

Internal-growth Strategies for Dynamic Process Optimization with Differential-algebraic Equations^{*}

Zhiqiang Wang^{*,**} Zhijiang Shao^{*} Xueyi Fang^{***} Jun Zhao^{*}
Zuhua Xu^{*}

^{*} Department of Control Science and Engineering, Zhejiang University,
Hangzhou 310027, China (e-mail: zjshao@ipc.zju.edu.cn)

^{**} Institute of Applied Mathematics, Hebei Academy of Sciences,
Shijiazhuang 050081, China (e-mail: wangzhiqiang@heb-math.org)

^{***} School of Electronics and Information, Northwestern Polytechnical
University, Xian 710072, China (e-mail: xyfang@nwpu.edu.cn)

Abstract: Simultaneous methods based on numerical differentiation for process dynamic optimization are discussed, and succinct variants of the full discretized models, which are collocated by Lobatto methods, are proposed. With the same original dynamic problem, its discretized models with different number of finite elements have good structural similarity, and their optimal results can also fit well with each other. A novel solving idea is proposed that the low-density discretized solution is interpolated as the starting point for solving high-density discretized model, then the warm-start technology for the interior point method and the initial value setting method of barrier parameter are integrated to develop a fast solving strategy for dynamic optimization, named internal-growth approach. Finally, several test models, including the crystallization process optimization problem, are solved by the proposed approach, and the excellent solving efficiency is illustrated.

Keywords: dynamic optimization, differential-algebraic equation, orthogonal collocation method.

1. INTRODUCTION

Dynamic is a general feature for chemical process systems, covering from basic chemical reaction units to advanced scheduling systems. According to the increasingly competitive market economic environment as well as the rapid development of computer technology, research and application of dynamic optimization attract more and more attention from the modern process industries. Currently, the dynamic optimization has been used to solve many practical problems of process industries, such as optimal control problem, nonlinear predictive control, dynamic real-time optimization, and parameter estimation of process systems (Kameswaran and Biegler, 2006).

Dynamic process systems mainly involve continuous physical variation, and are described commonly as differential equation models. A generalized differential system is discussed in this study, called differential-algebraic equations (DAE) system, also known as implicit differential equation system (Kunkel and Mehrmann, 2006). Therefore, the dynamic optimization problems which based on DAEs are called DAE optimization problems. Since most of DAE optimization problems can not be solved directly, a compared popular approach is converting DAE

optimization problems to nonlinear programming (NLP) problems, which can be handled by NLP solvers. The methods that apply NLP solvers can be separated into sequential and simultaneous strategies (Biegler et al., 2002; Biegler, 2010). The sequential methods only discretize the control variables, and use nested two-layer framework to obtain the optimal control curves. In the nested framework, the inner structure uses an initial value method to simulate the DAEs, and the outer NLP solver adjusts the iterative direction continuously by the inner results until any terminal condition is satisfied. Sequential methods are relatively easy to construct, but difficult to obtain accurate gradient information, so that lead to poor convergence of iteration and low solving efficiency (Biegler, 2007). The simultaneous strategies discretize all continuous variables and combine simulation and optimization into one layer to calculate, thus they can achieve high efficiency. Therefore, the simultaneous approach is current research hot-spot.

Although the simultaneous approach can dramatically improve solving efficiency, it is still expected to achieve higher accuracy and speed in the practical application. For a special discretization method, there are two ways to improve the solution accuracy, i.e. adopting the variable collocation elements and increasing discretization segments. Despite variable element methods have broad prospects for simultaneous strategy (Biegler, 2007, 2010), there still lacks some efficient criterion to adjust the length of elements during iteration process. In addition, unequal intervals can

^{*} Resrach supported by the National Basic Research Program of China (2012CB720503), the National Natural Science Foundation of China (61273146, 61374167), and the Science and Technology Project of Hebei Academy of Sciences (14601).

increase the condition value of discretized model, thence, it means that the solving process is more expensive. This study adopts conservative strategy yet with equal length elements, but uses several techniques to improve solution accuracy while ensuring the efficiency of results.

