Proceedings of the 38th
Conference on Decision & Control
Phoenix, Arizona USA • December 1999

Constrained model predictive control with on-line input parametrization

Edwin T. van Donkelaar Okko H. Bosgra Paul M.J. Van den Hof[1]
Mechanical Engineering Systems and Control Group
Delft University of Technology
Mekelweg 2, 2628 CD Delft, The Netherlands
e-mail: e.t.vandonkelaar@wbmt.tudelft.nl

Abstract

The efficiency of model predictive control can be increased by the on-line choice of a low number of degrees of freedom in the parametrization of the input trajectory. The complexity of the optimization is reduced by searching a solution in a lower dimensional subset of the full input space. This is done by division of the set of constraints in a set of active, inactive and possibly active constraints. A low dimensional subset of the full input space is constructed that complies with the division of the constraints implying that the optimization complexity can be decreased considerably without sacrificing too much in terms of performance.
Keywords: model predictive control, parametrization

1 Introduction

For the control of systems with hard input and state constraints basically one technology is available: model predictive control [5]. The computational requirement for this control strategy is high due to the on-line optimization. This has restricted the use to relatively slow sampled systems with limited dynamic performance specifications such as encountered in the petrochemical and chemical industry. A large gain in effciency is required to make model predictive control applicable to systems that are sampled (much) faster such as industrial robots, aircrafts, cars and steel and aluminium production processes.

In literature several paths are followed to obtain an efficiency increase in model predictive control algorithms without degrading the performance too much. First, the use of new optimization strategies, such as interior point algorithms, are investigated [11]. Secondly, an attempt is made to dispose of the on-line optimization by precalculation of the non-linear control law that is obtained with model predictive control [4] [6]. A third way to improve efficiency in MPC is to decrease the number of degrees of freedom in the optimization problem to obtain a lower order optimization problem. Examples are the reduction of the free variables in the input parametrization by introduction of a control horizon, e.g. used in [1] in a finite horizon MPC and in [7] in an infinite horizon MPC, the use of blocking [2], the use of functions in predictive functional control (PFC, [8]) and the more recent work of [9] where the input trajectory is parameterized in terms of only one variable.

In this article it is shown how the efficiency of model predictive control can be increased by the on-line choice of a low number of degrees of freedom in the parametrization of the input trajectory. The complexity of the optimization is reduced by searching a solution in a low dimensional subset of the full input space.

2 Preliminaries

Let the system be given by the state-space description

$$\begin{aligned} x(t+1) &= Ax(t) + Bu(t), x(0) = x_0 \\ y(t) &= Cx(t) \end{aligned}$$

subject to input and state constraints

$$K_u u(t) \leq k_u, K_x x(t) \leq k_x \text{ for all } t \tag{1}$$

where $K_u \in \mathbb{R}^{n_u \times n_u}, K_u \in \mathbb{R}^{n \times n}$, $x(t) \in \mathbb{R}^n$ and $u(t) \in \mathbb{R}^{n_u}$. The pair (A, B) is controllable and (C, A) is observable. The control cost function is given by $J(u(\cdot)) =$

$$\sum_{t=0}^{P-1} \{x^T(t)Q_1x(t) + u^T(t)Q_2u(t)\} + x^T(P)Q_0x(P) \tag{2}$$

with the weighting matrices $Q_1, Q_2 \geq 0$ and the pair $\{Q_1, A\}$ is detectable. The optimal input trajectory can be found by solving an optimization problem in terms of the vector $U^T = [u^T(0)\ u^T(1) \dots u^T(P-1)]$, which is given by the quadratic programming problem

$$\min_U \{U^T HU + 2x_0^T G_x^T U\} \text{ subject to } KU \leq k \tag{3}$$

[1]Now with the Department of Applied Physics, Delft University of Technology

The variables in these expressions are well described in literature. At each time instant the value x_0 in the quadratic program is updated with the measured value of the state (full information case) or a prediction of the current state (partial information case). In this way at each time instant an updated quadratic program is specified where possibly another set of constraints is active.

3 Selection of degrees of freedom in quadratic programming

The high computational load of quadratic programming solvers for large scale problems is mainly due to the fact that the active set of constraints has to be searched for. Once the set of active constraints is known, the solution of the quadratic programming problem can be computed analytically and fast from the Lagrange necessary conditions. Using the prior knowledge that is available at each time instant it can be possible to make a selection of constraints that are certainly not active, certainly active and constraints that are either active or inactive. By searching only over the uncertain constraints a decrease in computational load can be obtained. This strategy basically involves the following steps.

