

We are IntechOpen, the world's leading publisher of Open Access books Built by scientists, for scientists

4,100

Open access books available

116,000

International authors and editors

125M

Downloads

Our authors are among the

154

Countries delivered to

TOP 1%

most cited scientists

12.2%

Contributors from top 500 universities



WEB OF SCIENCE™

Selection of our books indexed in the Book Citation Index
in Web of Science™ Core Collection (BKCI)

Interested in publishing with us?
Contact book.department@intechopen.com

Numbers displayed above are based on latest data collected.
For more information visit www.intechopen.com



Improved Nonlinear Model Predictive Control Based on Genetic Algorithm

Wei CHEN, Tao ZHENG, Mei CHEN and Xin LI
Department of automation, Hefei University of Technology, Hefei, China

1. Introduction

Model predictive control (MPC) has made a significant impact on control engineering. It has been applied in almost all of industrial fields such as petrochemical, biotechnical, electrical and mechanical processes. MPC is one of the most applicable control algorithms which refer to a class of control algorithms in which a dynamic process model is used to predict and optimize process performance. Linear model predictive control (LMPC) has been successfully used for years in numerous advanced industrial applications. It is mainly because they can handle multivariable control problems with inequality constraints both on process inputs and outputs.

Because properties of many processes are nonlinear and linear models are often inadequate to describe highly nonlinear processes and moderately nonlinear processes which have large operating regimes, different nonlinear model predictive control (NMPC) approaches have been developed and attracted increasing attention over the past decade [1-5].

On the other hand, since the incorporation of nonlinear dynamic model into the MPC formulation, a non-convex nonlinear optimal control problem (NOCP) with the initial state must be solved at each sampling instant. At the result only the first element of the control policy is usually applied to the process. Then the NOCP is solved again with a new initial value coming from the process. Due the demand of an on-line solution of the NOCP, the computation time is a bottleneck of its application to large-scale complex processes and NMPC has been applied almost only to slow systems. For fast systems where the sampling time is considerably small, the existing NMPC algorithms cannot be used. Therefore, solving such a nonlinear optimization problem efficiently and fast has attracted strong research interest in recent years [6-11].

To solve NOCP, the control sequence will be parameterized, while the state sequence can be handled with two approaches: sequential or simultaneous approach. In the sequential approach, the state vector is handled implicitly with the control vector and initial value vector. Thus the degree of freedom of the NLP problem is only composed of the control parameters. The direct single shooting method is an example of the sequential method. In the simultaneous approach, state trajectories are treated as optimization variable. Equality constraints are added to the NLP and the degree of freedom of the NLP problem is composed of both the control and state parameters. The most well-known simultaneous method is based on collocation on finite elements and multiple shooting.

Both single shooting method and multiple shooting based optimization approaches can then be solved by a nonlinear programming (NLP) solver. The conventional iterative optimization method, such as sequential quadratic programming (SQP) has been applied to NMPC. As a form of the gradient-based optimization method, SQP performs well in local search problems. But it cannot assure that the calculated control values are global optimal because of its relatively weak global search ability. Moreover, the performance of SQP greatly depends on the choice of some initialization values. Improper initial values will lead to local optima or even infeasible solutions.

Genetic Algorithms (GAs) are a stochastic search technique that applies the concept of process of the biological evolution to find an optimal solution in a search space. The conceptual development of the technique is inspired by the ability of natural systems for adaptation. The increasing application of the algorithm has been proved to be efficient in solving complicated nonlinear optimization problems, because of their ability to search efficiently in complicated nonlinear constrained and non-convex optimization problem, which makes them more robust with respect to the complexity of the optimization problem compared to the more conventional optimization techniques.

Compared with SQP, GAs can reduce the dimension of search space efficiently. Indeed, in SQP the state sequence is treated as additional optimization variables; as such, the number of decision variables is the sum of the lengths of both the state sequence and the control sequence. In contrast, in GAs, state equations can be included in the objective function, thus the number of decision variables is only the length of control sequence. Furthermore, the search range of the input variable constraints can be the search space of GA during optimization, which makes it easier to handle the input constraint problem than other descent-based methods.

However, a few applications of GAs to nonlinear MPC [12][13] can partially be explained by the numerical complexity of the GAs, which make the suitable only for processes with slow dynamic. Moreover, the computational burden is much heavier and increases exponentially when the horizon length of NMPC increases. As a result, the implementation of NMPC tends to be difficult and even impossible.

In this paper an improved NMPC algorithm based on GA is proposed to reduce the severe computational burden of conventional GA-based NMPC algorithms. A conventional NMPC algorithm seeks the exact global solution of nonlinear programming, which requires the global solution be implemented online at every sampling time. Unfortunately, finding the global solution of nonlinear programming is in general computationally impossible, not mention under the stringent real-time constraint. We propose to solve a suboptimal descent control sequence which satisfies the control, state and stability constraints in the paper. The solution does not need to minimize the objective function either globally or locally, but only needs to decrease the cost function in an effective manner. The suboptimal method has relatively less computational demands without deteriorating much to the control performance.

The rest of the paper is organized as follows. Section 2 briefly reviews nonlinear model predictive control. Section 3 describes the basics of GAs, followed by a new GA-based computationally efficient NMPC algorithm. Section 4 analyses the stability property of our nonlinear model predictive control scheme for closed-loop systems. Section 5 demonstrates examples of the proposed control approach applied to a coupled-tank system and CSTR. Finally we draw conclusions and give some directions for future research.

2. Nonlinear model predictive control

2.1 System

Consider the following time-invariant, discrete-time system with integer k representing the current discrete time event:

$$x(k+1) = f[x(k), u(k)] \quad (1)$$

In the above, $x(k) \in X \subseteq R^{nx}$ is the system state variables; $u(k) \in U \subseteq R^{mu}$ is the system input variables; the mapping $f: R^{nx} \times R^{mu} \rightarrow R^{nx}$ is twice continuously differentiable and $f(0,0) = 0$.

2.2 Objective function

The objective function in the NMPC is a sum over all stage costs plus an additional final state penalty term [14], and has the form:

$$J(k) = F(x(k+P|k)) + \sum_{j=0}^{P-1} l(x(k+j|k), u(k+j|k)) \quad (2)$$

where $x(k+j|k)$ and $u(k+j|k)$ are predicted values at time k of $x(k+j), u(k+j)$. P is the prediction horizon. In general, $F(x) = x^T \bar{Q}x$ and $l(x, u) = x^T Qx + u^T Ru$. For simplicity, $\bar{Q} > 0$ defines a suitable terminal weighting matrix and $Q \geq 0, R > 0$.

