Feasible Real-time Nonlinear Model Predictive Control

Matthew J. Tenny* and James B. Rawlings†
Department of Chemical Engineering
University of Wisconsin-Madison
Madison, WI 53706

Rahul Bindlish‡
Dow Chemical Company
Freeport, TX 77541

Abstract

This paper discusses an algorithm for efficiently calculating the control moves for constrained nonlinear model predictive control. The approach focuses on real-time optimization strategies that maintain feasibility with respect to the model and constraints at each iteration, yielding a stable technique suitable for suboptimal model predictive control of nonlinear process. We present a simulation to illustrate the performance of our method.

Keywords

Nonlinear processes, Model predictive control, Suboptimal control, Optimization strategies, Feasiblity

Introduction

Model predictive control (MPC) has become a well-established tool for advanced control applications, now one of the most prominent industrial control strategies. For many applications, linear MPC has proven to be a sufficient control tool. However, many processes exist that possess such a high degree of nonlinearity that linearized MPC produces inadequate results. For these processes, nonlinear model predictive control (NMPC) has been proposed. Some of the major challenges of NMPC are the solution of a global optimization problem in real-time and the appropriate handling of process constraints. In this paper, we present a method for real-time application of constrained NMPC with the possibility of running suboptimally, if required by a short sampling time.

Recent Advances in NMPC

In NMPC, stability is guaranteed for the finite horizon problem by applying a terminal constraint. Originally, the terminal state was required to be at the origin (Keerthi and Gilbert, 1988; Mayne and Michalska, 1990). The historical trend in the research that followed had the common theme of relaxing the constraints for an easier optimization formulation. Termination in a neighborhood around the origin with a terminal penalty is becoming a more popular formulation (Michalska and Mayne, 1993; Parisini and Zoppoli, 1995; Nicolao et al., 1998; Chen and Allgöwer, 1998a).

Less stringent criteria for stability were developed by Scokaert et al. (1999) and Chen and Allgöwer (1998b), who require only a decrease in the cost function at every time for stability, resulting in a suboptimal control law. The alternative formulation is that any controller that yields a closed-loop trajectory that ends in the terminal region is stable.

After the advances in NMPC theory, the next challenges came from the desire to calculate the control moves on-line in real-time. Significant effort has been made in reducing the computational burden of integrating the model over time; Bock et al. (1999) have investigated the use of simultaneous direct multiple shooting methods in which the solution to an ODE is determined at all points simultaneously. This method has been demonstrated to be faster than the traditional method of integration in series (Nagy et al., 2000).

Significant research also has been performed on the optimization approach utilized by the regulator. Some investigators have focused on global optimization, specifically genetic algorithms (Staus et al., 1996; Onnen et al., 1997; Rauch and Herremoës, 1999). However, this approach tends to be slow, and thus not implementable in real-time unless the time constants for the process are large.

Others have chosen to increase the speed of local optimization methods by tailoring them to take advantage of the specific structure of the MPC formulation. The approach uses an interior point method to solve a sequential quadratic programming problem (SQP) that, due to causal structure of the model, has a banded or almost block diagonal structure (Rao et al., 1998; Albuquerque et al., 1999). This method has been successful in easing the computational burden of optimizing large systems, and has recently come into favor with researchers.

This article describes the desired qualities of a real-time NMPC algorithm that can be run suboptimally, if required. We then discuss the method for achieving the desired objectives. Finally, we demonstrate our method on a nonlinear example.

Algorithm Requirements

We employ a typical ordinary differential equation model throughout this discussion. The process model has the form

$$\frac{dx}{dt} = f(x, u) \tag{1}$$

that, when integrated, becomes the discrete model

$$x_{i+1} = F(x_i, u_i) = x_i + \int_{t_i}^{t_{i+1}} f(x(\tau), u_i) d\tau \tag{2}$$

*tenny@bevo.che.wisc.edu
†jbraw@bevo.che.wisc.edu, author to whom correspondence should be addressed
‡RBindlish@dow.com

in which i represents the sampling time and a zero-order hold is assumed for the inputs.