Main procedure

Step 1. *Divide the set of constraints in a set of inactive, active and possibly active constraints.*

Step 2. *Determine a low dimensional subset of the full input space $\mathbb{R}^{Nn_u}$ that contains all input trajectories that comply with the three sets of constraints.*

Step 3. *Solve a reduced order quadratic programming problem by searching a solution in the selected subset.*

This strategy is further developed in this section.

Step 1: Dividing the set of constraints

The Lagrange multiplier λ indicates whether a constraint is active or not. By definition all elements of the Lagrange multiplier are nonnegative and if the element $\lambda_i = 0$ then the i^{th} constraint is inactive and if $\lambda_i > 0$ the corresponding constraint is active.

An estimate of λ can be found in an approximate solution of the dual quadratic program. The dual of the quadratic program (3) is given by [3]

$$\min_{\lambda}\{\lambda^T P \lambda + d^T \lambda\} \text{ subject to } \lambda \geq 0 \qquad (4)$$

with $P = KH^{-1}K^T$ and $d = KH^{-1}G_x x_0 + k$.

This optimization problem has n_c free variables which in general much more than the number of free parameters. However, the constraints are merely nonnegativity constraints which are easier to handle than general linear constraints.

To obtain an approximate solution of this optimization problem, Hildreths quadratic programming procedure can be used [3]. Let the optimal solution of this quadratic program be given by λ^* with the corresponding solution to the primal problem $U^* = H^{-1}(k - K^T\lambda^*)$. λ^* can be obtained with successive optimization of each element of λ separately. This can be expressed explicitly as $\lambda_i^{k+1} = max(0, w_i^{k+1})$ where

$$w_i^{k+1} = -\frac{1}{p_{ii}}\left(d_i + \sum_{j=1}^{i-1} p_{ij}\lambda_j^{k+1} + \sum_{j=i+1}^{n} p_{ij}\lambda_j^k\right)$$

A full cycle of these calculations is denoted as *Hildreth iteration.* An iteration of Hildreth iterations converges monotonically to the optimal Lagrange multiplier in a finite number of iterations [3].

A distinction can be made between certainly active, certainly inactive and uncertain constraints by inspection of the elements of w rather than λ using the following procedure. The three sets of constraints are denoted with $\mathcal{A}_c, \mathcal{A}_i, \mathcal{A}_u$ respectively.

Procedure 1: Estimation of active constraints.

0. *Check the unconstrained solution. If it is feasible terminate else go to step 1.*

1. *Start with a feasible Lagrange multplier. Take the Lagrange multiplier from the previous time instant λ_{t-1}^* and shift the corresponding input and output constraints one sample back in time.*

2. *Apply n_H Hildreth iterations to obtain an approximation of the Lagrange multiplier.*

3. *Set the set $\mathcal{A}_c$ to the constraints with a corresponding value $w_i > \delta$ and $\mathcal{A}_u$ to those with a value $|w_i| \leq \delta$ where δ is positive real.*

Step 2: Construction of a parametrization

Once the active set of constraints is known, the optimum can be calculated analytically by solving the Lagrangian equations. Generally the active set is not known but let us assume that we are certain that the set of constraints $L_1 U = l_1$ denoted with $\mathcal{A}_c$ is active at the optimal solution and that we are uncertain whether the constraints $L_2 U = l_2$ denoted with $\mathcal{A}_u$ are active or not. Further assume that the other constraints are inactive.

Now, we can define an optimization problem that only tries to find out whether the uncertain constraints are active or not. This can be done by choosing as free parameter θ, the bound l_2. This parameter is strongly related to the slack variable for the corresponding constraint. This results in a set of equations given by

$$\begin{bmatrix} H & L_1^T & L_2^T \\ L_1 & 0 & 0 \\ L_2 & 0 & 0 \end{bmatrix} \begin{bmatrix} U \\ \lambda_1 \\ \lambda_2 \end{bmatrix} = \begin{bmatrix} -G_x x_0 \\ l_1 \\ \theta \end{bmatrix} \qquad (5)$$

All optimal input trajectories that satisfy the assumption about the constraints can be described by

$$U(\theta) = [\phi_x \; \phi_l \; \phi] \begin{bmatrix} x_0 \\ l_1 \\ \theta \end{bmatrix} = \varphi + \phi\theta \qquad (6)$$

where ϕ_k can be calculated from the inverse[1] of the partitioned matrix on the left hand side of (5). Now we can specify a quadratic programming problem in the new parameter θ by substitution of (6) in (3), which is of much lower order than the original problem.