2.3 General form of NMPC

The general form of NMPC law corresponding to (1) and (2) is then defined by the solution at each sampling instant of the following problem:

$$\min_{u(k|k), u(k+1|k), \dots, u(k+P-1|k)} J(k) \quad (3a)$$

$$s.t. x(k+i+1|k) = f(x(k+i|k), u(k+i|k)) \quad (3b)$$

$$x(k+i|k) \in X, u(k+i|k) \in U, i = 0, 1, \dots, P-1 \quad (3c)$$

$$x(k+P|k) \in X_F \quad (3d)$$

where X_F is a terminal stability constraint, and $\mathbf{u}(k) = [u(k|k), \dots, u(k+P-1|k)]$ is the control sequence to be optimized over.

The following assumptions A1 - A4 are made:

A1: $X_F \subset X$, X_F closed, $0 \in X_F$

A2: the local controller $\kappa_F(x) \in U$, $\forall x \in X_F$

A3: $f(x, \kappa_F(x)) \in X_F$, $\forall x \in X_F$

A4: $F(f(x, \kappa_F(x))) - F(x) + l(x, \kappa_F(x)) \leq 0$, $\forall x \in X_F$

Based on the formulation in (3), model predictive control is generally carried out by solving online a finite horizon open-loop optimal control problem, subject to system dynamics and constraints involving states and controls. At the sampling time k , a NMPC algorithm attempts to calculate the control sequence $\mathbf{u}(k)$ by optimizing the performance index (3a)

under constraints (3b) (3c) and terminal stability constraint (3d). The first input $u(k|k)$ is then sent into the plant, and the entire calculation is repeated at the subsequent control interval $k+1$.

3. NMPC algorithm based on genetic algorithm

3.1 Handling constraints

An important characteristic of process control problems is the presence of constraints on input and state variables. Input constraints arise due to actuator limitations such as saturation and rate-of-change restrictions. Such constraints take the form:

$$\mathbf{u}_{\min} \leq \mathbf{u}(k) \leq \mathbf{u}_{\max} \quad (4a)$$

$$\Delta \mathbf{u}_{\min} \leq \Delta \mathbf{u}(k) \leq \Delta \mathbf{u}_{\max} \quad (4b)$$

State constraints usually are associated with operational limitations such as equipment specifications and safety considerations. System state constraints are defined as follows:

$$\mathbf{x}_{\min} \leq \mathbf{x}(k) \leq \mathbf{x}_{\max} \quad (4c)$$

where $\Delta \mathbf{u}(k) = [u(k|k) - u_{k-1}, \dots, u(k+P-1|k) - u(k+P-2|k)]$, $\mathbf{x}(k) = [x(k+1|k), \dots, x(k+P|k)]$.

The constraints (4a) and (4b) can be written as an equivalent inequality:

$$\begin{bmatrix} I \\ -I \\ I \\ -I \end{bmatrix} \mathbf{u}(k) \leq \begin{bmatrix} \mathbf{u}_{\max} \\ -\mathbf{u}_{\min} \\ S\Delta \mathbf{u}_{\max} + c\mathbf{u}_{k-1} \\ -S\Delta \mathbf{u}_{\min} - c\mathbf{u}_{k-1} \end{bmatrix} \quad (5)$$

where

$$S = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & \dots \\ \dots & \dots & \dots & 0 \\ 1 & \dots & \dots & 1 \end{bmatrix}, \quad c = [I, I, \dots, I]^T.$$

3.2 Genetic algorithm

GA is known to have more chances of finding a global optimal solution than descent-based nonlinear programming methods and the operation of the GA used in the paper is explained as follows.

3.2.1 Coding

Select the elements in the control sequence $\mathbf{u}(k)$ as decision variables. Each decision variable is coded in real value and $n_u * P$ decision variables construct the $n_u * P$ -dimensional search space.

3.2.2 Initial population

Generate initial control value in the constraint space described in (5). Calculate the corresponding state value sequence $\mathbf{x}(k)$ from (3b). If the individual (composing control

value and state value) satisfies the state constraints (4c) and terminal constraints (3d), select it into the initial population. Repeat the steps above until PopNum individuals are selected.

3.2.3 Fitness value

Set the fitness value of each individual as $1/(J+1)$.

3.2.4 Genetic operators

Use roulette method to select individuals into the crossover and mutation operator to produce the children. Punish the children which disobey the state constraints (4c) and terminal constraints (3d) with death penalty. Select the best PopNum individuals from the current parent and children as the next generation.

3.2.5 Termination condition

Repeat the above step under certain termination condition is satisfied, such as evolution time or convergence accuracy.

3.3 Improved NMPC algorithm based on GA

In recent years, the genetic algorithms have been successfully applied in a variety of fields where optimization in the presence of complicated objective functions and constraints. The reasons of widely used GAs are its global search ability and independence of initial value. In this paper GAs are adopted in NMPC applications to calculate the control sequence. If the computation time is adequate, GAs can obtain the global optimal solution. However, it needs to solve on line a non-convex optimization problem involving a total number of $n_u * P$ decision variables at each sampling time. To obtain adequate performance, the prediction horizon should be chosen to be reasonably large, which results in a large search space and an exponentially-growing computational demand. Consequently, when a control system requires fast sampling or a large prediction horizon for accurate performance, it becomes computationally infeasible to obtain the optimal control sequence via the conventional GA approach. There is thus a strong need for fast algorithms that reduce the computational demand of GA.

The traditional MPC approach requires the global solution of a nonlinear optimization problem. This is in practice not achievable within finite computing time. An improved NMPC algorithm based on GA does not necessarily depend on a global or even local minimum. The optimizer provides a feasible decedent solution, instead of finding a global or local minimum. The feasible solution decreases the cost function instead of minimizing the cost function. Judicious selection of the termination criterions of GA is the key factor in reducing the computation burden in the design of the suboptimal NMPC algorithm. To this end, the following two strategies at the $(k+1)$ -th step are proposed.

- The control sequence output at the k -th control interval in the genetic algorithm is always selected as one of the initial populations at the $(k+1)$ -th control interval. Furthermore, some of the best individuals at the k -th control interval are also selected into the current initial population. Most of all, the elite-preservation strategy is adopted. Figure 1 shows one of the choices of the initial population per iteration. This strategy guarantees the quality of current population and the stability of the NMPC algorithm.