One of the strengths of model predictive control is its ability to handle constraints in the regulator. The finite horizon regulator problem we consider is

$$\begin{aligned} \min_{u_i} \quad & \Phi(u_i, x_0) = \sum_{i=0}^{N} L(x_i, u_i) + h(x_N) \\ \text{subject to: } & x_N \in W_\alpha \\ & x_{i+1} = F(x_i, u_i) \\ & x \in X, u \in U \end{aligned} \tag{3}$$

Here, a terminal penalty is added to approximate the cost of the infinite horizon problem from the end of the trajectory in the terminal region. The approximation is valid because the terminal region is constructed in such a way that the nonlinear plant is not significantly different from the linear approximation in that neighborhood of the steady-state target. The problem is also constrained in the states and inputs. We assume the existence of a sequence of control moves that steers the state to setpoint asymptotically.

It is not necessary, however, to find the global optimum of Equation 3. Instead, asymptotic stability is guaranteed provided that the trajectory of states terminates in the region W_α (Scokaert et al., 1999). Therefore, the horizon length N need only be long enough to satisfy this restriction and the cost function does not need to be minimized for the control technique to reach the set-point. This concept eases the computation since a *suboptimal* set of control moves provides a stable controller. Therefore, the optimizer can stop early if the sampling time is small, and the process can still be controlled.

Algorithm Description

In the last section, we developed the requirements for regulating a nonlinear system with finite horizon MPC. Now, we must address the issue of solving the optimization problem according to our requirements. For this optimization, a sequential quadratic programming (SQP) technique is used. We exploit the structure of the problem to speed up the computation to run in real-time.

First, we present the fundamentals of the SQP method. Suppose we wish to solve the following problem:

$$\begin{aligned} \min_{w} \quad & h(w) \\ \text{subject to: } & c(w) = 0 \\ & d(w) \geq 0 \end{aligned} \tag{4}$$

The method reduces the problem in Equation 4 to a series of quadratic programs. Quadratic programs are well-studied and quickly solved with available methods, making SQP methods a suitable choice for nonlinear problems.

We present two methods for performing the quadratic programming approximation; the first does not have quadratic convergence properties, but may be easier to set up, depending on the structure of $h(w)$. The second has quadratic convergence properties for points near the solution, but the set-up may be computationally intensive.

Straightforward Formulation. First, define the superscript j to represent an iteration of the SQP method. In this formulation, we approximate $h(w)$ as a quadratic function around the current iterate w^j. We then compute a linear approximation to the constraints $c(w)$ and $d(w)$ around w^j. Defining $p = w - w^j$, we solve

$$\begin{aligned} \min_{p} \quad & \frac{1}{2} p^T \nabla^2 h(w^j) p + \nabla h(w^j)^T p \\ \text{subject to: } & \nabla c(w^j)^T p + c(w^j) = 0 \\ & \nabla d(w^j)^T p + d(w^j) \geq 0 \end{aligned} \tag{5}$$

No knowledge of the Lagrange multipliers by the user is required to form Equation 5. However, no special local convergence properties exist for this problem. It converges, but not quadratically near the solution. Quadratic convergence is important only when the initial guess is good enough that the minimizer is known to be close to the open loop prediction.

Quadratic Convergence Formulation. In order to obtain quadratic local convergence properties, the Lagrangian is first defined:

$$\mathcal{L}(w, \lambda) = h(w) - \lambda^T \begin{bmatrix} c(w) & d(w) \end{bmatrix} \tag{6}$$

The Hessian of the Lagrangian with respect to the variables is denoted by

$$\mathcal{H}(w, \lambda) = \nabla^2_{ww} \mathcal{L}(w, \lambda) \tag{7}$$

By the first-order KKT conditions, Equation 5 is equivalent to

$$\begin{aligned} \min_{w} \quad & \frac{1}{2} p^T \mathcal{H}(w^j, \lambda^j) p + \nabla h(w^j)^T p \\ \text{subject to: } & \nabla c(w^j)^T p + c(w^j) = 0 \\ & \nabla d(w^j)^T p + d(w^j) \geq 0 \end{aligned} \tag{8}$$

The solution to this problem is equal to a step of Newton's method (Nocedal and Wright, 1999), which is quadratically convergent in the region near the solution. This method requires an estimate of the Lagrange multipliers, which may not be estimated well until a few iterations have been performed. Also, a second order approximation of the constraints are required, which may be computationally intensive. We now relate the method of solution to the nonlinear MPC problem structure.

