Chapter 9
Conclusions

Please cite the book:
Maciej Ławryńczuk: Nonlinear Predictive Control Using Wiener Models: Computationally Efficient Approaches for Polynomial and Neural Structures. Studies in Systems, Decision and Control, vol. 389, Springer, Cham, 2022.

Abstract This Chapter summarises the whole book. Properties of the discussed computationally efficient MPC approaches are pointed out. Additionally, the benefits of using Wiener models in MPC, in particular of a neural structure, are stressed.

In this book, we have thoroughly studied simulation results of MPC algorithms in the input-output configuration for as many as seven benchmark processes, including the neutralisation reactor and the fuel cell. We have also considered simulation results of MPC algorithms in the state-space configuration for three benchmark processes. Basing on the obtained results, we may formulate the following observations:

1. Because of nonlinear nature of the considered processes as well as significant and fast set-point changes, the LMPC algorithm based on the parameter-constant linear model gives insufficient quality of control.
2. The simple MPC algorithms with on-line model linearisation and quadratic optimisation, i.e. MPC-SSL and MPC-NPSL approaches, make it possible to control the nonlinear processes, although the resulting trajectories are far from those obtained in the MPC-NO scheme with nonlinear optimisation repeated at each sampling instant. In general, the MPC-NPSL approach, in which the full nonlinear model is used for free trajectory calculation, gives better results than the MPC-SSL scheme, in which a linear approximation of the nonlinear model is used for this purpose.
3. The more advanced MPC algorithms with on-line trajectory linearisation and quadratic optimisation make it possible to obtain much better control quality than the simple MPC schemes with on-line model linearisation. The MPC-NPLT1 and MPC-NPLPT2 schemes, in which one linearisation is performed at each sampling instant, give slightly worse control quality than the reference MPC-NO algorithm, but the differences are really small. The MPC-NPLPT algorithm, in

which a few repetitions of trajectory linearisation and quadratic optimisation are possible at each sampling instant, gives process trajectories practically the same as those obtained in the computationally demanding MPC-NO approach in which nonlinear optimisation is used.
4. The classical MPC-inv algorithm, in which an inverse model of the nonlinear static part of the Wiener model is used to cancel the influence of process nonlinearity, works quite well in the case of a perfect model and no disturbances.
5. In the case of model errors and disturbances, the considered MPC algorithms with on-line model or trajectory linearisation work well, some problems are observed for the simplest MPC-SSL scheme. Of course, due to model imperfections and disturbances, control quality is lower when compared to the ideal case.
6. Unfortunately, the MPC-inv algorithm is not robust. For the considered benchmark processes, it gives very bad control quality in the case of an imperfect model and disturbances. The unwanted strong oscillations may be reduced by increasing the parameter λ, but in such a case, the whole algorithm is very slow, all process input and output trajectories are much slower than those obtained in the case of the MPC-NPLPT scheme.
7. Of course, the MPC-inv approach is possible only when the inverse model exists. Moreover, it may be practically used only when the nonlinear static blocks have only one input and one output. For more complex Wiener structures, the inverse models may be very complicated which makes implementation difficult or impossible.

As far as computational effort of the discussed MPC algorithms is concerned, we may observe the following issues:

1. Computational efficiency of all MPC algorithms with model and trajectory linearisation is twofold. Firstly, they need solving quadratic optimisation problems, complicated nonlinear optimisation used in the MPC-NO scheme is unnecessary. For correctly selected tuning coefficients, quadratic optimisation MPC problems have only one (global) solution; the multiple minima problem in nonlinear optimisation does not exist. Secondly, the computational time is shorter than in the case of the MPC-NO algorithm. For the analysed benchmarks, the computational time required by the discussed algorithms is only a fraction of that necessary in the MPC-NO scheme. For example, considering the input-output approach and the default horizons, that fraction is: approximately 40-60%, 20-30% and 1-2% for the SISO process, the MIMO process with two inputs and two outputs as well as the MIMO system with ten inputs and two outputs, respectively. For the state-space configuration, these numbers are better: some 20-30%, 10-15% and 0.5-2%.
2. The control horizon has a major impact on the calculation time, the prediction one has a much lower influence. All discussed MPC algorithms work well for long horizons.
3. The computational time of the discussed MPC algorithms does not grow significantly when the number of process manipulated variables increases, which is not true in the case of the MPC-NO approach.

Computational efficiency may be improved using parameterisation of the calculated sequence of the decision variables using Laguerre functions:

1. The parameterisation approach may be used in all discussed MPC algorithms, including the MPC-NO scheme with nonlinear optimisation and the computationally efficient methods with on-line model or trajectory linearisation.
2. The MPC algorithms with parameterisation are particularly useful in the case of processes with complex dynamics which require long control horizons.
3. As a result of parameterisation, the number of decision variables of the MPC optimisation task is reduced.
4. The higher the number of the Laguerre functions, the better the quality of control.
5. For the discussed benchmark process with complex dynamics, the MPC-NPLPT-P algorithm with parameterisation (the optimisation problem has only $n_{\mathrm{L}} = 20$ decision variables) gives practically the same results as the classical MPC-NPLPT algorithm without parameterisation (the optimisation problem has as many as $N_{\mathrm{u}} = 100$ decision variables).
6. For the discussed benchmark process, the rudimentary MPC-NPLPT algorithm requires approximately more than 30% longer optimisation time than the MPC-NPLPT-P algorithm.

For prediction in MPC, Wiener models are recommended. It is because such models can approximate very well properties of many processes. Furthermore, due to the model's specialised structure, implementation of the discussed MPC algorithms is relatively simple. Implementation details for as many as six input-output Wiener structures and three state-space ones are discussed. In the state-space configuration, a very efficient prediction model is used to guarantee offset-free control, much easier and better than the classical approach in which the augmented state disturbance model is used. For two technological processes, i.e. a neutralisation reactor and a fuel cell, effectiveness of polynomials and neural networks used in the nonlinear static part of the Wiener model is thoroughly compared. In general, it turns out that the neural Wiener model outperforms the polynomial one in terms of the number of parameters and accuracy. Polynomials cannot give a comparable modelling accuracy to that possible when the neural approach is used and lead to numerical problems when their degree is high. Hence, the use of neural Wiener models is recommended.

As far as future research is concerned, the following issues are worth considering:

1. Although in this book MLP neural networks with one hidden layer are successfully used in all considered neural Wiener models, it is possible to use other types of neural networks.
2. Alternative, more complex cascade models may be used in MPC algorithms, e.g. parallel ones.
3. In this book, we consider classical process control benchmarks described by ordinary differential equations (or their discrete-time versions). It may be an interesting idea to develop computationally efficient MPC algorithms for fractional-order Wiener systems.
4. In this book, model or trajectory linearisation is used to formulate quadratic optimisation tasks in place of computationally demanding nonlinear ones. An

interesting alternative is to use the Koopman operator rather than on-line linearisation.

5. In this book, Laguerre functions are used to parameterise the calculated decision vector which makes it possible to reduce the number of the decision variables. Alternative approaches to parameterisation may be considered.