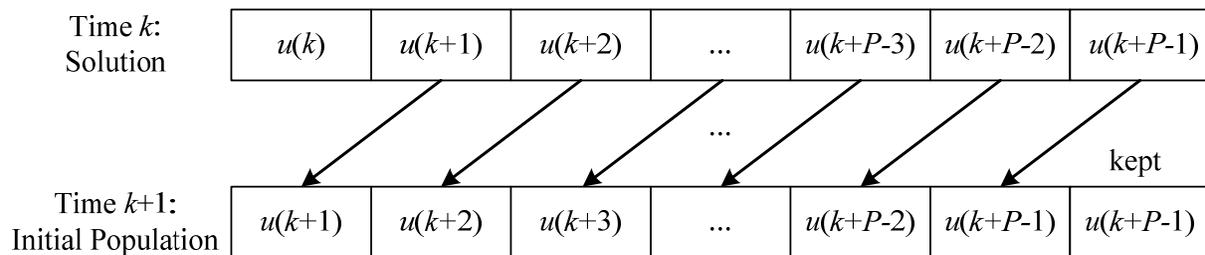


Fig. 1. the choice of initial population per iteration

- Stopping criterions of GA are the key factor of decreasing the computation burden. GA is used to compute the control sequence. The objective value $J(k+1)$ at the $(k+1)$ -th control interval is computed and compared with $J(k)$ that stored at the k -th control interval. If $J(k+1)$ is smaller than $J(k)$, that at the k -th control interval, then the control sequence $\mathbf{u}(k+1)$ is retained as a good feasible solution, and its first element $u(k+1 | k+1)$ is sent to the plant. Otherwise, if there does not exist a feasible value for $\mathbf{u}(k+1)$ to yield $J(k+1) < J(k)$, then the best $\mathbf{u}(k+1)$ is chosen to decrease the objective function the most.

The traditional MPC approach requires the global solution of a nonlinear optimization problem. This is in practice not achievable within finite computing time. An improved NMPC algorithm based on GA does not necessarily depend on a global or even local minimum. The optimizer provides a feasible decedent solution, instead of finding a global or local minimum. The feasible solution decreases the cost function instead of minimizing the cost function. Judicious selection of the termination criterions of GA is the key factor in reducing the computation burden in the design of the suboptimal NMPC algorithm. To this end, the following two strategies at the $(k+1)$ -th step are proposed.

With the above two strategies, the computational complexity of the control calculation is substantially reduced. Summarizing, our proposed improved NMPC algorithm performs the following iterative steps:

Step 1. [Initialization]:

choose parameters P, X_F, Q, R, Q' and model $x(k+1) = f(x(k), u(k))$; initialize the state and control variables at $k = 0$; compute and store $J(0)$.

Step 2. [modified Iteration]:

- at the k -th control interval, determine the control sequence $\mathbf{u}(k)$ using GA satisfies constraints (3b) (3c), terminal stability constraint (3d) and $J(k) < J(k-1)$. The first input $u(k | k)$ is then sent into the plant.
- store $J(k)$ and set $k = k+1$;
- if there does not exist a feasible value for $\mathbf{u}(k)$ to yield $J(k) < J(k-1)$, then the best $\mathbf{u}(k)$ is chosen to decrease the objective function the most.

Step 3. [Termination]

The entire calculation is repeated at subsequent control interval $k+1$ and goes to Step 2.

Though the proposed method does not seek a globally or locally optimal solution within each iteration step, it may cause little performance degradation to the original GA due to its iterative nature, which is known to be capable of improving suboptimal solution step by step until reaching near-optimal performance at the final stage. Besides its near-optimal performance, the proposed algorithm possesses salient feature; it guarantees overall system stability and, most of all, leads to considerable reduction in the online computation burden. Finding a control sequence that satisfies a set of constraints is significantly easier than solving a global optimization problem. Here it is possible to obtain the suboptimal control

sequence via GA for practical systems with very demanding computation load, that is, systems with a small sampling time or a large prediction horizon.

4. Stability of nonlinear model predictive control system

The closed-loop system controlled by the improved NMPC based on GA is proved to be stable.

Theorem 1: For a system expressed in (1) and satisfying the assumption A1-A4, the closed-loop system is stable under the improved NMPC framework.

Proof: Suppose there are an admissible control sequence $\mathbf{u}(k)$ and a state sequence $\mathbf{x}(k)$ that satisfy the input, state and terminal stability constraints at the sampling time k .

At the sampling time k , the performance index, which is related to $\mathbf{u}(k)$ and $\mathbf{x}(k)$, is described as

$$J(k) = J^*(k; \mathbf{u}(k), \mathbf{x}(k)) \quad (6)$$

In the closed-loop system controlled by the improved NMPC, define the feasible input and state sequences for the successive state are $x^+ = f(x, u(k|k))$.

$$\begin{aligned} \mathbf{u}^+(k+1) &= [u^T(k+1|k), \dots, u^T(k+P-1|k), K_F(x(k+P|k))]^T \\ \mathbf{x}^+(k+1) &= [x^T(k+1|k), \dots, x^T(k+P|k), f(x(k+P|k), K_F(x(k+P|k)))]^T \end{aligned} \quad (7)$$

The resulting objective function of $\mathbf{u}(k+1)$ and $\mathbf{x}(k+1)$ at the $(k+1)$ -th sampling time is

$$J^+(k+1) = J(k+1; \mathbf{u}(k+1), \mathbf{x}(k+1)) \quad (8)$$

If the optimal solution is found, it follows that

$$\begin{aligned} J^*(k+1) - J^*(k) &\leq J^+(k+1) - J^*(k) \\ &= -l(x, K_P(x)) + F(f(x, K_F(x))) - F(x) + l(x, K_F(x)) \end{aligned} \quad (9)$$

From A4, the following inequality holds

$$J^*(k+1) - J^*(k) \leq -l(x, K_P(x)) \quad (10)$$

Or using the improved NMPC algorithm, it follows that

$$J^*(k+1) - J^*(k) \leq J^+(k+1) - J^*(k) \leq 0 \quad (11)$$

Thus the sequence $J^*(k+i)$ over P time indices decreases. As such and given the fact that the cost function $l(x, u)$ is lower-bounded by zero, it is evident that $J^*(k+i)$ converges. Taking the sum, we obtain

$$J^*(k+P) - J^*(k) \leq \sum_{i=1}^P [-l(x(k+i), u(k+i))] \quad (12)$$

Also, because the sequence $J^*(k+i)$ is decreasing, then as $N \rightarrow \infty$, we have $l(x(k+i), u(k+i)) \rightarrow 0$ and $x \rightarrow 0$. Hence, the closed-loop system is stable.