2. SIMULTANEOUS SOLVING FOR DAE OPTIMIZATION MODELS

2.1 Standard description of DAE optimization problem

Consider DAE optimization problems in the following form:

$$\min_{x(t), u(t), p} \varphi(x(t_f)) \quad (1a)$$

$$\text{s.t. } \dot{x}(t) = f(x(t), y(t), u(t), p), \quad x(t_0) = x_0, \quad (1b)$$

$$g(x(t), y(t), u(t), p) = 0, \quad (1c)$$

$$h(x(t_f)) = 0, \quad (1d)$$

$$x^L \leq x(t) \leq x^U, \quad y^L \leq y(t) \leq y^U, \quad (1e)$$

$$u^L \leq u(t) \leq u^U, \quad p^L \leq p \leq p^U, \quad (1f)$$

where, differential variables (state variables) $x(t) \in \mathbb{R}^{n_x}$, algebraic variables $y(t) \in \mathbb{R}^{n_y}$, and control variables $u(t) \in \mathbb{R}^{n_u}$ are functions of time $t \in [t_0, t_f]$, as well as $p \in \mathbb{R}^{n_p}$ is time-independent parameters. Here, n_x, n_y, n_u, n_p denote the dimension of variables respectively. Obviously, the equality constraints (1b)-(1c) are DAEs. Without loss of generality, it is assumed that DAE constraints are index 1 (Biegler, 2007).

2.2 Discretization of DAE optimization models

Discretization is the key step in the solving process of simultaneous strategy, since an incorrect discretized model must get wrong result, no matter any powerful solver is used. The orthogonal collocation on finite elements (OCFE) method, which is developed by Biegler et al. (2002), is the most popular discretization method, and discretizes a DAE optimization problem to a NLP model. The discretized model can approximate the original problem very well, and has high sparse structure apt to be solved. The basic principle of OCFE is implicit Runge-Kutta methods, and this method is also called IRK discretization method.

In order to describe the discretization of the system (1) more clearly, new symbol $w(t)$ is introduced to replace $\dot{x}(t)$, then the differential constraint (1b) become an algebraic one

$$w(t) = f(x(t), y(t), u(t), p),$$

and a new relationship need be supplemented as follows

$$\dot{x}(t) = w(t). \quad (2)$$

Divide optimization domain $[t_0, t_f]$ into n finite elements, then get a time series

$$S^T = \{t_0, t_1, \dots, t_n\},$$

where, the terminal time $t_f = t_n$. The duration of each element can be denoted as

$$h_i = t_i - t_{i-1}, \quad i = 1, 2, \dots, n.$$

This study considers only equidistant discretization, and all elements have the same duration $h = (t_f - t_0)/n$.

For general collocation methods, assume that each element has m collocation points, the relative position of each point can be recorded as $c_j \in [0, 1]$, $j = 1, 2, \dots, m$. In the i -th finite element, discretized variables at the j -th collocation point are denoted as $x_{i,j}$ and $w_{i,j}$, which are corresponding to $x(t)$ and $w(t)$ respectively, and ζ_i denotes the right endpoint about $x(t)$. Discretize (2) by IRK formulas (Hairer et al., 1993; Ascher and Petzold, 1998), obtain the following equations:

$$x_{i,j} - \zeta_{i-1} = h_i \sum_{k=1}^m a_{j,k} w_{i,k}, \quad j = 1, 2, \dots, m \quad (3a)$$

$$\zeta_i - \zeta_{i-1} = h_i \sum_{k=1}^m b_k w_{i,k}, \quad i = 1, 2, \dots, n \quad (3b)$$

where

$$a_{j,k} = \int_0^{c_j} \prod_{i=1, \neq k}^m \frac{c - c_i}{c_k - c_i} dc, \quad b_k = \int_0^1 \prod_{i=1, \neq k}^m \frac{c - c_i}{1 - c_i} dc.$$

There are three orthogonal collocation methods most commonly used: Gauss method, Radau method and Lobatto method. According to the solving feature of the simultaneous method, if the collocation points coincide with the endpoint of element, the discretized model can be simplified further. For example, both the first and the last collocation points of Lobatto method coincide with the front and rear endpoints of elements respectively, so connect equations can be omitted, and the continuity of derivative functions also are ensured after be discretized.

