Proceedings of the Federated Conference on
Computer Science and Information Systems pp. 477–484
ISBN 978-83-60810-22-4

A modified multipoint shooting feasible-SQP method for optimal control of DAE systems

Krystyn Styczeń
Institute of Computer Engineering
Wrocław University of Technology
Janiszewskiego 11/17, 50-372 Wrocław, Poland
Email: krystyn.styczen@pwr.wroc.pl

Paweł Drąg
Institute of Computer Engineering
Wrocław University of Technology
Janiszewskiego 11/17, 50-372 Wrocław, Poland
Email: pawel.drag@pwr.wroc.pl

***Abstract*—Optimal control problem for state-constrained differential-algebraic (DAE) systems is considered. Such problems can be attacked by the multiple shooting approach well suited to unstable and ill-conditioned dynamic systems. According to this approach the control interval is partitioned into shorter intervals allowing the parallelization of computations with the reliable using of DAE solvers. A new modified method of this kind is proposed, which converts the partitioned problem with mixed equality and inequality constraints into the purely inequality constrained problem. An algorithm for obtaining a feasible initial solution of the converted problem is described. A feasible-SQP algorithm based on an active set strategy is applied to the converted problem. It avoids the inconsistency of the constraints of the QP subproblems (versus the infeasible path SQP methods) and delivers a locally optimal solution of the basic problem preserving all its constraints (including the equality ones), which is of a high practical meaning. Some further developments concerning the regularization of suboptimal solutions for large-scale DAE optimal control problems and multilevel versions of the method proposed are also discussed. The theoretical considerations are illustrated by a numerical example of optimization of a complex DAE chemical engineering system.**

I. Introduction

THE DEVELOPMENT of advanced technologies is often connected with the design of complex systems described by large-scale differential-algebraic equations (DAE) subject to the control and state path constraints, and to the terminal state constraints [3],[4]. Many of such systems possess unstable dynamic modes and high sensitivity to parameter changes. This may lead to unbounded state profiles and to ill-conditioning of the Jacobian and the Hessian matrices if the single shooting method is used to optimize the system performance. This also complicates the optimization of dynamic systems with boundary value conditions. The multiple shooting (MS) method has been proposed to resolve the above difficulties [11]. It discretizes the control interval into shorter subintervals within which the dynamic model is integrated independently and linked by the continuity constraints in the discretized model. Such an approach is well suited to deal with unstable and highly sensitive modes of nonlinear dynamic systems. Moreover, it guarantees the full parallelization of the data computations necessary for the application of advanced second-order optimization methods such as the sequential quadratic programming (SQP) method or the interior point (IP) method [4]. Many further advantages of the MS method concerning the exploitation of the problem sparse structure, the computation of the state sensitivities by the specialized algorithms, and the employment of the reduced or partially reduced SQP algorithms are depicted in the literature [11]. However, the main elaborations on this subject are connected with the iterative infeasible path approach, where all the constraints may be violated on the current iteration, and reached solely at the limit of the subsequence of convergent iterations. This may complicate the SQP algorithm because of the inconsistency of the constraints for the QP subproblem, and the need of its regularization, for example, by the homotopy approach [4]. Thus the current solution obtained at the moderate computation time may be impractical for its infeasibility.

In the present paper the feasible-SQP approach, known in many variants as the nonlinear programming methods guaranteeing the feasibility of the current solution at all the optimization iterations [6],[10],[15], is developed as a method specialized in the multiple shooting approach to the optimal control of DAE systems [5]. One of the main difficulties in the use of the feasible-SQP algorithms is the requirement of the knowledge of an initial feasible solution satisfying all the constraints. The problem of the determination of such an solution may be as difficult as the basic optimization problem. We show, however, that the exploitation of the specific structure of the MS method enables us to conveniently resolve this problem. To this end we apply the c-conversion [10] of the partitioned shooting problem with mixed equality and inequality constraints into the purely inequality problem. We describe an algorithm, which determines a feasible initial solution of the converted problem exploiting the results of the consecutive shots. Next we present a modified algorithm for the choice of the conversion coefficient, which employs the MATLAB R2010b feasible-SQP active set procedure. It avoids the inconsistency of the constraints of the QP subproblems and allows the full parallelization of the optimization computations. We verify the algorithms developed by a numerical example of optimization of a complex DAE chemical engineering system. Finally we discuss the regularization of suboptimal solutions for large-scale DAE optimal control problems with the help of bound constrained Newton method [2], [14], and multilevel versions of the method proposed [7].

Notation: $L^n_\infty(t_0,t_f)$, $W^{1,n}_\infty(t_0,t_f)$ and $PC^n(t_0,t_f)$ the spaces of n-dimensional essentially bounded, essentially bounded derivative, and piecewise continuous functions defined on the interval $[t_0,t_f]$, respectively; $S^n_r(t_0,t_f)$ the space of n-dimensional r-step functions defined on the interval $[t_0,t_f]$, n_x the dimension of a vector x, $|x|_\infty \doteq max\{|x_n| : n=1,2,...,n_x\}$.

