	– Single input single output
CV	– Control variables
DMC	– Dynamic matrix control
EAS	– Evolution algorithms
F_f	– Shared fitness
G	– Matrix of step coefficients of the model
G_c	– Control transfer function
G_f	– Coefficient of process step response model
G_p	– Process transfer function

ACKNOWLEDGEMENTS

An old approach and condensed version of this paper has been presented at the Automation and Systems (ICCAS), 2014 14th International Conference on IEEE, as: OSMAN, H. LQ evolution algorithm optimizer for model predictive control at model uncertainty. *Control, Automation and Systems (ICCAS), 2014. IEEE 14th International pp. 1272 – 1277. The author would like to express his gratitude towards Kang-Hyun Jo and Gary Montague for their valuable comments that led to modification and improvement of the algorithm to handle increasing in the positive high model uncertainty.*

The author extends his appreciation to the Deanship of Scientific Research at King Khalid University for administrative and technical support for this work.

References

- Mayne, D. Q., Control of constrained dynamic systems, *European Journal of Control* **7** (2001) 87. doi: <https://doi.org/10.3166/ejc.7.87-99>
- Astrom, K. J., Wittenmark, B., *Computer Controlled Systems: Theory and Design*, Prentice Hall Professional Technical Reference, 1984.
- Qin, S. J., Badgwell, T. A., A survey of industrial model predictive control technology, *Control engineering practice* **11** (2003) 733. doi: [https://doi.org/10.1016/S0967-0661\(02\)00186-7](https://doi.org/10.1016/S0967-0661(02)00186-7)
- a. Dougherty, D., Cooper, D., A practical multiple model adaptive strategy for single-loop MPC, *Control engineering practice* **11** (2003) 141. doi: [https://doi.org/10.1016/S0967-0661\(02\)00106-5](https://doi.org/10.1016/S0967-0661(02)00106-5)
- b. Shridhar, R., Cooper, D. J., A tuning strategy for unconstrained SISO model predictive control, *Industrial & Engineering Chemistry Research* **36** (1997) 729. doi: <https://doi.org/10.1021/ie9604280>
- Chu, D., Forbes, M., Backstrom, J. U., Google Patents, 2013.
- Tran, Q. N., Ozkan, L., Backx, A., “Mpc tuning based on impact of modeling uncertainty on closed-loop performance,” in *Proceedings of the 2012 AICHE annual meeting*, Pittsburgh, PA, USA, 2012.
- Júnior, G. A. N., Martins, M. A., Kalid, R., A PSO-based optimal tuning strategy for constrained multivariable predictive controllers with model uncertainty, *ISA transactions* **53** (2014) 560. doi: <https://doi.org/10.1016/j.isatra.2013.12.019>
- He, N., Shi, D., Wang, J., Forbes, M., Backström, J., Chen, T., User Friendly Robust MPC Tuning of Uncertain Paper-Making Processes, in *International Symposium on Advanced Control of Chemical Processes (ADCHEM)*, 2015.
- Garriga, J. L., Soroush, M., Model predictive control tuning methods: A review, *Industrial & Engineering Chemistry Research* **49** (2010) 3505. doi: <https://doi.org/10.1021/ie900323c>
- Nagrath, D., Prasad, V., Bequette, B. W., Isermann, H., Model predictive control of open-loop unstable cascade systems, in *American Control Conference, 2000. Proceedings of the 2000, 2000*, 3747. doi: <https://doi.org/10.1109/acc.2000.876921>
- Rani, K. Y., Unbehauen, H., Study of predictive controller tuning methods, *Automatica* **33** (1997) 2243. doi: [https://doi.org/10.1016/S0005-1098\(97\)00134-9](https://doi.org/10.1016/S0005-1098(97)00134-9)
- Georgiou, A., Georgakis, C., Luyben, W. L., Nonlinear dynamic matrix control for high-purity distillation columns, *AIChE Journal* **34** (1988) 1287. doi: <https://doi.org/10.1002/aic.690340807>
- Shridhar, R., Cooper, D. J., A tuning strategy for unconstrained multivariable model predictive control, *Industrial & engineering chemistry research* **37** (1998) 4003. doi: <https://doi.org/10.1021/ie980202s>