For m -Lobatto orthogonal collocation method, exist

$$\zeta_{i-1} = x_{i,1} = x_{i-1,m}, \quad w_{i,1} = w_{i-1,m}, \quad i = 1, 2, \dots, n.$$

To state the discretized formula more succinct, the coincident collocation points are considered to one point. After rearrange all collocation points, the total number of points is $N = (m-1) \cdot n$ (except the initial value point). Equation (3a) is rewritten as the following single subscript form:

$$x_i - x_{i-j} = h \cdot \sum_{k=1}^m a_{j,k} w_{i+k-j-1}, \quad (4)$$

where

$$j = ((i-1) \bmod (m-1)) + 1, \quad i = 1, 2, \dots, N.$$

Full discretized by Lobatto collocation method, model (1) can be described as the following form:

$$\min_{x_i, u_i, p} \phi(x_N) \quad (5a)$$

$$\text{s.t. } w_i = f(x_i, y_i, u_i, p), \quad (5b)$$

$$x_i - x_{i-j} = h \sum_{k=1}^m a_{j,k} w_{i+k-j-1}, \quad (5c)$$

$$g(x_i, y_i, u_i, p) = 0, \quad (5d)$$

$$h(x_N) = 0, \quad x_0 = x(t_0), \quad (5e)$$

$$x_L \leq x_i \leq x_U, \quad y_L \leq y_i \leq y_U, \quad (5f)$$

$$u_L \leq u_i \leq u_U, \quad p_L \leq p \leq p_U.$$

Furthermore, the continuity of the optimal control law need to be satisfied in many practical problems. If $u(t)$ is derivable, then a new equation can be established, that is $\dot{u}(t) = v(t)$. Ordering that $v(t)$ is constant in every

collocation element, the control variables need satisfy the following equations:

$$u_i - u_{i-j} = hv_{i-j} \sum_{k=1}^m a_{j,k}. \quad (6)$$

The scale of model (5) is proportional to the total number N of collocation point. If the collocation elements grow in number, there will be a large scale problem. Solving such problems by earlier NLP solvers most likely lead to memory overflow or system crash, due mainly to dense matrix calculation. Some new solvers, such as IPOPT, adopt the algorithms based on sparse matrix, which reduces the storage space of the gradient matrix (Jacobian and Hessian matrix) greatly, and are able to handle very large scale problems.

In this work, the process dynamic models are first principle models, and exist second-order partial derivatives for all variables. IPOPT is not only able to receive accurate Hessian Matrix, but also has flexible solution options. According to the specific situation of the exact problems, the users are able to set the solver flexibly and pertinently to accelerate the convergence speed and increase the success rate of solving (Chen et al., 2011).

3. THE INTERNAL-GROWTH SOLVING STRATEGY

3.1 The principle of fast solving strategy

For the iterative optimization algorithm, it is helpful to save solving time if the optimal solution can be estimated good enough and the estimation result is regarded as the starting point of solving (Seborg et al., 2004). This type of starting points are called advanced starting points (ASP) (Wu and Debs, 2001).

In practical applications, some systems parameters fluctuate continuously, therefore, their optimization models must be solved quickly to adapt to these real-time disturbances. If the adjustment of the parameters are small at each time, the last optimal result can be regarded as the current ASP, and it is called hot-start. However, the optimal solution of the preceding problem is usually on or near the boundary of the feasible set, making it a particularly difficult starting point for IPM (Gondzio, 1998; Gondzio and Grothey, 2008). This numerical difficulty promotes the search for an alternative starting point, which is not optimal but close to the optimal and is sufficiently far from the boundary of the feasible region. The improved approach is called warm-start (Wu and Debs, 2001; Polyak, 1992; Wright, 1997). IPOPT has the warm-start option and the initialization subprogram, which modify the starting point to satisfy the boundary conditions, i.e. IPOPT simplifies a warm-start interface to a hot-start interface, so it is applied by many optimization studies.