4 Algorithmic properties

The algorithm has two important tuning variables, δ and n_H, that can be used to trade off computational speed and accuracy. A larger value for δ gives a larger set of uncertain constraints and as a consequence a larger computational load for the reduced QP. But the chance becomes larger that the input parametrization contains the optimal trajectory or a very good approximate thereof which improves the accuracy. A larger value for n_H increases the computational load but also increases the chance that the correct division of the constraint set. If n_H is such that the Hildreth iteration has converged or the sign of Lagrange multipliers is correct, no optimization is needed and the optimum can be calculated analytically.
The computational price consists mainly of two parts: computation of the constraint division in Step 1 and solution of the reduced QP in Step 3. Step 1 has complexity $O(P)$ because a finite number of computations is necessary for the correction of each element of the Lagrange mutliplier. And the size of the Lagrange multiplier is an integer times the number of free variables, i.e. input upper and lower bound, input rate constraints and (possibly) output upper and lower bound. Step 2 has a computational price of $O(n_\theta)$ if an interior point algorithm is used. Therefore the overall complexity of the algortithm is of order $O(P)$, which is comparable to interior point algorithms.
The computer storage is larger than primal methods like the active set method but is similar to methods where also the dual problem is needed like primal-dual interior point methods.

5 Model predictive control with on-line input parametrization

The following algorithm applies the ideas of the previous sections.

Algorithm 1. *Let the quasi infinite LQ cost function be given by* (2) *with P such that after this time instant no constraints are active and $Q_0 = Y$ is the solution to the Lyapunov equation*

$$(A - BF)^T Y (A - BF) + F^T Q_2 F + Q_1 = Y$$

where F is the LQ optimal state feedback. Let the input be parametrized as $U(\theta) = \phi\theta + \varphi$ as given in (6) *with constraint sets $\mathcal{A}_u, \mathcal{A}_c$ estimated with Procedure 1.*

The controller of Algortithm 1 is:
1. equivalent to LQ control if no constraint are active,
2. equivalent to constrained LQ control provided that the division of the constraint set is correct.
3. attaining closed-loop stability if and only if the optimization problem is feasible.

The proof of the last property follows a Lyapunov argument.
The constrained infinite horizon model predictive control approach described here is similar to the one presented in [10] but it differs in the applied parametrization of the input. In their approach constrained LQ optimality is obtained at each time instant possibly at the cost of a large computational load. Here optimality at each time instant is sacrificed, i.e. optimality is obtained only under the condition that the division of the constraint set is correct, in return for a gain in computational efficiency. The favourable property of constrained stability is preserved with the presented approach.

6 Simulation example

The simulation example illustrates some properties of the proposed MPC algorithm and the Hildreth iteration used to estimate the active constraints. A typical mechanical system which is highly oscillatory and is nonminimum-phase is given by the transfer function

$$G(z) = \frac{-5.7980z^3 + 19.5128z^2 - 21.6452z + 7.9547}{z^4 - 3.0228z^3 3.8630z^2 - 2.6426z + 0.8084}$$

The input is constrained by $-0.275 \leq u(t) \leq 0.275$. To obtain integral action in the controller the system is augmented with an integrator at the input. Simultions are performed on a PC with a 233 MHz Pentium processor.
If MPC is applied to this system it is necessary for satisfactory performance to take a prediction horizon that is long enough to incorporate at least one full period, i.e. P=100. If a pulse parametrization is applied, the choice of the control horizon M is critical for this system. Decreasing the degrees of freedom easily gives bad performance. Reduction of the control horizon upto M=75 is possible without considerable loss of performance, further reduction gives bad performance due to the slow oscillation. Therefore the computational burden is large for high performance MPC.
In figure 1 the closed loop response is given for the proposed Algorithm 1 together with constrained LQ optimal control.

[1] Provided $[L_c^T \; L_u^T]^T$ has full column rank.

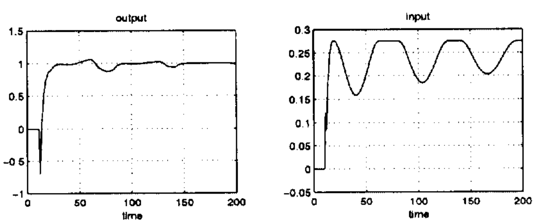

Fig. 1: *Output (left) and input (right) of the closed-loop step response with constrained LQ control with P=100 (solid) and Algorithm 1* $\delta = 0$ *and* $n_H = 2$ *(dashed).*

The two simulations practically conincide, therefore a more precise indication of the trade-off between accuracy and complexity is given in table 1.