II. Optimal control problem for DAE systems

Consider the following optimal control problem for DAE systems (the D problem): minimize the objective function

$$\mathcal{J}(x,z,u,p) \doteq h(x(1),z(1),p) \tag{1}$$

subject to a system of differential-algebraic equations of index one

$$\dot{x}(t) = f(x(t),z(t),u(t),p,t), \quad t\in[0,1], \tag{2}$$

$$0 = g(x(t),z(t),u(t),p,t), \quad t\in[0,1], \tag{3}$$

to the terminal constraint

$$\hbar(x(1),z(1),p)=0, \tag{4}$$

to the bound constraints

$$x(t)\in X, \quad z(t)\in Z, \quad u(t)\in U, \quad t\in[0,1], \quad p\in P, \tag{5}$$

and to the physical realizability condition for the control

$$u\in PC^{n_u}(0,1), \tag{6}$$

where $x\in W^{1,n_x}_\infty(0,1)$ is the differential state trajectory of the DAE system, $z\in L^{n_z}_\infty(0,1)$ is its algebraic state trajectory, $u\in L^{n_u}_\infty(0,1)$ is its control, $p\in R^{n_p}$ is its global parameter, and $X\doteq[x_-,x_+]$, $Z\doteq[z_-,z_+]$, $U\doteq[u_-,u_+]$, and $P\doteq[p_-,p_+]$ are parallelepipeds with the bounds $x_\pm\in R^{n_x}$, $z_\pm\in R^{n_z}$, $u_\pm\in R^{n_u}$ and $p_\pm\in R^{n_p}$, and the functions

$$h: R^{n_x}\times R^{n_z}\times R^{n_p}\to R, \quad \hbar: R^{n_x}\times R^{n_z}\times R^{n_p}\to R^{n_\hbar},$$

$$f: R^{n_x}\times R^{n_z}\times R^{n_u}\times R^{p}\times R\to R^{n_x},$$

$$g: R^{n_x}\times R^{n_z}\times R^{n_u}\times R^{n_p}\times R\to R^{n_z}$$

are twice continuously differentiable in all their arguments.

We normalize the nonunit control interval $[0,\tau], \tau\neq 1$ by the time scaling $t:=t/\tau$. We include the variable process duration τ into the global parameter p. The latter parameter may also concern the design variables (such as the level of the fixed bed catalyst or the reactor volume) and the slack variables converting the terminal inequality constraints into the equality ones. We reduce the general control and state inequality path constraints $q(x(t),z(t),u(t),t)\leq 0$ to the equality form (3) with the help of the slack control $\tilde{u}(t)\geq 0$ satisfying the condition $q(x(t),z(t),u(t),t)+\tilde{u}(t)=0$, $t\in[0,\tau]$. Thus the formulation (1)-(6) encompasses a wide class of optimal control problems for DAE systems.

III. Modified multipoint shooting feasible-SQP method

We use the discretized time $t_k = k/l \quad (k=0,1,...,l)$. We connect with the time interval $[t_k,t_{k+1}]$ its shooting initial differential state $x_k\in R^{n_x}$, its shooting initial algebraic state $z_k\in R^{n_z}$, its shooting control parameters $u_k \doteq (u^T_{k1},u^T_{k2},...,u^T_{kr_k})^T \in R^{n_{u_k}}$ $(n_{u_k}\doteq n_u r_k,\ r_l = 0,\ u_{l1}\doteq u_{l-1,r_{l-1}})$, its shooting global parameter $p_k\in R^{n_p}$, and its shooting solution $w_k\doteq(x^T_k,z^T_k,u^T_k,p^T_k)^T\in R^{w_k}$ $(n_{w_k}\doteq n_x+n_z+n_{u_k}+n_p)$, its differential state trajectory $\widetilde{x}_k\in W^{1,n_x}_\infty(t_k,t_{k+1})$ determined by w_k, its algebraic state trajectory $\widetilde{z}_k\in L^{n_z}_\infty(t_k,t_{k+1})$ determined by w_k, and its control $\widetilde{u}_k\in S^{n_u}_{r_k}(t_k,t_{k+1})$ determined by u_k. We reformulate the D problem as the multipoint shooting problem for DAE systems (the MSD problem): minimize the objective function

$$J(w)\doteq h(x_l,z_l,p_l) \tag{7}$$

subject to the continuity conditions for the differential state trajectory and the discretized parameters

$$\widetilde{x}_k(t_{k+1},w_k)-x_{k+1}=0, \quad p_k-p_{k+1}=0 \quad (k=0,1,...,l-1), \tag{8}$$