5. Simulation and experiment results

5.1 Simulation results to a continuous stirred tank reactor plant

5.1.1 Model of continuous stirred tank reactor plant

Consider the highly nonlinear model of a chemical plant (continuous stirred tank reactor-CSTR). Assuming a constant liquid volume, the CSTR for an exothermic, irreversible reaction, $A \rightarrow B$, is described by

$$\begin{aligned}\dot{C}_A &= \frac{q}{V}(C_{Af} - C_A) - k_0 e^{-E/(RT)} C_A \\ \dot{T} &= \frac{q}{V}(T_f - T) + \frac{-\Delta H}{\rho C_p} k_0 e^{-E/(RT)} C_A + \frac{UA}{V\rho C_p}(T_c - T)\end{aligned}\quad (13)$$

where C_A is the concentration of A in the reactor, T is the reactor temperature and T_c is the temperature of the coolant stream. The parameters are listed in the Table 1.

| Variables | Meaning | Value | Unit |
|------------|---|---------------------|-------------------|
| q | the inlet flow | 100 | l/min |
| V | the reactor liquid volume | 100 | l |
| C_{Af} | the concentration of inlet flow | 1 | mol/l |
| k_0 | reaction frequency factor | $7.2 \cdot 10^{10}$ | min ⁻¹ |
| E/R | | 8750 | K |
| E | activation energy | | |
| R | gas constant | $8.3196 \cdot 10^3$ | J/(mol K) |
| T_f | the temperature of inlet flow | 350 | K |
| ΔH | the heat of reaction | $-5 \cdot 10^4$ | J/mol |
| ρ | the density | 1000 | g/l |
| C_p | the specific heat capacity of the fluid | 0.239 | J/(g K) |
| UA | | $5 \cdot 10^4$ | J/(min K) |
| U | the overall heat transfer coefficient | | |
| A | the heat transfer area | | |

Table 1. List of the model parameters

5.1.2 Simulation results

The paper present CSTR simulated examples to confirm the main ideas of the paper. The nominal conditions, $C_A = 0.5 \text{ mol/l}$, $T = 350 \text{ K}$, $T_c = 300 \text{ K}$, correspond to an unstable operating point. The manipulated input and controlled output are the coolant temperature (T_c) and reactor temperature (T). And the following state and input constraints must be enforced:

$$\begin{aligned}X &= \left\{ \begin{bmatrix} C_A \\ T \end{bmatrix} \in \mathbb{R}^2 \mid 0 \leq C_A \leq 1, 280 \leq T \leq 370 \right\} \\ U &= \{ T_c \in \mathbb{R} \mid 280 \leq T_c \leq 370 \}\end{aligned}\quad (14)$$

The simulation platform is MATLAB and simulation time is 120 sampling time. The sampling time is $T_s = 0.05 \text{ s}$, mutation probability is $P_c = 0.1$, population size is 100, maximum generation is 100, and the fitness value is $1/(J+1)$.

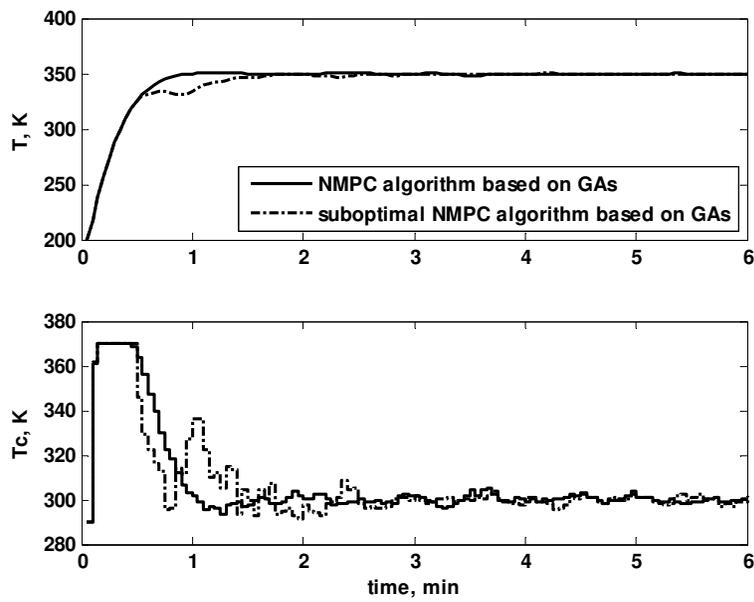


Fig. 2. comparative simulation between the conventional NMPC algorithm and the suboptimal NMPC algorithm

| method | settling time, min | percent overshoot, % |
|---------------------------------------|--------------------|----------------------|
| NMPC algorithm based on GA | 0.75 | 1 |
| Suboptimal NMPC algorithm based on GA | 2.5 | 0 |

Table 2. Performance comparisons of simulation results

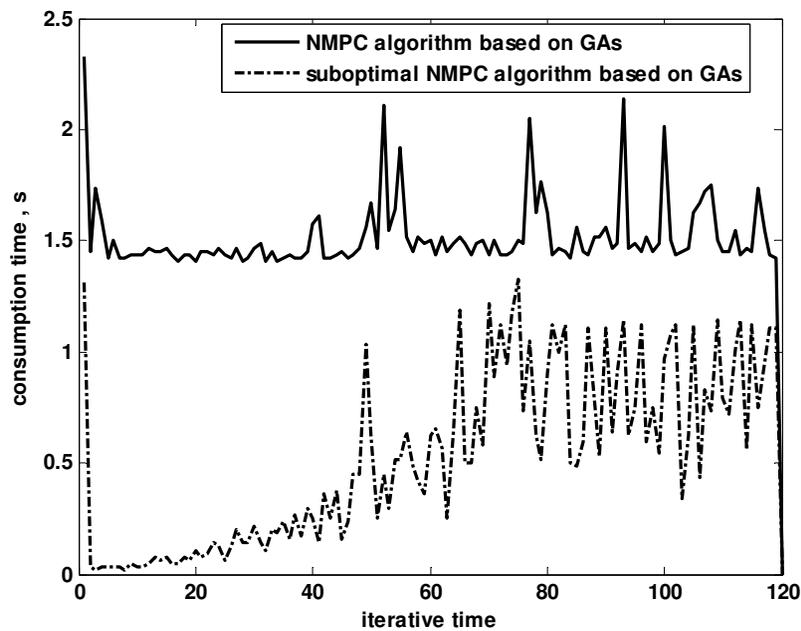


Fig. 3. The time consumptions of the two methods

For comparison, the same simulation setup is used to test both the conventional NMPC algorithm based GA and the suboptimal NMPC algorithm. The resulting control values are depicted in Fig 2. Table 2 compares the performance of the two algorithms using the metrics settling time and percent of overshoot. The conventional NMPC algorithm has a faster transient phase and a smaller percentage of overshoots.

When the population sizes or the maximum generation is relatively large, the time consumption of the two methods is compared in Fig 3.

From Fig 2 and Table 2, it is apparent that the control performance of the two methods is almost same. But from Fig 3, it is evident that the suboptimal NMPC algorithm based on GA has a considerably reduced demand on computational complexity.

5.2 Simulation results to a coupled-tank system

5.2.1 Model of coupled-tank system

The apparatus ^[15], see Fig.4, consists of two tanks T_1 and T_2 , a reservoir, a baffle valve V_1 and an outlet valve V_2 . T_1 has an inlet commanded through a variable pump based on *PMW* and T_2 has an outlet that can be adjusted through a manually controlled valve only. The outlets communicate to a reservoir from which the pumps extract the water to deliver it to the tank. The two tanks are connected through the baffle valve, which again can only be adjusted manually.