Dense Hessian SQP

In this naïve approach, we substitute Equation 2 into Equation 3 and solve it using an SQP method. We need first to define $w^j = u^j$. The equality constraints $c(w)$ are eliminated. The function that must be minimized, $h(w)$, is a complex nonlinear function. To form the quadratic approximation of $h(w)$, the value of $\nabla^2 h(w^j)$ is required. It can be seen that each x_i relies on all the $u_{i-k}, k = 1, 2, \ldots, i$. In fact, the last term relies on *all* of the u_i. The implications of this fact are that the Hessian of $h(w)$ is *dense*, making it difficult to take advantage of the specific architecture of the MPC problem in this formulation. To handle the MPC problem more efficiently, we propose a different approach.

Banded Hessian SQP

The banded Hessian SQP approach for MPC is described by Rao, Wright, and Rawlings (Rao et al., 1998) for the case of linear MPC. However, we can naturally extend it to the nonlinear case with a few modifications.

The key difference in the banded Hessian approach is not to plug the model equation into the Equation 3. Instead, it is left as an explicit equality constraint. Note that the constraint matrix is highly structured due to the causality of *only* the past state and input on the current state. In the quadratically convergent method, we require the Hessian of $c(w)$ as well. The differentiation could be performed by finite differences, which is slow and innaccurate, or an approximate Hessian could be calculated. However, popular Hessian update strategies, such as the BFGS update, destroy the banded structure of the Hessian, yielding instead a dense matrix. Until a more sophisticated update strategy is employed, it may be more suitable to solve Equation 8 for real-time applications.

The terminal region may be calculated offline (Tenny, 2000). However, we exclude the terminal region constraint in the quadratic program. This is done for a number of reasons:

1. If the horizon length N is chosen to be too short, the problem is infeasible.
2. If the horizon length is nominally long enough to reach the terminal region, the closed-loop solution may differ appreciably from the open-loop prediction.
3. Ellipsoidal constraints in quadratic programs are not exact; the constraint would be approximated.

A banded QP solver is then used for the structured optimization. The method of solution is an interior point method (Rao et al., 1998) that has been geared for the MPC structure. The cost of this approach is linear with respect to horizon length N, compared to cubic growth for the dense Hessian approach.

A solution to the approximate problem is then calculated. Instead of solving the nonlinear problem, the solution to the quadratic program points in a direction of objective function decrease. The next iterate is found using a trust region constraint.

Feasibility

One of the desired properties of the optimization algorithm is feasibility at all times with respect to the constraints. State inequalities are handled as soft constraints; they may be violated, but a term is added to the cost function to penalize the violation of such a constraint. The integration of the model can be accomplished using multiple shooting methods (Bock et al., 1999). We now describe how we maintain feasibility with respect to the input constraints and the equality constraints at all times so that we may terminate the optimization at any point to run the regulator suboptimally.

To guarantee feasibility of the initial guess w^0, we generate each u_i^0 as follows:

- In the case of small or no disturbances, the result from the previous open-loop prediction is feasible and becomes the initial guess for the current regulator problem.
- For startup or large disturbances, u_i^0 is determined based on the feedback law $u_i^0 = Kx_i^0$ in which K is the linear quadratic regulator feedback gain of the system linearized about the origin. If $Mu_i^0 > m$, we simply take those elements of u_i^0 that violate the constraint and clip them such that they meet the constraint. We maintain an initial guess of a feedback law in case large unmeasured disturbances are present and previous iterates are no longer valid. Alternatively, a local constrained linear MPC problem can be solved and one can use its solution as an initial guess for the regulator problem.
- Each x_i^0 is generated by substituting u_{i-1}^0, x_{i-1}^0 into Equation 2.

We now have a w^0 that satisfies the input and equality constraints.

The linearized system of constraints is formed and a quadratic program is solved. The state variables $\bar{x}$ from the result $\bar{w}$, are discarded and replaced with the states resulting from injecting the inputs $\bar{u}$ into the nonlinear model. If the cost function increases, we refine the trust region or line search method and regenerate the state predictions via the nonlinear model. However, if the cost function decreases, the method yields a new iterate w^1 that is feasible with respect to both the input and equality constraints *and* has a lower cost function. We now repeat the process using w^1 as the new initial guess. The algorithm repeats this process until the next sampling time is reached (suboptimal MPC) or until the iterates converge ($\|w^j - w^{j-1}\| \leq \delta$).

Grade	Polymer production rate (kg/min)	MMA Mole fraction in copolymer	Copolymer viscosity ($10^{-6}m^3/kg$)
A	0.3	0.60	35000
B	0.25	0.75	36725

Table 1: Product grades for MMA-VA copolymerization.