Unlike parametric NLPs, general DAE optimization problems can not obtain outside ASPs to promote solving process, but their particular discretization properties contribute to solving. The structure of discretized model likes digital photo, different segments can affect the clarity of details but not change the overall layout. Since the optimal solutions of discretized models can be used to approximate the original ones, the low-density discretized solutions

can be used to approximated high-density solutions for the same reason. Moreover, if the number of discretized segments is small, the discretized NLP will have small scale and can be solved very quickly. Thus, the internal-growth solving strategy is proposed: discretize the model in low-density and use IPOPT to solve it, then interpolate the optimal solution with the high-density, at last using the interpolation results as ASP to solve high-density discretized NLP again.

3.2 Integration of ASP strategy and IPOPT

Good starting point does not guarantee good solving effect, but also need suitable solver options. Consider IPOPT to solver NLPs of the form

$$\begin{aligned} \min_{x \in \mathbb{R}^l} f(x) \\ \text{s.t. } c(x) = 0, \\ x_L \leq x \leq x_U, \end{aligned} \quad (7)$$

where x denotes the vector composed of all discretized variables, l is the total number of all variables.

If IPM is used to solve problem (7), the problem is transformed to a sequence of barrier problems

$$\begin{aligned} \min_{x \in \mathbb{R}^l} f(x) - \mu \sum_{i \in I_L} \ln(x^{(i)} - x_L^{(i)}) - \mu \sum_{i \in I_U} \ln(x_U^{(i)} - x^{(i)}) \\ \text{s.t. } c(x) = 0, \end{aligned} \quad (8)$$

for a decreasing sequence of barrier parameters μ converging to zero, where $I_L = \{i : x_L^{(i)} \neq -\infty\}$, $I_U = \{i : x_U^{(i)} \neq \infty\}$ (for more details, please refer to the section 2.1 and 3.4 of Wächter and Biegler (2006)).

Equivalently, barriers problems (8) can be interpreted as applying a homotopy method to the primal-dual equations

$$\nabla f + \nabla c \lambda - z_L + z_U = 0, \quad (9a)$$

$$c = 0, \quad (9b)$$

$$(x^{(i)} - x_L^{(i)})z_L^{(i)} - \mu = 0, \quad i \in I_L, \quad (9c)$$

$$(x_U^{(i)} - x^{(i)})z_U^{(i)} - \mu = 0, \quad i \in I_U, \quad (9d)$$

with the homotopy parameter μ , which is driven to zero. Here, ∇f denotes $\nabla_x f(x)$, ∇c denotes $\nabla_x c(x)$, λ , z_L and z_U represent the Lagrangian multipliers for the equality constraints, the lower boundary and the upper boundary, respectively.

According to Lemma 3.13 in Forsgren et al. (2002), it is proved that for the NLP model (7), if the objective function $f(x)$ and the equality constraints $c(x)$ have the continuous second-order derivatives and the boundary conditions $x_L \leq x \leq x_U$ are satisfied strictly, then

$$\|\chi - \chi^*\| = O(\mu), \quad (10)$$

where χ represents $(x^T, z_L^T, z_U^T)^T$.

For IPOPT, the barrier parameter is updated by

$$\mu_{k+1} = \max \left\{ \frac{\epsilon_{\text{tol}}}{10}, \min \{ \kappa_\mu \mu_k, \mu_k^{\theta_\mu} \} \right\}, \quad (11)$$

where, $\kappa_\mu \in (0, 1)$ and $\theta_\mu \in (1, 2)$ are constants, the subscript k denotes the sequence number of current iteration, and ϵ_{tol} denotes the expectation tolerance (Wächter and Biegler, 2006). For any given initial barrier parameter

μ_0 , there exists a decreasing sequence $\{\mu_k\}$ corresponding to the given updating criterion. Hence, the decreasing sequence for a same model can be expressed by

$$M(\mu_0) = \{\mu_0, \mu_1, \dots, \mu_k, \mu_{k+1}, \dots, 0\}.$$

Assuming that a given starting point x_0 is the solution of a sub-problem (8) and is close to the optimal solution x^* enough, μ_0 can be set a small value accordint to the principle (10). Given that the decreasing sequence of μ determines the outer iteration steps, a smaller μ_0 means fewer iterations under a certain updating criterion.

The monotone updating option can be set when IPOPT runs with ASP strategy. There exists another barrier parameter updating criterion in IPOPT, i.e. adaptive criterion. This criterion is suitable to solve NLP as random starting points. During the application of internal-growth solving strategy, adaptive criterion should be adopted at the first solving, since there is no ASP in the beginning, and the monotone criterion should be used at next solving process with ASP, as well as μ_0 should be set small enough.