The objective of the control problem is to adjust the inlet flow so as to maintain the water level of the second tank close to a desired setpoint. The water levels h_1 and h_2 , which are translated through the pressure transducer into a *DC* voltage ranging from 0V to 5V, are sent to *PC* port via *A/D* transition. The tank pump control, which is computed by the controller in *PC* with the information of the water level h_1 and h_2 , is a current level in the range 4mA to 20mA, where these correspond to the pump not operating at all, and full power respectively.

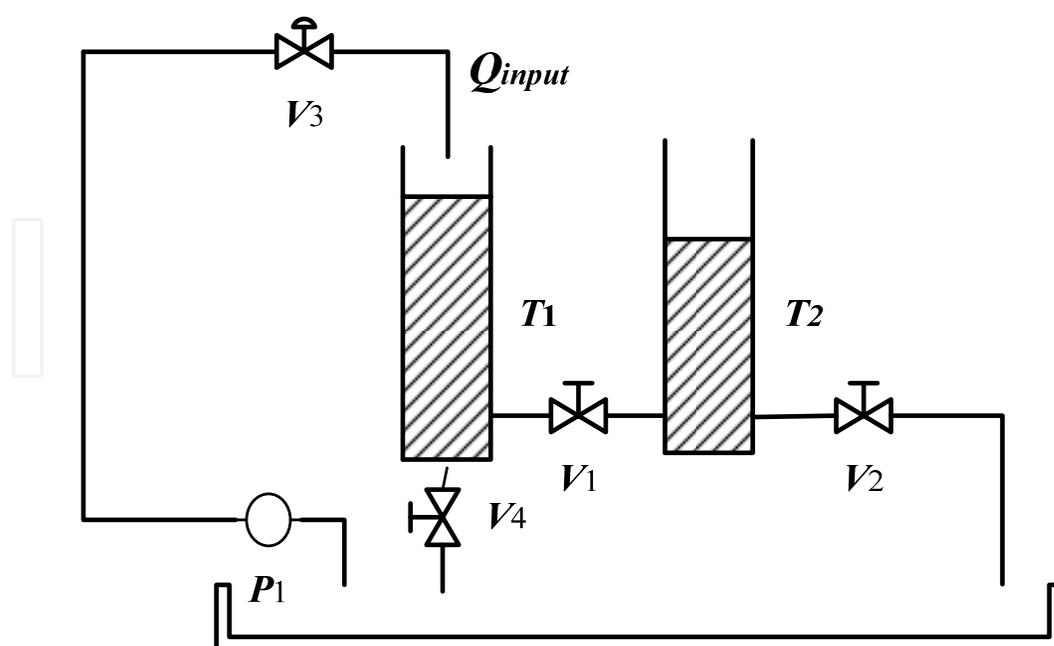


Fig. 4. Coupled-tank apparatus

The dynamics of the system are modeled by the state-space model equations:

$$\begin{aligned} A \frac{dh_1}{dt} &= Q_{input} - Q_{12} \\ A \frac{dh_2}{dt} &= Q_{12} - Q_{20} \end{aligned} \quad (15)$$

where the flows obey Bernoulli's equation^[16], i.e.

$$Q_{12} = \mu_1 S \operatorname{sgn}(h_1 - h_2) (2g|h_1 - h_2|)^{1/2} \quad (16)$$

$$Q_{20} = \mu_2 S (2gh_2)^{1/2} \quad (17)$$

and $\operatorname{sgn}(z) = \begin{cases} 1, z \geq 0 \\ -1, z < 0 \end{cases}$ is the symbol function of parameter z .

The output equation for the system is

$$y = h_2 \quad (18)$$

The cross-section areas, i.e. A and S , are determined from the diameter of the tanks and pipes. The flow coefficients, μ_1 and μ_2 , have experimentally (from steady-state measurements) been determined. Table 3 is the meanings and values of all the parameters in Eqn.15

| Signal | Physics Meaning | Value |
|---------|----------------------------|------------------------------------|
| A | Cross-section area of tank | $6.3585 \times 10^{-3} \text{m}^2$ |
| S | Cross-section area of pipe | $6.3585 \times 10^{-5} \text{m}^2$ |
| g | acceleration of gravity | 9.806m/s^2 |
| μ_1 | flow coefficient 1 | 0.3343 |
| μ_2 | flow coefficient 2 | 0.2751 |

Table 3. meanings and value of all the parameters

Several constraints have to be considered. Limited pump capacity implies that values of Q_{input} range from 0 to $50 \text{cm}^3/\text{s}$. The limits for the two tank levels, h_1 and h_2 , are from 0 to 50cm .

5.2.2 Simulation results

The goal of the couple-tank system is to control the level of Tank 2 to setpoint. The initial levels of the two tanks, h_1 , h_2 , are 0cm . The objectives and limits of the tank system: Input constraint is $0 \leq u \leq 100\%$; State objectives are $0 \leq h_1, h_2 \leq 0.5 \text{m}$, and the setpoint of Tank 2 is 0.1m .

The simulation platform is MATLAB and simulation time is 80 sample time. In NMPC, select prediction horizon $P=10$, weighting parameters $\bar{Q} = Q = 8$, $R=1$, sample time $T_s=5 \text{s}$, mutation probability $P_c=0.1$, population size is 200, maximum generation is 100, the fitness value is $1/(J+1)$.

For the purpose of comparison the same simulation is carried out with the conventional NMPC algorithm based GA and the fast NMPC algorithm. The result is shown in Fig 5. The performance indexes of the two algorithms are shown in Table 4.