to the consistency conditions for the algebraic states

$$g(x_k,z_k,u_{k1},p_k,t_k)=0 \quad (k=0,1,...,l), \tag{9}$$

to the terminal equality constraints

$$\hbar(x_l,z_l,p_l)=0, \tag{10}$$

and to the bound constraints

$$x_k\in X, \quad z_k\in Z, \quad u_k\in U_k, \quad p_k\in P \quad (k=0,1,...,l), \tag{11}$$

where $w\doteq(w^T_1,w^T_2,...,w^T_l)^T\in R^{n_w}$ $(n_w\doteq n_{w_1}+n_{w_2}+...+n_{w_l})$ is the solution of the MSD problem, and $U_k\doteq[u_{k-},u_{k+}]$, $u_{k\pm}\doteq(\underbrace{u^T_\pm,u^T_\pm,...,u^T_\pm}_{r_k\ times})^T$.

Let $X_\varepsilon \doteq [\varepsilon+x_-,-\varepsilon+x_+]$ $(\varepsilon\in R^{n_x}_+)$ and $Z_\epsilon\doteq[\epsilon+z_-,-\epsilon+z_+]$ $(\epsilon\in R^{n_z}_+)$ be the restricted bound sets for the differential and algebraic states, and let $\varepsilon_n,\epsilon_n,\widetilde{x}_{kn},x_{kn},p_{kn},x_{n-},x_{n+},z_{n-},z_{n+},g_n$ and $\hbar_n$ be the nth coordinates of the quantities $\varepsilon,\epsilon,\widetilde{x}_k,x_k,p_k,x_-,x_+,z_-,z_+,g$ and $\hbar$.

Algorithm 1: The conversion of the MSD problem (7)-(11) to the parametric MSD$_c$ problem with a known feasible initial solution.

Step 0: Choose $\varepsilon=0.05(x_+-x_-)$, $\epsilon=0.05(z_+-z_-)$, $\breve{x}_0\in X_\varepsilon$, $\breve{z}_0\in Z_\epsilon$, $\breve{u}_0\in U_0$ and $\breve{p}_0\in P$, set $\breve{w}_0\doteq(\breve{x}^T_0,\breve{z}^T_0,\breve{u}^T_0,\breve{p}^T_0)^T$ and $k=0$.

Step 1: If $k=l$ go to *Step 4*. Else determine the differential and algebraic state trajectories $\widetilde{x}_k$ and $\widetilde{z}_k$ by the shot in the kth time interval.

Step 2: Using the results of the current shot determine

- the consecutive shooting differential state $\breve{x}_{k+1,n}=\varepsilon_n+x_{n-}$ if $\widetilde{x}_{kn}(t_{k+1})\leq\varepsilon_n+x_{n-}$, and $\breve{x}_{k+1,n}=\widetilde{x}_{kn}(t_{k+1})$ if $\widetilde{x}_{kn}(t_{k+1})\in(\varepsilon_n+x_{n-},-\varepsilon_n+x_{n+})$, and $\breve{x}_{k+1,n}=-\varepsilon_n+x_{n+}$ if $\widetilde{x}_{kn}(t_{k+1})\geq-\varepsilon_n+x_{n+}$ $(n=1,2,...,n_x)$,

• the consecutive shooting algebraic state $\breve{z}_{k+1,n} = \epsilon_n + z_{n-}$ if $\widetilde{z}_{kn}(t_{k+1}) \le \varepsilon_n + z_{n-}$, and $\breve{z}_{k+1,n} = \widetilde{z}_{kn}(t_{k+1})$ if $\widetilde{z}_{kn}(t_{k+1}) \in (\varepsilon_n + z_{n-}, -\varepsilon_n + z_{n+})$, and $\breve{z}_{k+1,n} = -\varepsilon_n + z_{n+}$ if $\widetilde{z}_{kn}(t_{k+1}) \ge -\varepsilon_n + z_{n+}$ $(n = 1, 2, ..., n_z)$, and choose the consecutive shooting control $\breve{u}_{k+1} \in U_{k+1}$, and the consecutive shooting parameter $\breve{p}_{k+1} = \breve{p}_k$, and denote the solution found for the consecutive interval as $\breve{w}_{k+1} \doteq (\breve{x}^T_{k+1}, \breve{z}^T_{k+1}, \breve{u}^T_{k+1}, \breve{p}^T_{k+1})^T$.