Method	tuning	computation time 200 simulation steps (CPU s)	error $\|\|y - y_r\|\|^2$
Alg. 1	$n_H = 1$	22.553	3.1×10^{-3}
Alg. 1	$n_H = 2$	25.626	1.2×10^{-3}
Alg. 1	$n_H = 5$	38.486	8.4×10^{-9}
QP	P=100	367.348	0

Table 1: *Trade-off between performance and complexity for* $\delta = 0$ *and several values for* n_H*. Here* y *is the closed loop response of the output using MPC with free parametrization up to* $P = 100$ *using an active set QP and* y_r *using Algorithm 1.*

If the active set quadratic programming routine is used, the simulation of 200 steps of the model predictive controlled plant takes 367.348 CPU s. Due to the choice for $\delta = 0$ no reduced QP has to be solved which has a favourable influence on the computational load. This tuning is possible without considerable closed loop performance loss because of the high quality of the estimates of the constraint set. Note that this is obtained with only a few Hildreth iterations.

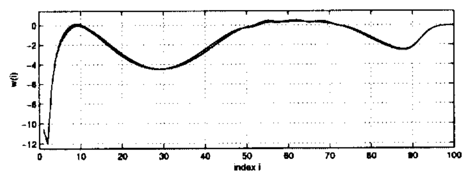

Fig. 2: *Evolution of estimate of* w *for 2, 5, 10 and 50 Hildreth iterations.*

The relatively fast convergence of the Hildreth iterations to an approximate solution of the Lagrange multiplier can also be seen in figure 2 where the evolution of w is depicted for several values for n_H.
Another favourable properties of the algorithm are the relative insensitivity of the algorithm for the number of constraints that are active at the solution. In that case the increase in efficiency compared to the active set method can be as high as a factor 20.

7 Conclusions

The efficiency of model predictive control algorithms can be improved by choosing on-line a parametrization of the input trajectory with a small number of degrees of freedom. An algorithm is suggested that selects every time instant a low dimensional subset of the full input space on the basis of an estimation of the set of active constraints. This is used in an infinite horizon model predictive controller that attains LQ optimal control if no constraints are active, constrained LQ optimal control if the selected subset contains the optimum and attains closed loop stability. The tuning parameters in the algorithm provide a way to trade-off computational complexity and performance. A simulation example shows that this approach can provide a large gain in efficiency compared to standard QP solvers.

References

[1] C.R. Cutler and B.L. Ramaker. Dynamic matrix control - a computer control algorithm. *Proc. Automatic Control Conf.,Paper WP5-B*, 1980.

[2] J.H. Lee, Y. Chikkula, and Z. Yu. Improving computational efficiency of model predictive control algorithm using wavelet transformation. *Int. J. Control*, 61(4):859–883, 1995.

[3] D.G. Luenberger. *Optimization by vector space methods.* John Wiley, New York, 1969.

[4] D.Q. Mayne and W.R. Schroeder. Nonlinear control of constrained linear systems. *Int. J. Control*, 60(5):1035–1043, 1994.

[5] M. Morari and J.H. Lee. Model predictive control: past, present and future. *Proceedings PSE'97-ESCAPE Symposium, Trondheim, Norway*, pages 1–12, 1997.

[6] M. Pottmann and D.E. Seborg. A nonlinear predictive control strategy based on radial basis function models. *Computers Chem. Engng.*, 21(9):965–980, 1997.

[7] J.B. Rawlings and K.R. Muske. The stability of constrained receding horizon control. *IEEE Trans. Autom. Control*, AC-38(10):1512–1516, 1993.

[8] J. Richalet, S.A. El Ata-Doss, C. Arber, H.B. Kuntze, A. Jacubasch, and W. Schill. Predictive functional control application to fast and accurate robots. *Proc. 10th IFAC world congres, Munich, Germany*, 1987.

[9] J.A. Rossiter, M.J. Rice, J. Schuurmans, and B. Kouvaritakis. A computationally efficient constrained predictive control law. *Proceedings 1998 American Contr. Conf., Philadelphia, PA*, 1998.

[10] P.O.M. Scokaert and J.B. Rawlings. Constrained lqr regulation. *IEEE Trans. Autom. Control*, AC-43(8):1163–1169, 1998.

[11] S.J. Wright. Applying new optimization algorithms to model predictive control. *Conf. Chemical Process Control V, Lake Tahoe, CA*, pages 147–155, 1996.