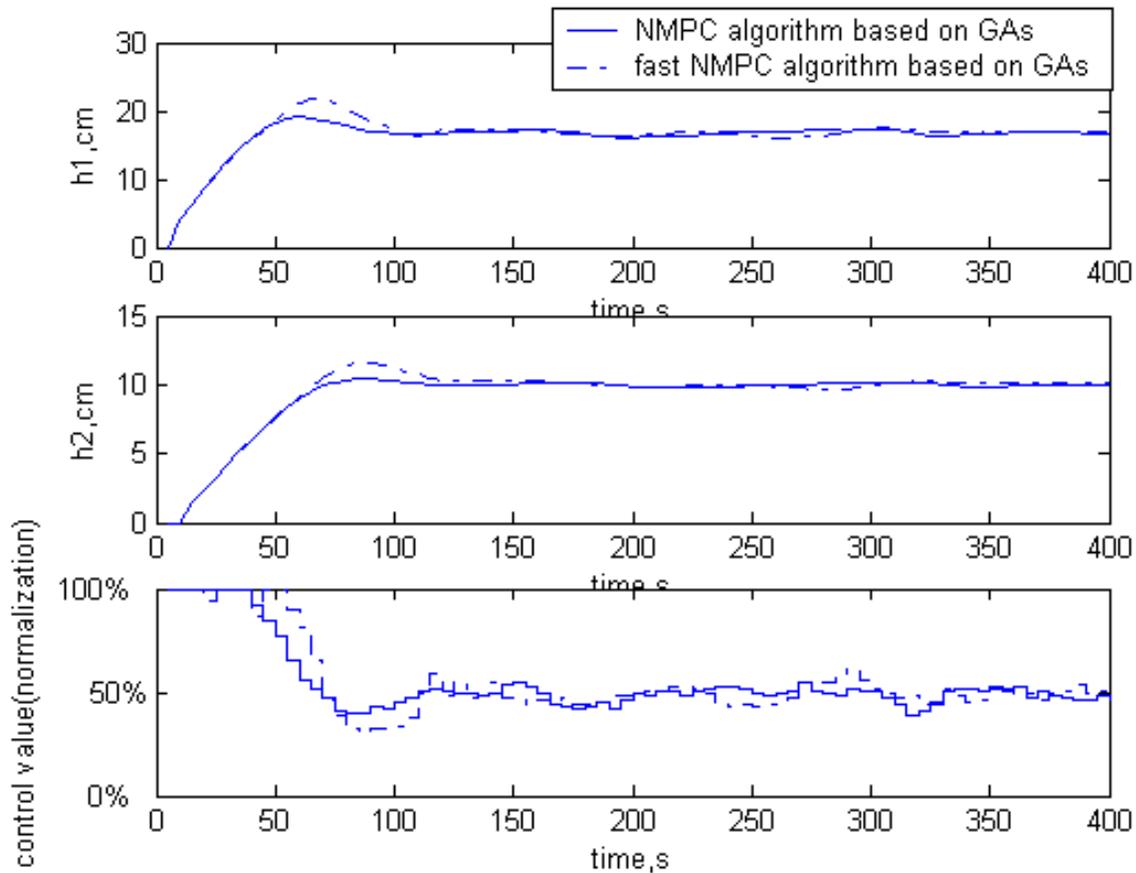


Fig. 5. Compared simulation results based on conventional NMPC and fast NMPC algorithm

| method | Settling time, s | percent overshoot, % |
|---------------------------------|------------------|----------------------|
| NMPC algorithm based on GA | 70 | 3 |
| Fast NMPC algorithm based on GA | 100 | 7 |

Table 4. Performance index of simulation results

When the population sizes or the maximum generation is relatively larger, the time consumptions of the two method is shown in Fig 6.

From Fig 5 and Table 4, it is apparent that the control performance of the two methods is almost same. But from Fig 6, the computation demand reduces significant when the fast NMPC algorithm based on GA is brought into the system.

5.2.3 Experiment results

The objectives and limits of the system: Input constraint is $0 \leq u \leq 100\%$; State objectives are $0 \leq h_1, h_2 \leq 0.5m$, and the setpoint of Tank 2 is 0.1m. Select prediction horizon $P=10$, weighting parameters $\bar{Q} = Q = 8$, $R=1$, sample time $T_s=5s$, mutation probability $P_c=0.1$, population size is 200, maximum generation is 100.

The tank apparatus is controlled with the NMPC algorithm based on conventional GA, the experimental curve is shown in Fig 7 and performance index is shown in Table 5.