Step 3: Determine

• the defect functions for the shooting differential states $G_{1kn}(w) \doteq \widetilde{x}_{kn}(t_{k+1}) - x_{k+1,n}$ if $\widetilde{x}_{kn}(t_{k+1}) \le \varepsilon_n + x_{n-}$ or $\widetilde{x}_{kn}(t_{k+1}) \in (\varepsilon_n + x_{n-}, -\varepsilon_n + x_{n+})$, and $G_{1kn}(w) \doteq -\widetilde{x}_{kn}(t_{k+1}) + x_{k+1,n}$ if $\widetilde{x}_{kn}(t_{k+1}) \ge -\varepsilon_n + x_{n+}$ $(n = 1, 2, ..., n_x)$,

• the defect functions for the shooting algebraic states $G_{2kn}(w) \doteq g_n(x_k, z_k, u_{k1}, p_k, t_k)$ if $g_n(x_k, z_k, u_{k1}, p_k, t_k) \le 0$, and $G_{2kn}(w) \doteq -g_n(x_k, z_k, u_{k1}, p_k, t_k)$ in the opposite case $(n = 1, 2, ..., n_z)$,

• the defect functions for the shooting parameter $G_{3kn}(w) \doteq p_k - p_{k+1}$. Set $k = k + 1$.

Step 4: Determine the defect functions for the terminal constraints $G_{ln}(w) \doteq \hbar_n(x_l, z_l, p_l)$ if $\hbar_n(x_l, z_l, p_l) \le 0$, and $G_{ln}(w_l) \doteq -\hbar_n(x_l, z_l, p_l)$ in the opposite case $(n = 1, 2, ..., n_\hbar)$. Save the solution found as $\breve{w} \doteq (\breve{w}^T_1, \breve{w}^T_2, ..., \breve{w}^T_l)^T$.

Step 5: Set up the functions required for the formulation of the MSD$_c$ problem:

$$G_{1k}(w) \doteq (G_{1kn}(w))^{n_x}_{n=1}, \;\; G_{2k}(w) \doteq (G_{2kn}(w))^{n_z}_{n=1},$$

$$G_{3k}(w) \doteq (G_{3kn}(w))^{n_p}_{n=1},$$

$$G_k(w) \doteq (G^T_{1k}(w), G^T_{2k}(w), G^T_{3k}(w))^T \;\; (k = 0, 1, ..., l-1),$$

$$G_l(w) \doteq (G_{ln}(w))^{n_\hbar}_{n=1},$$

$$G_{l+1+k}(w) \doteq (-w^T_k + w^T_{k-}, w^T_k - w^T_{k+})^T,$$

$$w_{k\pm} \doteq (x^T_\pm, z^T_\pm, u^T_{k\pm}, p^T_\pm)^T \;\; (k = 0, 1, .., l).$$

Step 6: State the MSD$_c$: minimize the objective function

$$J_c(w) \doteq J(w) - c \sum_{k=0}^{l} G_k(w_k) \tag{12}$$

subject to the constraints

$$G_k(w) \le 0 \;\; (k = 0, 1, ..., 2l + 1). \tag{13}$$

where $c \in R_+$ is the cost coefficient of the problem conversion.

If the coefficient c is sufficiently large the MSD problem and the MSD$_c$ problem have the same KKT points [12],[8],[10],[15]. *Algorithm 1* yields by its formulation a feasible solution $\breve{w}$ of the MSD$_c$ problem, which can be further assumed as an initial solution $w^0 \doteq \breve{w}$ for efficient feasible-SQP type algorithms solving this problem. The issues concerning a suitable choice of the coefficient c, and the verification of the feasibility of the MSD problem (and eventually of the D problem) by an optimal solution of the MSD$_c$ problem are taken up by

Algorithm 2: The search for a locally optimal solution w^* of the MSD problem and for a locally suboptimal solution (x^*, z^*, u^*, p^*) of the D problem by the multipoint shooting feasible-SQP (MSFSQP) method.

Step 0: Input the initial solution w^0 found by *Algorithm 1*, a symmetric positive definite matrix $H \in R^{n_w \times n_w}$, and positive constants $c, \bar{c}$ and $\varrho > 1$.

Step 1: Use the Matlab R2010b feasible-SQP active set procedure to find a locally optimal solution $w(c)$ of the MSD$_c$ problem, and the Lagrange multipliers $\lambda_k(c)$ associated with the constraints $G_k(w)$ $(k = 0, 1, ..., l)$.

Step 2: If $c < \lambda_+(c) \doteq max\{|\lambda_k(c)|_\infty, \; k = 0, 1, ..., l\}$ set $c := \lambda_+(c) + \bar{c}$ and return to *Step 1*.

Step 3: If $\sum_{k=0}^{l} |G_k(w(c))|_\infty = 0$ set $w^* = w(c)$. Else set $c := c + \bar{c}$.

Step 4: If the bound constraints (5) for the differential states $\widetilde{x}_k(t, w^*)$ and for the algebraic states $\widetilde{z}_k(t, w^*)$, $t \in [t_k, t_{k+1}]$, $(k = 0, 1, ..., l-1)$ are satisfied determine a locally suboptimal feasible solution of the D problem as $x^*(t) = \widetilde{x}_k(t, w^*)$, $z^*(t) = \widetilde{z}_k(t, w^*)$, $u^*(t) = \widetilde{u}_k(t, w^*)$, $p^*(t) = \widetilde{p}_k(t, w^*)$, $t \in [t_k, t_{k+1}]$, $(k = 0, 1, ..., l-1)$. Else set $\varepsilon := \varrho\varepsilon$ and $\epsilon := \varrho\epsilon$ and go to *Step 0*.