3.3 Interpolation for high-density ASP generation

Interpolation methods is needed to generate high-density starting points from low-density optimal results in the internal-growth approach. For using IPOPT to solve the model (5), the optimal result can be denoted as (X^*, Z_L^*, Z_U^*) , where

$$X^* = \{x_i^*, w_i^*, u_i^*, v_i^*, y_i^*, p^* | i = 1, 2, \dots, N\},$$

and the elements in Z_L^* and Z_U^* are the optimal value of Lagrangian multipliers for the lower boundary and the upper boundary with the elements in X^* . Except p^* and its Lagrangian multipliers, every discretized variables should be interpolated.

In order to maximize the reuse of data, the interpolated results are the integer multiple of original collocation points in this study. For example,

$$\{x_i^* | i = 1, 2, \dots, N\} \xrightarrow{\text{interpolation}} \{x_j^0 | j = 1, 2, \dots, mN\},$$

where, m is integer. How to decide a perfect value of m is under research, but from our practical experience, $m = 4$ is a excellent setting.

There are lots of interpolation methods can be used to obtain high-density ASPs. But for good approximate effect and avoidance of Runge's phenomenon, the piecewise polynomial interpolation is used. Figure 1 illustrates the interpolation principle, where o-points are orthogonal collocation points from low-density optimal results, and x-points are requested interpolation points.

3.4 Procedure of the internal-growth approach

The following steps describe how to implement the internal-growth strategy to solve a DAE optimization problems in AMPL environment:

- (1) Use Lobatto-III method to discretize the original dynamic model and get discretized NLP model in forms like (5).
- (2) Set a small number of discretized elements, e.g. $N = 50$ in this study, and set IPOPT options, especially updating strategy of the barrier parameter (`mu_strategy`) as "adaptive", then run solving.

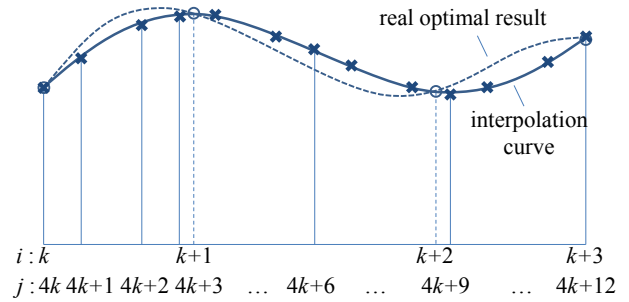


Fig. 1. Using 4 times collocation points to generate ASP

- (3) If solve successfully, save the optimal solution and Lagrange multipliers of the corresponding boundary constraints.
- (4) Increase the amount of elements, e.g. $N = 200$ in this study, and interpolate the recorded data by the new amount.
- (5) Adjust IPOPT options, set warm-start option as "yes", μ update policy as "monotone", and μ initial value (`mu_init`) as a very tiny value, in this study it is 1×10^{-10} .
- (6) Import the interpolation result from step 4 as the ASP, and solve again.

4. NUMERICAL EXPERIMENTS

4.1 Crystallization process

To illustrate the effectiveness of internal-growth solving strategy, a more complex example is considered as follows. The dynamic optimization model of crystallization process was developed by Lang et al. (1999), and adopted by several research as test case (Biegler, 2007, 2010). Conventional crystallization kinetics are characterized in terms of two dominant phenomena: nucleation and crystal growth. Both competing phenomena consume desired solute material during the crystallization process. To obtain larger (and fewer) crystals, nucleation needs to be minimized, and the goal of the optimization is to find operating strategies that will allow researchers to minimize this phenomenon. To do this, a profile for the cooling jacket temperature is determined, and described by differential constraints as follows:

$$\begin{aligned} \dot{L}_s &= K_g L_s^{0.5} \Delta T^{1.1}, & L_s(0) &= 0.0005, \\ \dot{N} &= B_n \Delta T^{5.72}, & N(0) &= 0, \\ \dot{L} &= N \cdot \dot{L}_s + L_0 \dot{N}, & L(0) &= 0, \\ \dot{A} &= 2\alpha L \cdot \dot{L}_s + L_0^2 \dot{N}, & A(0) &= 0, \\ \dot{V}_c &= 3\beta A \cdot \dot{L}_s + L_0^3 \dot{N}, & V_c(0) &= 0, \\ \dot{M} &= \frac{3W_{s0}}{L_{s0}^3} L_s^{2.5} \dot{L}_s + \rho V \cdot \dot{V}_c, & M(0) &= 2.0, \\ \dot{C} &= -\frac{\dot{M}}{V}, & C(0) &= 5.4, \\ \dot{T} &= \frac{K_c \dot{M} - K_e(T - T_j)}{W \cdot C_p}, & T(0) &= 75. \end{aligned} \quad (12)$$

The objective function is to maximize $L_s(t_f)$. All of the definitions of variables and parameters are reported

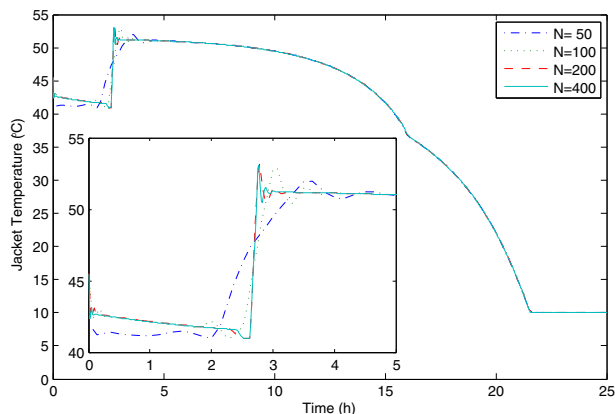


Fig. 2. Optimal control laws under different discretization elements

in Lang et al. (1999). The following algebraic equation constraints are also need to be satisfied:

$$\Delta T = \max(0, T_{\text{equ}} - T), \quad C_w = \frac{100C}{1.35 + C}, \quad (13)$$

$$T_{\text{equ}} = \sum_{i=1}^4 a_i C_w^{i-1}, \quad T_{\text{alw}} = \sum_{i=1}^4 b_i C_w^{i-1},$$

where, $T_{\text{alw}} \leq T_j \in [10, 100]$, and polynomial coefficients of correlation are

$$a = [-66.431, 2.8604, -0.022579, 6.7117 \times 10^{-5}],$$

$$b = [16.088, -2.7083, 0.067069, -3.5685 \times 10^{-4}].$$

This is batch process, and set $t_f = 25\text{h}$.

According to the difference of the discretization density, four groups of the optimal control variable (i.e. the temperature curves of the cooling water jacket) corresponding to the different N are compared in figure 2. It is shown that the overall trends are consistent even though the different segments. The optimal solution of the problem is a curve with discontinuous point, and it is approximated by continuous function during the discretization process. Hence, the sudden change can not be illustrated by the low-density discretization. Unless segmented enough, the solution curve with the discontinuous or dramatic changes can not be fitted sufficiently.

In order to illustrate the solving effect of the internal-growth strategy sufficiently, let $N = 200$, and use IPOPT to solve under different options. The relevant statistical data is displayed in table 1. By comparing data in the table, it is shown that the algorithm employing ASP has relatively higher efficiency than the situation without ASP. If no ASP, the adaptive option is suitable for barrier parameter updating strategy; if with ASP, monotone updating strategy matching with the smaller μ_0 will achieve better calculation efficiency. For this experiment, it costs 38.53s at least to obtain optimal solution by ordinary strategy, but internal-growth strategy with best option spends only 6.63s to complete solving, which is no more than 1/5 of the former.