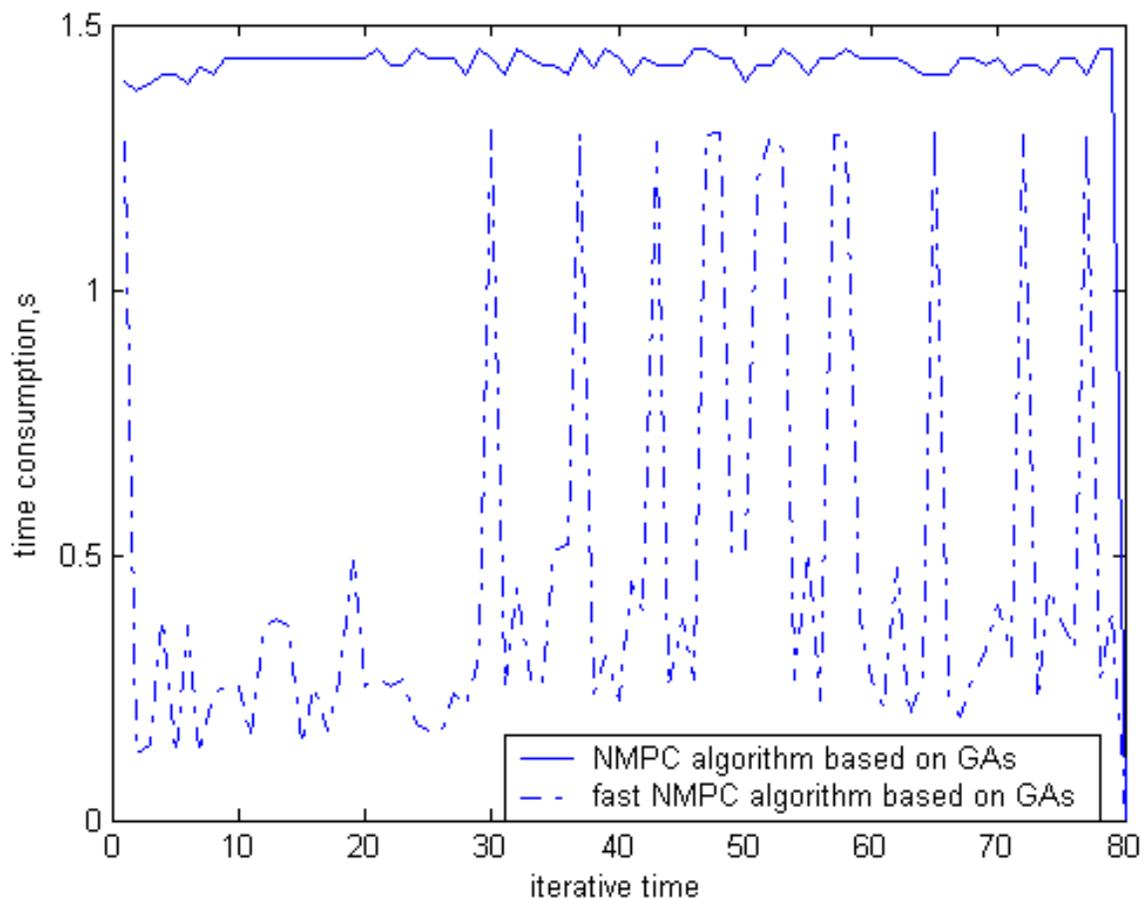


Fig. 6. Time consumptions for two methods

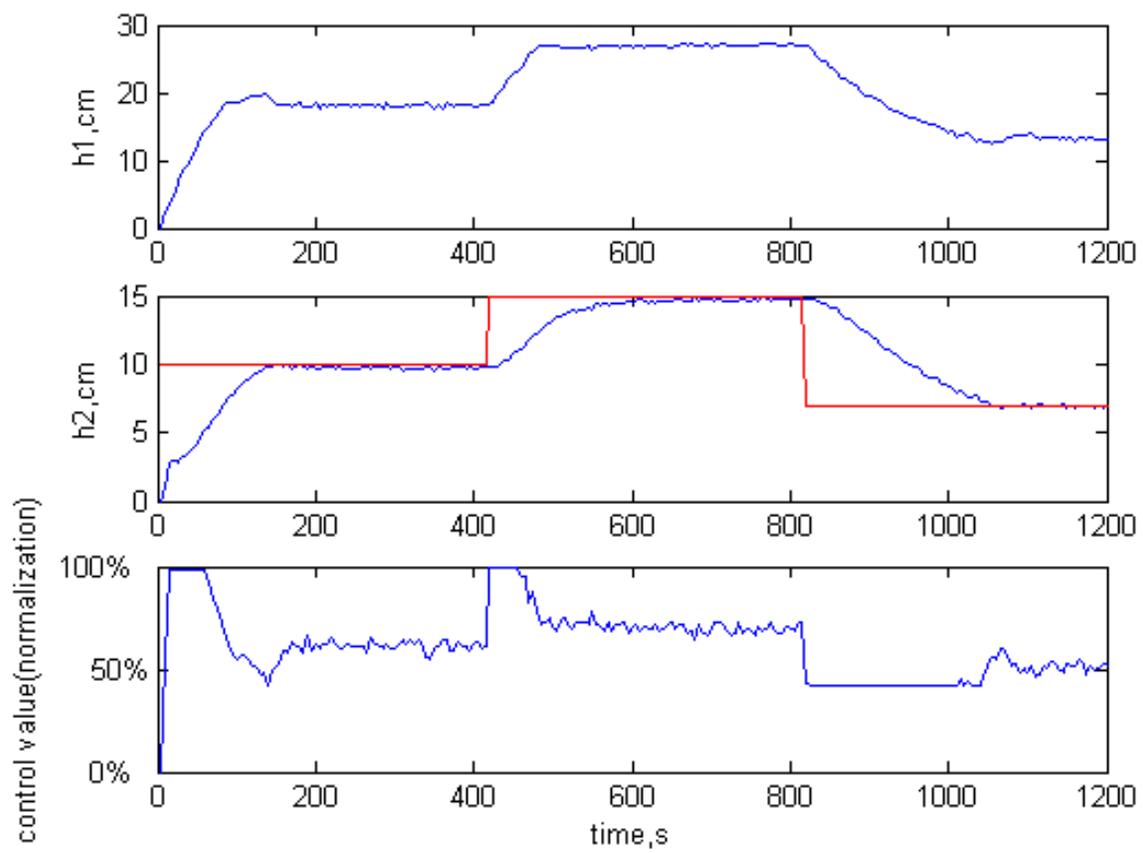


Fig. 7. Experimental curve of NMPC based on GA

| Time, s | 0-400 | 401-800 | 801-1200 |
|-------------------|-------|---------|----------|
| Setpoint, m | 0.1 | 0.15 | 0.07 |
| Settling time, s | 159 | 190 | 210 |
| Percent overshoot | None | None | None |

Table 5. Performance index of experimental result with conventional NMPC

The same experiment is carried out with fast NMPC algorithm based on GA. The result is shown in Fig 8 and performance index is shown in Table 5.

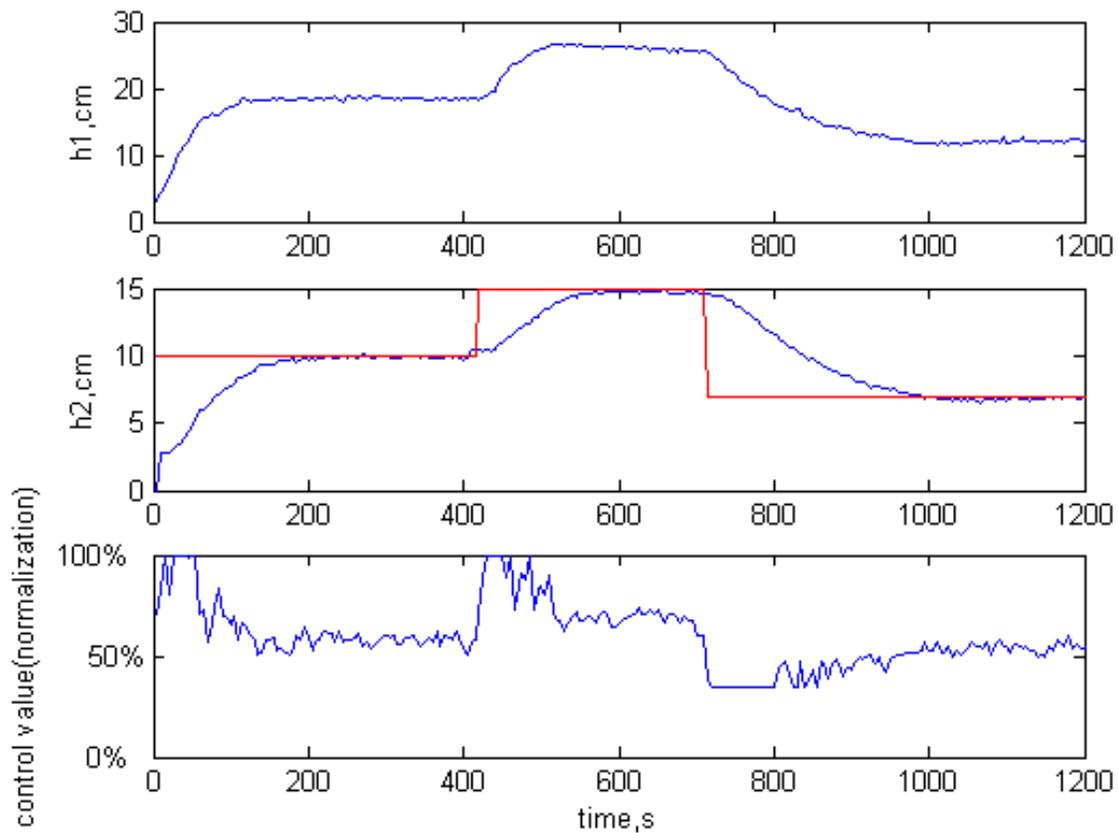


Fig. 8. Experimental curve of fast NMPC based on GA

| Time, s | 0-400 | 401-800 | 801-1200 |
|-------------------|-------|---------|----------|
| Setpoint, m | 0.1 | 0.15 | 0.07 |
| Settling time, s | 190 | 190 | 260 |
| Percent overshoot | None | None | None |

Table 6. Performance index of experimental result with fast NMPC

6. Conclusions

In this paper an improved NMPC algorithm based on GA has been proposed. The aim is to reduce the computational burden without much deterioration to the control performance. Compared with traditional NMPC controller, our approach has much lower computational burden, which makes it practical to operate in systems with a small sampling time or a large prediction horizon.