Once the iterates converge, the resulting vector w^{final} is the minimizer of the cost function without a terminal region constraint. Now, the final state in the w^{final} vector is x_N^{final}. We check to see if $x_N^{\text{final}} \in W_\alpha$. If it is, then the algorithm terminates. If not, then the horizon length N is not long enough. Therefore, N is increased (Rao et al., 1998) and the initial guess is the solution w^{final} with new inputs generated by Kx_N^{final}, etc. The initial guess is still feasible, and it takes advantage of the work of the previous optimization. We ignore the case of degenerate terminal regions for the purposes of this discussion.

An Example Process

We investigate the process of MMA-VA copolymerization as described by Bindlish (1999), which consists of a well-mixed reactor followed by a product separator. Several grades of polymer, characterized by the mole fraction of monomer A (MMA) in the copolymer product and the intrinsic viscosity of the copolymer product, are manufactured using this process. Depending on the demands of the market, the desired copolymer viscosity and composition are varied during operation. The feed to the reactor consists of the monomers (MMA and VA), initiator (AIBN), transfer agent (acetaldehyde) and inhibitor (m-dinitrobenzene) dissolved in a solvent (benzene). To remove heat released by the polymerization reaction, a coolant is employed. The polymer product is then separated from the unreacted hydrocarbons in a downstream separator. The reactor and separator are represented by a physical model based on first principles. The model has 15 states, 3 inputs, and 7 output variables, of which only three have set-points. The sampling time for this process is 5 minutes and the prediction horizon is 20 time steps (100 minutes). We wish to switch from product grade A to grade B, with output set-points as shown in Table 1. Figure 1 shows the output variables during the simulation for both the proposed nonlinear controller and nominal linear model predictive control based on a linearization of the nonlinear model at Grade B. In the case of the linear model predictive controller, the system becomes unstable. We note that the nonlinear model predictive controller returns a local optimal solution in this scenario because the computation time is smaller than the sampling time for this example.

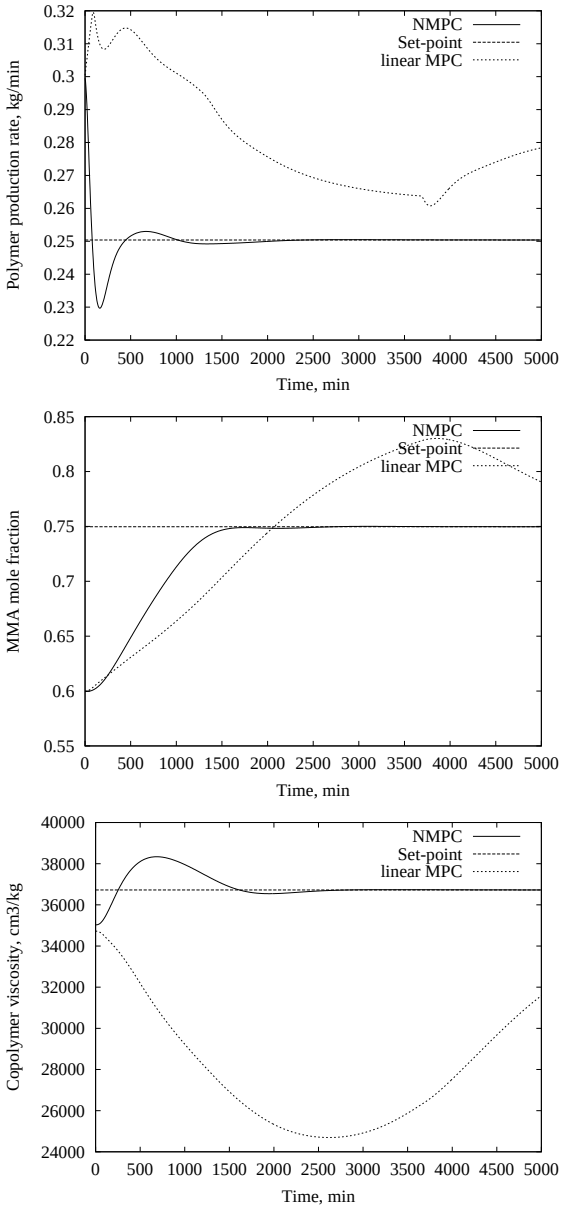

Figure 1: Output of transition from Grade A to B.