The algorithm exploits the equivalence of the KKT points of the MSD$_c$ and MSD problems for sufficiently large c, which should exceed the maximum modulus of the Lagrange multipliers for the converted constraints $G_k(w)$ $(k = 0, 1, ..., l)$ (the c-condition). If this condition is violated the coefficient c is increased (*Step 2*), and the optimization process is repeated. Else the fulfilling of the equality constraints of the MSD problem is verified. It can be violated even if the c-condition is satisfied for numerical errors propagation in large-scale DAE systems. Then some further increase of the coefficient c may be helpful (*Step 3*). The violation of the bound constraints for the differential and algebraic states can be removed by the manipulation of the parameters ε and ϵ in view of the calmness of the DAE systems under discussion [13]. This leads to a locally suboptimal feasible solution of the basic D problem (*Step 4*).

The regularization of the solution $w(c)$ satisfying insufficiently accurately the equality constraints of the D problem may concern the application of the bound-constrained trust-region and inexact Newton method. In particular the consistent algebraic states for large-scale DAE systems can be found with the help of the superlinearly convergent trust-region approach [1],[2]: minimize in δz_k the quadratic model of the consistency equations

$$\frac{1}{2}|g(x_k, z_k, u_{k1}, p_k, t_k) + g'_{z_k}(x_k, z_k, u_{k1}, p_k, t_k)\delta z_k|^2$$

subject to the trust-region constraints

$$|\mathcal{D}\delta z_k| \le \mathit{\Delta},$$

where $\mathcal{D}$ is the scaling matrix and $\mathit{\Delta}$ is the trust-region matrix. This approach may be yet enhanced by the bound-constrained inexact Newton method [14] applied to the consistency equations (9) in z_k

$$z_k^+ = z_k + \alpha \delta z_k, \quad z_k^+ \in Z_k,$$

$$|g(x_k, z_k^+, u_{k1}, p_k, t_k) + g'_{z_k}(x_k, z_k, u_{k1}, p_k, t_k)\delta z_k|$$

$$< \beta |g(x_k, z_k, u_{k1}, p_k, t_k)|,$$

where $\alpha,\ \beta \in (0, 1)$.

The feasibility of the solution obtained may facilitate the incorporation of the fixed dimension l multiple shooting method into the variable dimension l multiple shooting method exploiting the multilevel feasible approach based on the convergence of point-to-set mappings [7]. The imposition of the lower dimension variations of the shooting controls on the fine dimension shooting solution does not destroy the problem feasibility because the zero solution is feasible in the lower dimension problem.

A wide class of complex DAE systems is encountered in chemical engineering. This concerns, for example, processes of nonlinear chemical reactions performed in tank reactors or multizone reactors, and processes of heat exchange, distillation and separation [4]. A high practical meaning has the optimization of integrated processes of such a kind, which leads to the problems with complex DAE models requireing advanced optimization methods. The application of the above described algorithms to optimal control of some chemical engineering systems is proposed with the use of the Matlab toolbox of the parallel computations.

IV. Numerical examples

The new method, which we presented above, now we would like to use to solve a certain D problem [9]. Before we solve this task with the presented method and give results, we are going to introduce this problem and its details. Then we are going to be able to better understand the method presented in this paper.

Y. J. Huang et all ([9]) described a model decomposition based method for solving general dynamic optimization problems. The authors gave in their paper three interesting examples from chemical engineering. There were catalyst mixing problem, fed-batch penicillin fermentation and pressure-constrained batch reactor. From our point of view the most interesting is the 3^{rd} problem.

In the reactor three reactions take place. $A \to 2B$, with reaction constant k_1. There is a reverse reaction $2B \to A$, with reaction constant k_2. The last reaction is $A + B \to D$, with reaction constant k_3. Description of the dynamic optimization problem is as follows

$$min_F J = C_D(t_f),$$

subject to

$$\dot{C_A} = -k_1 C_A + k_2 C_B C_B + \frac{F}{V} - k_3 C_A C_B,$$

$$\dot{C_B} = k_1 C_A - k_2 C_B C_B - k_3 C_A C_B,$$

$$\dot{C_D} = k_3 C_A C_B.$$

There are the algebraic and state constraints too

$$N = V(C_A + C_B + C_D),$$

$$PV = NRT,$$

$$P \leq 340000[Pa],$$

$$0 \leq F \leq 8.5[\frac{mol}{h}],$$

together with initial conditions

$$[C_A(0), C_B(0), C_D(0)] = [100, 0, 0].$$

We know, that $t_f = 2$ hours.