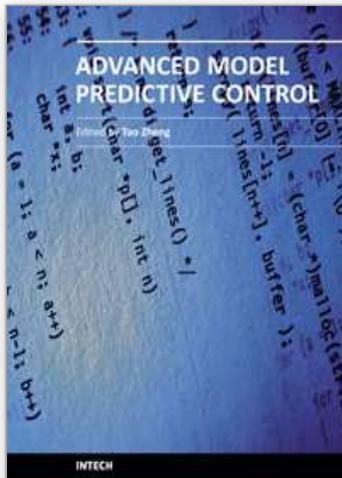
The proposed approach has been tested in CSTR and a real-time tank system. Both computer simulations and experimental testing confirm that the suboptimal NMPC based on GA resulted in a controller with less computation time.

7. Acknowledgment

This work is supported by National Natural Science Foundation of China (Youth Foundation, No. 61004082) and Special Foundation for Ph. D. of Hefei University of Technology (No. 2010HGBZ0616).

8. References

- Michael A.Henson, Nonlinear model predictive control: current status and future directions [J], *Computers and Chemical Engineering* 23(1998):187-202
- D.Q.Mayne, J.B.Rawlings, C.V.Rao and P.O.M.Scokaert, Constrained model predictive control: Stability and optimality [J], *Automatica* 36(2000): 789-814
- S.Joe Qin and Thomas A.Badgwell, a survey of industrial model predictive control technology [J], *Control Engineering Practice* 11(2003): 733-764
- Mark Cannon, Efficient nonlinear model predictive control algorithms [J], *Annual Reviews in Control* 28(2004): 229-237
- Basil Kouvaritakis and Mark Cannon, *Nonlinear predictive control: Theory and practice* [M], the institution of electrical engineers, 2001
- Frode Martinse, Lorenz T.Biegler and Bjarne A.Foss, A new optimization algorithm with application to nonlinear MPC[J], *Journal of Process Control*, 2004:853-865
- Moritz Diehl, Rolf Findeisen, Frank Allgower, Hans Georg Bock and Johannes Schloder, stability of nonlinear model predictive control in the presence of errors due to numerical online optimization, *Proceedings of the 42nd IEEE Conference Decision and Control*, December 2003: 1419-1424
- Rolf Findeisen, Moritz Diehl, Ilknur Disli-Uslu and Stefan Schwarzkopf, Computation and Performance Assessment of Nonlinear Model Predictive Control[J]. *Proc. Of the 41st IEEE Conference on Decision and Control* December 2002:4613-4618
- Moritz Diehl, Rolf Findeisen, Stefan Schwarzkopf, An efficient algorithm for nonlinear predictive control of large-scale systems, *Automatic Technology*, 2002:557-567
- L.T.Biegler, Advaneces in nonlinear programming concepts for process control, *Proc. Cont. Vol.8, Nos.5-6*, pp.301-311, 1998
- Alex Zheng and Frank Allgower, towards a practical nonlinear predictive control algorithm with guaranteed stability for large-scale systems, *Proceedings of the American Control Conference*, 1998:2534-2538
- Jianjun Yang, Min Liu, Cheng Wu. Genetic Algorithm Based on Nonlinear Model Predictive Control Method [J]. *Control and Decision*. 2003,2(18):141-144.
- Fuzhen Xue, Yuyu Tang, Jie Bai. An Algorithm of Nonlinear Model Predictive Control Based on BP Network [J]. *Journal of University of Science and Technology of China*. 2004,5(34): 593-598.
- H.Michalska and D.Q.Mayne, robust receding horizon control of constrained nonlinear systems, *IEEE Transactions on Automatic Control*, Vol.38, No.11,1993:1623-1633
- Wei Chen, Gang Wu. Nonlinear Modeling of Two-Tank System and Its Nonlinear Model Predictive Control [A]. *Proceedings of the 24th Chinese Control Conference*. 2005,7:396-301
- E.John Finnemore and Joseph B.Franzini, *Fluid Mechanics with Engineering Applications(Tenth Edition)*[M], March 2003
- N.K.Poulsen, B.Kouvaritakis and M.Cannon, Nonlinear constrained predictive control applied to a coupled-tank apparatus[J], *IEE Proc.-Control Theory Appl.* , Vol.148 , No 1 , January 2001:17-24
- Rolf Findeisen, Moritz Diehl, Ilknur Disli-Uslu and Stefan Schwarzkopf, Computation and Performance Assessment of Nonlinear Model Predictive Control[J]. *Proc. Of the 41st IEEE Conference on Decision and Control* December 2002:4613-4618



Advanced Model Predictive Control

Edited by Dr. Tao ZHENG

ISBN 978-953-307-298-2

Hard cover, 418 pages

Publisher InTech

Published online 24, June, 2011

Published in print edition June, 2011

Model Predictive Control (MPC) refers to a class of control algorithms in which a dynamic process model is used to predict and optimize process performance. From lower request of modeling accuracy and robustness to complicated process plants, MPC has been widely accepted in many practical fields. As the guide for researchers and engineers all over the world concerned with the latest developments of MPC, the purpose of "Advanced Model Predictive Control" is to show the readers the recent achievements in this area. The first part of this exciting book will help you comprehend the frontiers in theoretical research of MPC, such as Fast MPC, Nonlinear MPC, Distributed MPC, Multi-Dimensional MPC and Fuzzy-Neural MPC. In the second part, several excellent applications of MPC in modern industry are proposed and efficient commercial software for MPC is introduced. Because of its special industrial origin, we believe that MPC will remain energetic in the future.

How to reference

In order to correctly reference this scholarly work, feel free to copy and paste the following:

Wei Chen, Tao Zheng, Mei Chen and Xin Li (2011). Improved Nonlinear Model Predictive Control Based on Genetic Algorithm, Advanced Model Predictive Control, Dr. Tao ZHENG (Ed.), ISBN: 978-953-307-298-2, InTech, Available from: <http://www.intechopen.com/books/advanced-model-predictive-control/improved-nonlinear-model-predictive-control-based-on-genetic-algorithm>

INTECH
open science | open minds

InTech Europe

University Campus STeP Ri
Slavka Krautzeka 83/A
51000 Rijeka, Croatia
Phone: +385 (51) 770 447
Fax: +385 (51) 686 166
www.intechopen.com

InTech China

Unit 405, Office Block, Hotel Equatorial Shanghai
No.65, Yan An Road (West), Shanghai, 200040, China
中国上海市延安西路65号上海国际贵都大饭店办公楼405单元
Phone: +86-21-62489820
Fax: +86-21-62489821

© 2011 The Author(s). Licensee IntechOpen. This chapter is distributed under the terms of the [Creative Commons Attribution-NonCommercial-ShareAlike-3.0 License](#), which permits use, distribution and reproduction for non-commercial purposes, provided the original is properly cited and derivative works building on this content are distributed under the same license.

IntechOpen

IntechOpen