The data in table 1 and above analysis sufficiently illustrate the feasibility of the internal-growth strategy, but the reliability of the strategy still need to be verified. The solving principles usually assume that the optimal solution

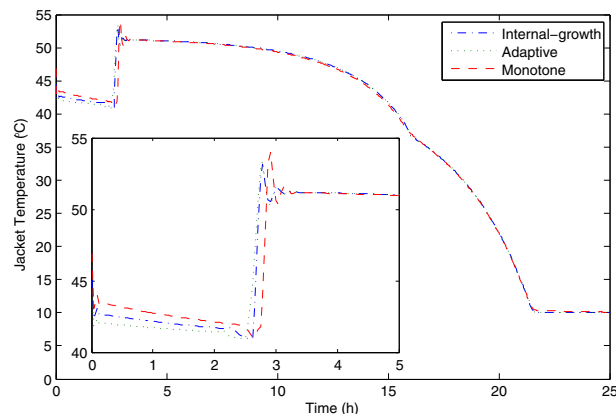


Fig. 3. Optimal control laws by three solving settings when $N = 200$

exists and is unique. However, it can not be guaranteed that NLPs are always convex in practical application, so the solutions obtained by the solvers like IPOPT are local optimums actually. There often exists difference for the results solved by different strategies, since different starting points and parameter settings lead to the changes of convergence path. But the deviation is tiny and will not affect consistency of the optimal solution. Figure 3 compares the optimal control curves under three solving setting when $N = 200$. The three settings are the internal-growth strategy, the common method with monotone μ updating criteria, and the common method with adaptive μ updating criteria. Obviously, the three results are consistent, and the result from the internal-growth strategy is very close to the solution from the adaptive common strategy. The consistency of the solutions certifies sufficiently that the internal-growth strategy is reliable.

4.2 More tests

In order to test the scalability of the internal-growth strategy, we select five typical dynamic optimization problems from Rutquist and Edvall (2010), i.e. (1) hang glider, (2) Goddard rocket, (3) Lee-Ramirez bioreactor, (4) Park-Ramirez bioreactor, and (5) batch production. Figure 4 displays the comparison of the results. The solving environment is same as the previous example. For each test case, the bar above indicates solving time by using the internal-growth strategy, which includes two steps; the bar below indicates solving cost with common program. The figure illustrates the internal-growth strategy is valid for

Table 1. Data statistics of the crystallization model under different solving options

N	warm-start	μ -update	μ_0	iters	CPU-time(s)
50	off	adaptive	-	240	+5.28
200	on	adaptive	-	21	7.61(2.33)
200	on	monotone	1E-01	61	11.03(5.75)
200	on	monotone	1E-08	24	7.80(2.52)
200	on	monotone	1E-10	14	6.63(1.35)
200	off	adaptive	-	492	46.09
200	off	monotone	1E-01	422	38.53
200	off	monotone	1E-08	782	68.71
200	off	monotone	1E-10	1005	105.46

Note: The solving platform is HP Z820 Workstation with Intel Xeon CPU E5-2680.

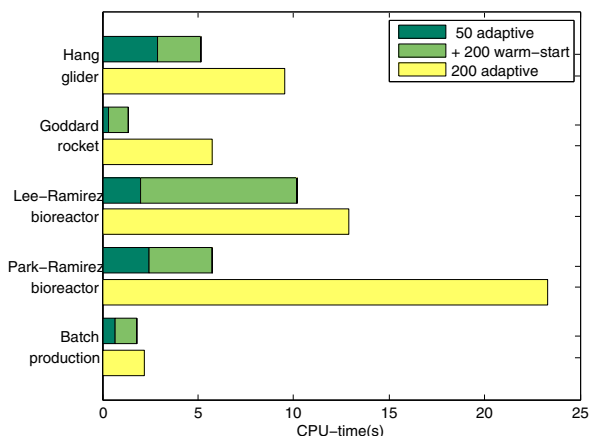


Fig. 4. Time consuming comparison of five examples for using the internal-growth strategy

all cases, especially the more complicate ones. However, the application effects are not distinct for some problems, the reason requires the further research on convergence mechanism to find out.

5. CONCLUSION

This work has analyzed the feature of OCFE discretization with the Lobatto collocation pattern for the DAE optimization models, and the discretized NLP can be described more concisely due to the coincidence of the finite element endpoints and collocation points. Since the discretization model is an approximation of the original problem, higher solving accuracy requires the more segment elements. However, the more finite elements will generate a larger scale problem, which will consume more computer resource on solving. According to the structural similarity under different discrete segments, the optimal solution with low-density is analogous as the high-density result. Integration of structural similarity, the warm-start technology for IPM and suitable barrier parameter updating criterion can bring out the internal-growth strategy to achieve the fast solving of dynamic optimization.