Conclusion

In this paper, we have devised and demonstrated a prototype algorithm for NMPC that can run suboptimally, if required. The method is capable of stabilizing nonlinear systems and obtaining optimal performance in real-time. Future directions for this research include real-time nonlinear state estimation.

References

Albuquerque, J., V. Gopal, G. Staus, L. T. Biegler, and E. B. Ydstie, "Interior Point SQP Strategies for Large-scale, Structured Process Optimization Problems," *Comput. Chem. Eng.*, **23**(4), 543–554 (1999).

Bindlish, R., *Modeling and Control of Polymerization Processes*,

PhD thesis, University of Wisconsin-Madison (1999).

Bock, H. G., M. Diehl, D. Leineweber, and J. Schlöder, Efficient Direct Multiple Shooting in Nonlinear Model Predictive Control, In Keil, F., W. Mackens, H. Voss, and J. Werther, editors, *Scientific Computing in Chemical Engineering II, Volume 2: Simulation, Image Processing, Optimization, and Control*. Springer, Berlin, Heidelberg (1999).

Chen, H. and F. Allgöwer, "A Computationally attractive nonlinear predictive control scheme with guaranteed stability for stable systems," *J. Proc. Cont.*, **8**(5–6), 475–485 (1998a).

Chen, H. and F. Allgöwer, Nonlinear Model Predictive Control Schemes with Guaranteed Stability, In Berber, R. and C. Kravaris, editors, *NATO ASI on Nonlinear Model Based Process Control*, pages 465–494. Kluwer (1998b).

Keerthi, S. S. and E. G. Gilbert, "Optimal Infinite-Horizon Feedback Laws for a General Class of Constrained Discrete-Time Systems: Stability and Moving-Horizon Approximations," *J. Optim. Theo. Appl.*, **57**(2), 265–293 (1988).

Mayne, D. Q. and H. Michalska, "Receding Horizon Control of Nonlinear Systems," *IEEE Trans. Auto. Cont.*, **35**(7), 814–824 (1990).

Michalska, H. and D. Q. Mayne, "Robust Receding Horizon Control of Constrained Nonlinear Systems," *IEEE Trans. Auto. Cont.*, **38**(11), 1623–1633 (1993).

Nagy, Z., R. Findeisen, M. Diehl, F. Allgöwer, H. G. Bock, S. Agachi, J. P. Schlöder, and D. Leineweber, Real-time Feasibility of Nonlinear Predictive Control for Large Scale Processes—a Case Study, *IST Report No. 2000-4*, Institut für Systemtheorie technischer Prozesse, Universität Stuttgart (2000).

Nicolao, G. D., L. Magni, and R. Scattolini, "Stabilizing Receding-Horizon Control of Nonlinear Time-Varying Systems," *IEEE Trans. Auto. Cont.*, **43**(7), 1030–1036 (1998).

Nocedal, J. and S. J. Wright, *Numerical Optimization*. Springer-Verlag, New York (1999).

Onnen, C., R. Babuska, U. Kaymak, J. M. Sousa, H. B. Verbruggen, and R. Isermann, "Genetic Algorithms for Optimization in Predictive Control," *Control Eng. Practice*, **5**(10), 1363–1372 (1997).

Parisini, T. and R. Zoppoli, "A receding-horizon regulator for nonlinear systems and a neural approximation," *Automatica*, **31**(10), 1443–1451 (1995).

Rao, C. V., S. J. Wright, and J. B. Rawlings, "On the application of interior point methods to model predictive control," *J. Optim. Theo. Appl.*, **99**, 723–757 (1998).

Rauch, W. and P. Herremoës, "Genetic Algorithms in Real Time Control Applied to Minimize Transient Pollution from Urban Wastewater Systems," *Water Research*, **33**(5), 1265–1277 (1999).

Scokaert, P. O. M., D. Q. Mayne, and J. B. Rawlings, "Suboptimal Model Predictive Control (Feasibility Implies Stability)," *IEEE Trans. Auto. Cont.*, **44**(3), 648–654 (1999).

Staus, G. H., L. T. Biegler, and B. E. Ydstie, Adaptive control via non-convex optimization, In Floudas, C. and P. Pardolas, editors, *Nonconvex optimization and its applications*, pages 1–19. Kluwer (1996).

Tenny, M. J., Practical Development in Nonlinear Model Predictive Control (2000). PhD Preliminary Report, University of Wisconsin-Madison.