To complete the description, there are values of another magnitudes used in equations: $k_1 = 0.8[\frac{1}{h}]$, $k_2 = 0.02[\frac{m^3}{mol \cdot h}]$, $k_3 = [\frac{m^3}{mol \cdot h}]$, the volume $V = 1.0[m^3]$ and the temperature $T = 400[K]$.

For purposes of our presentation, we can rewrite the constraints equations. Because

$$N = V(C_A + C_B + C_D)$$

and

$$PV = NRT,$$

so we have two eguations with two unknowns N and P. Then we can write

$$PV = V(C_A + C_B + C_D)RT.$$

Now, because $V = 1 \Rightarrow V \neq 0$, in the next step we can state, that

$$P = (C_A + C_B + C_D)RT.$$

We know, that the gas constant equals to $R = 8.314472[\frac{J}{mol \cdot K}]$ and in this situation the temperature is constant too, so

$$P \leq 340000 \Rightarrow (C_A + C_B + C_D)RT \leq 340000.$$

The last step is to compute the constraint explicite

$$(C_A + C_B + C_D) \leq \frac{340000}{RT} \Rightarrow (C_A + C_B + C_D) \leq 102.2314[\frac{mol}{m^3}].$$

We can check our calculations

$$[\frac{Pa}{\frac{J}{mol \cdot K} \cdot K}] = [\frac{Pa \cdot mol \cdot K}{J \cdot K}] = [\frac{Pa \cdot mol}{J}] =$$

$$= [\frac{\frac{N}{m^2} \cdot mol}{\frac{kg \cdot m^2}{s^2}}] = [\frac{\frac{kg \cdot m}{s^2 \cdot m^2} \cdot mol}{\frac{kg \cdot m^2}{s^2}}] = [\frac{kg \cdot mol \cdot s^2}{s^2 \cdot m \cdot kg \cdot m^2}] = [\frac{mol}{m^3}].$$

Now this problem has another constraints, which have the same meaning: one state constraint and one constrained control variable.

To better understand this problem, we made some easy simulations for various constant control variable with using single shooting method. All simulation were made in Matlab R2010b, with settings $RelTol = 10^{-7}$, $AbsTol = 10^{-7}$ for DAE solver $ode15s$, $TolFun = 10^{-7}$, $TolX = 10^{-7}$ and

Table I
RESULTS FOR SINGLE SHOOTING METHOD

F	$C_A(t_f)$	$C_B(t_f)$	$C_D(t_f)$	$\sum_{i=A,B,D} C_i(t_f)$
8.00	50.66	41.60	11.87	104.13
6.93	49.35	41.21	11.64	102.20
6.00	48.23	40.88	11.45	100.56
5.00	47.01	40.51	11.24	98.76
4.00	45.80	40.15	11.03	96.98
0.00	40.93	38.66	10.21	89.80

$fin-diff-grads$ as Hessian update method for SQP active set method optimization algorithm.

In the Table I, where the solutions are given, we observe that for the problem stated in paper [9], the solution is $F = 0$.

When we would like to compare our results with results in [9], we have to change the objective function

$$max_F J = C_D(t_f),$$

subject to the new constraints, which make in this situation some difficulties.

Now we want to present our results and some steps, which the algorithm made. We performed the study in this way. We divided the time domain in 2, 4 and 10 parts. It means, that we have to do 2, 4 and 10 shots, respectively. In every part we can use one constant control function. Then we solved the same problem with only 2 shots, but there were 2 and 5 piecewise control functions in each time interval. Thus we considered 5 ways and in this situation we can compare methods with less number of shots, but the same number of control functions, 4 and 10 , respectively. In examples we divided time domain into equal parts.

A. 2 shots and 1 control function in each interval

We have two control functions u_0 and u_1 for first and second time interval, respectively. Because the control functions are constant, at the beginning of each interval they have to satisfy the constraints

$$0 \leq u_0 \leq 8.5$$

and

$$0 \leq u_1 \leq 8.5.$$

State variables at the beginning of the second interval should satisfy the inequalities

$$0 \leq C_A(0.5t_f) \leq 100,$$

$$0 \leq C_B(0.5t_f) \leq 70,$$

$$0 \leq C_D(0.5t_f) \leq 20.$$

The above constraints are not very restrictive. They are useful for shooting method, because now we can look for solutions in a reasonable range. Next constraint is more restrictive

$$C_A(0.5t_f) + C_B(0.5t_f) + C_D(0.5t_f) \leq 102.2314.$$

We have to converse this problem to the c-problem. The first step is to form vectors, which would be able to describe the parts of the problem. So, this vector, which represents results of one part, can have a form
$w_i =$ (initial values of state variables, control variable).