- Wojsznis, W., Gudaz, J., Blevins, T., Mehta, A., Practical approach to tuning MPC Based on Practical Approach to Tuning MPC by Wojsznis, Gudaz, Mehta, and Blevins, published in the *Proceedings of the ISA 2001 Conference*, September 10-13, 2001, Houston, TX [11], *ISA transactions* **42** (2003) 149. doi: [https://doi.org/10.1016/S0019-0578\(07\)60121-9](https://doi.org/10.1016/S0019-0578(07)60121-9)
- Rowe, C., Maciejowski, J., Tuning MPC using H/sub/spl infin//loop shaping, in *American Control Conference, 2000. Proceedings of the 2000, 2000*, 1332. doi: <https://doi.org/10.1109/acc.2000.876717>
- Hinde, R. F., Cooper, D. J., A pattern-based approach to excitation diagnostics for adaptive process control, *Chemical engineering science* **49** (1994) 1403. doi: [https://doi.org/10.1016/0009-2509\(94\)85069-0](https://doi.org/10.1016/0009-2509(94)85069-0)
- Kumar, A. S., Ahmad, Z., Model Predictive Control (MPC) and its current issues in chemical engineering, *Chemical Engineering Communications* **199** (2012) 472. doi: <https://doi.org/10.1080/00986445.2011.592446>
- Clarke, D., Mohtadi, C., Tuffs, P., Generalized predictive control—part II extensions and interpretations, *Automatica* **23** (1987) 149. doi: [https://doi.org/10.1016/0005-1098\(87\)90088-4](https://doi.org/10.1016/0005-1098(87)90088-4)
- Goldberg, D. E., *Genetic algorithms in search, optimization, and machine learning*, Vol.13, Addison-Wesley, 1989, pp 90–101.
- Lundström, P., Lee, J. H., Morari, M., Skogestad, S., Limitations of dynamic matrix control, *Computers & Chemical Engineering* **19** (1995) 409. doi: [https://doi.org/10.1016/0098-1354\(94\)00063-T](https://doi.org/10.1016/0098-1354(94)00063-T)
- Fitzpatrick, J. M., Grefenstette, J. J., Genetic algorithms in noisy environments, *Machine learning* **3** (1988) 101. doi: <https://doi.org/10.1007/BF00113893>
- Osman, H., Modeling and PID control optimization for the crude oil network storage tanks, in *Control, Automation and Systems (ICCAS), 2016 16th International Conference (2016)* 826.
- Yuanhao, S. H. I., Jingcheng, W., Zhang, Y., Sliding mode predictive control of main steam pressure in coal-fired power plant boiler, *Chinese Journal of Chemical Engineering* **20** (2012) 1107. doi: [https://doi.org/10.1016/S1004-9541\(12\)60594-1](https://doi.org/10.1016/S1004-9541(12)60594-1)
- a. Deb, K., Pratap, A., Agarwal, S., Meyarivan, T., A fast and elitist multiobjective genetic algorithm: NSGA-II, *IEEE Transactions on Evolutionary Computation* **6** (2002) 182. doi: <https://doi.org/10.1109/4235.996017>
- b. Zitzler, E., Deb, K., Thiele, L., Comparison of multiobjective evolutionary algorithms: empirical results, *Evol. Comput.* **8** (2000) 173. doi: <https://doi.org/10.1162/106365600568202>
- Osman, H., LQ evolution algorithm optimizer for model predictive control at model uncertainty, in *Control, Automation and Systems (ICCAS), 2014 14th International Conference, 2014*, 1272.

26. Yuanhao, S., Jingcheng, W., Zhang, Y., Sliding mode predictive control of main steam pressure in coal-fired power plant boiler, *Chinese Journal of Chemical Engineering* **20** (2012) 1107.
doi: [https://doi.org/10.1016/S1004-9541\(12\)60594-1](https://doi.org/10.1016/S1004-9541(12)60594-1)
27. Bemporad, A., Morari, M., Robust model predictive control: A survey, *Robustness in identification and control*, Springer, 1999, pp. 207.
doi: <https://doi.org/10.1007/BFb0109870>
28. Nafsun, A., Yusoff, N., Effect of model-plant mismatch on MPC controller performance, *J. Appl. Sci.* **21** (2011) 3579.
doi: <https://doi.org/10.3923/jas.2011.3579.3585>
29. Badwe, A. S., Patwardhan, R. S., Shah, S. L., Patwardhan, S. C., Gudi, R. D., Quantifying the impact of model-plant mismatch on controller performance, *Journal of Process Control* **20** (2010) 408.
doi: <https://doi.org/10.1016/j.jprocont.2009.12.006>