There exist some theoretical research works need to be enriched, especially the assessment of convergence caused by error from warm-start interpolation. In future work, we are also looking for the integration of the internal-growth strategy into IPM solvers.

REFERENCES

Ascher, U.M. and Petzold, L.R. (1998). *Computer Methods for Ordinary Differential Equations and Differential-Algebraic Equations*. SIAM, Philadelphia, PA.

Biegler, L.T. (2007). An overview of simultaneous strategies for dynamic optimization. *Chemical Engineering and Processing*, 46(11), 1043–1053. doi:10.1016/j.cep.2006.06.021.

Biegler, L.T. (2010). *Nonlinear Programming: Concepts, Algorithms, and Applications to Chemical Processes*. SIAM, Philadelphia, PA.

Biegler, L.T., Cervantes, A.M., and Wächter, A. (2002). Advances in simultaneous strategies for dynamic process optimization. *Chemical Engineering Science*, 57(4), 575–593. doi:10.1016/S0009-2509(01)00376-1.

Chen, W., Shao, Z., Wang, K., Chen, X., and Biegler, L.T. (2011). Random sampling-based automatic parameter tuning for nonlinear programming solvers. *Industrial & Engineering Chemistry Research*, 50(7), 3907–3918. doi:10.1021/ie100826y.

Forsgren, A., Gill, P.E., and Wright, M.H. (2002). Interior methods for nonlinear optimization. *SIAM Review*, 44(4), 525–597. doi:10.1137/S0036144502414942.

Gondzio, J. (1998). Warm start of the primal-dual method applied in the cutting-plane scheme. *Mathematical Programming*, 83(1), 125–143. doi:10.1007/BF02680554.

Gondzio, J. and Grothey, A. (2008). A new unblocking technique to warmstart interior point methods based on sensitivity analysis. *SIAM Journal on Optimization*, 19(3), 1184–1210. doi:10.1137/060678129.

Hairer, E., Nørsett, S.P., and Wanner, G. (1993). *Solving Ordinary Differential Equations I: Nonsiff Problems*. Springer-Verlag, Berlin Heidelberg, GER, 2 edition.

Kameswaran, S. and Biegler, L.T. (2006). Simultaneous dynamic optimization strategies: Recent advances and challenges. *Computers & Chemical Engineering*, 30(10-12), 1560–1575. doi:10.1016/j.compchemeng.2006.05.034.

Kunkel, P. and Mehrmann, V. (2006). *Differential-algebraic equations analysis and numerical solution*. European Mathematical Society, Switzerland.

Lang, Y.D., Cervantes, A.M., and Biegler, L.T. (1999). Dynamic optimization of a batch cooling crystallization process. *Industrial & Engineering Chemistry Research*, 38(4), 1469–1477. doi:10.1021/ie980585u.

Polyak, R. (1992). Modified barrier functions (theory and methods). *Mathematical Programming*, 54(1-3), 177–222. doi:10.1007/BF01586050.

Rutquist, P.E. and Edvall, M.M. (2010). PROPT - MATLAB optimal control software. URL http://tomopt.com/docs/TOMLAB_PROPT.pdf.

Seborg, D.E., Edgar, T.F., and Mellichamp, D.A. (2004). *Process Dynamics and Control*, chapter 19, 510–533. John Wiley & Sons, Hoboken, NJ, 2 edition.

Wächter, A. and Biegler, L.T. (2006). On the implementation of an interior-point filter line-search algorithm for large-scale nonlinear programming. *Mathematical Programming*, 106(1), 25–57. doi:10.1007/s10107-004-0559-y.

Wright, S.J. (1997). *Primal-Dual Interior-Point Methods*, chapter 11, 224–225. Society for Industrial and Applied Mathematics, Philadelphia, US, 1 edition.

Wu, Y.C. and Debs, A.S. (2001). Initialisation, decoupling, hot start, and warm start in direct nonlinear interior point algorithm for optimal power flows. *IEEE Proceedings: Generation, Transmission & Distribution*, 148(1), 67–75. doi:10.1049/ip-gtd:20010008.