We know, that each interval has one control variable. Then we can write

$$w_0 = (C_A(0), C_B(0), C_D(0), u_0),$$

$$w_1 = (C_A(0.5t_f), C_B(0.5t_f), C_D(0.5t_f), u_1).$$

Because we want to keep state constraints, we have to check, if the state variable are near from bounds. If they are, we use the procedure described in paper. As the result we have

$$\breve{w} = \begin{pmatrix} 100.0000 & 0 & 0 & 8.5000 \\ 59.0719 & 38.8453 & 5.2914 & 8.5000 \end{pmatrix}$$

Now we have to introduce defect functions. This is the main idea. From vector w_0 we know, that we start from the point $(C_A(0), C_B(0), C_D(0))$ together with control function u_0. At the end of the first interval the process finishes with some results. Then the optimization algorithm should choose the variables $(C_A(0.5t_f), C_B(0.5t_f), C_D(0.5t_f))$, that the differences between results from the first interval and the initial points for second interval are minimized. Together with the final condition we have 4 constraints. Now we can start the optimization process with 5 variables: 3 discretized state variables $(C_A(0.5t_f), C_B(0.5t_f), C_D(0.5t_f))$, and 2 control variables u_0 and u_1. Because this example was quite easy, we can give a Table II with all iterations ($c = 1$).

Table II
SUCCESSIVE ITERATIONS FOR PROBLEM WITH 2 SHOTS AND 1 CONTROL FUNCTION IN EACH INTERVAL

iter	u_0	$C_{A_{0.5t_f}}$	$C_{B_{0.5t_f}}$	$C_{D_{0.5t_f}}$	u_1	c-prob
0	8.5000	59.0719	38.8453	5.2914	8.5000	8.2166
1	8.5000	58.1862	37.9737	4.7914	8.0594	7.6820
2	6.8264	57.3532	37.1814	5.0910	6.7671	9.2803
3	5.9685	57.1764	37.0381	6.3927	6.2319	9.8480
4	5.7684	57.3315	37.2011	6.4248	6.1903	9.8829
5	5.6935	57.2165	37.7093	6.3056	6.2560	10.9499
6	5.9701	57.2828	38.1142	6.2759	6.3548	11.0845
7	6.1807	57.2784	38.4279	6.1808	6.4828	11.3588
8	6.1790	57.2763	38.4529	6.1687	6.4910	11.3766
9	6.2562	57.3946	38.4548	6.1249	6.5671	11.4592
10	6.2815	57.4468	38.4576	6.0859	6.6125	11.4756
11	6.2819	57.4646	38.4571	6.0607	6.6228	11.4872
12	6.3616	57.5148	38.4753	5.9604	6.6138	11.5295
13	6.5238	57.6393	38.5097	5.7744	6.6390	11.5597
14	7.0229	58.0028	38.6019	5.2676	6.7087	11.6343
15	7.0690	58.0363	38.6104	5.2196	6.7166	11.6414
16	7.0697	58.0367	38.6105	5.2182	6.7175	11.6416
17	7.0698	58.0368	38.6105	5.2182	6.7175	11.6416
18	7.0698	58.0368	38.6105	5.2181	6.7176	11.6416
19	7.0698	58.0368	38.6105	5.2181	6.7176	11.6416
20	7.0698	58.0368	38.6105	5.2181	6.7176	11.6416
21	7.0698	58.0368	38.6105	5.2181	6.7176	11.6416
22	7.0698	58.0368	38.6105	5.2181	6.7176	11.6416
23	7.0698	58.0368	38.6105	5.2181	6.7176	11.6416

The solution we can see on the Figure 1.

When c is too small, for example $c = 0.1$, then we can have an unuseful solution (Figure 2).

Figure 1. Problem with 2 shots and 1 control function in each interval.

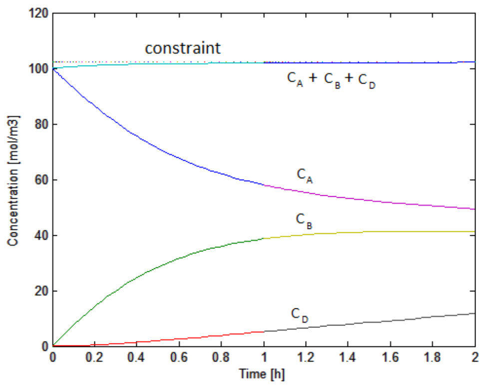
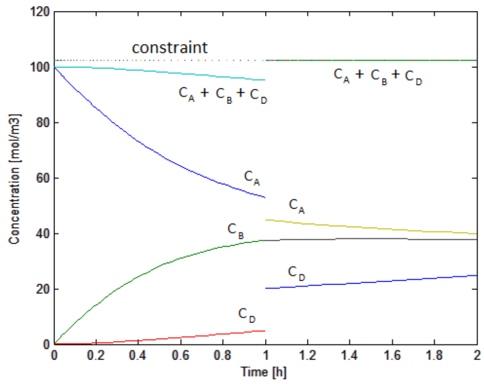


Figure 2. Problem with 2 shots and 1 control function in each interval, when c is too small.



B. 4 shoots and 1 control function in each interval

We start to construct vectors w_i

$$w_0 = (C_A(0), C_B(0), C_D(0), u_0),$$
$$w_1 = (C_A(0.25t_f), C_B(0.25t_f), C_D(0.25t_f), u_1),$$
$$w_2 = (C_A(0.5t_f), C_B(0.5t_f), C_D(0.5t_f), u_2),$$
$$w_3 = (C_A(0.75t_f), C_B(0.75t_f), C_D(0.75t_f), u_3).$$

Now we have 9 discretized state variables and 4 control functions. Together there are 13 variables with 3 constraints

$$C_A(0.25t_f), C_B(0.25t_f), C_D(0.25t_f) \leq 102.2314,$$
$$C_A(0.5t_f), C_B(0.5t_f), C_D(0.5t_f) \leq 102.2314,$$
$$C_A(0.75t_f), C_B(0.75t_f), C_D(0.75t_f) \leq 102.2314,$$

and 10 defect functions: 9 for discretized state variables and one defect function for final state.

Then the algorithm has to formulate the c-problem with a startpoint matrix

$$\breve{w} = \begin{pmatrix} 100.000 & 0 & 0 & 8.5000 \\ 71.8132 & 28.5364 & 1.9502 & 8.5000 \\ 59.0719 & 38.8453 & 5.2914 & 8.5000 \\ 53.8719 & 41.4803 & 8.6989 & 8.5000 \end{pmatrix}$$

Table III
RESULTS FOR PROBLEM WITH 4 SHOOTS AND 1 CONTROL FUNCTION IN EACH INTERVAL

time	state variables C_A	C_B	C_D	control function
$[0, 0.25t_f)$	100	0	0	7.1969
$[0.25t_f, 0.5t_f)$	71.2741	28.4493	1.9381	7.7283
$[0.5t_f, 0.75t_f)$	58.3454	38.6533	5.2325	6.6451
$[0.75t_f, t_f)$	52.5205	41.1556	8.5552	6.2900

Figure 3. Problem with 4 shots and 1 control function in each interval.

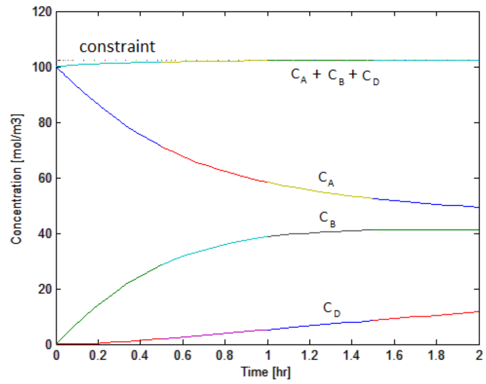


Table IV
PROBLEM WITH 10 SHOTS AND 1 CONTROL FUNCTION IN EACH INTERVAL

time	state variables C_A	C_B	C_D	control function
$[0, 0.1t_f)$	100	0	0	7.8360
$[0.1t_f, 0.2t_f)$	86.5555	14.2030	0.4044	7.7366
$[0.2t_f, 0.3t_f)$	75.7938	24.6084	1.3562	7.6587
$[0.3t_f, 0.4t_f)$	67.8853	31.6167	2.5723	7.3356
$[0.4t_f, 0.5t_f)$	62.2766	36.0457	3.8956	6.8303
$[0.5t_f, 0.6t_f)$	58.2879	38.6949	5.2485	6.7313
$[0.6t_f, 0.7t_f)$	55.4486	40.1879	6.5948	6.6217
$[0.7t_f, 0.8t_f)$	53.3526	40.9592	7.9195	6.4818
$[0.8t_f, 0.9t_f)$	51.7257	41.2896	9.2160	6.3075
$[0.9t_f, t_f)$	50.3911	41.3540	10.4855	6.1464

As the results we have values of control variables and discretized state variables (Table III).

The results we can see on the Figure 3.

C. 10 shots and 1 control function in each interval

This problem is bigger than the previous. We have 37 variables: $3 \cdot 9$ discretized state variables and 10 control variables. Because we have 27 discretized state variables, at the moment we need 27 defect functions. Together with one defect function for final state, we have to consider problem with 28 defect functions.

As the results we have values of control variables and discretized state variables (Table IV). There are results on the Figure 4.

D. 2 shots and 2 control functions in each interval

We considered this situation, because it is similar to problem with 1 shot and 4 control functions. In both situations control functions contribute 4 degrees of freedom. Now each control

Figure 4. Problem with 10 shots and 1 control function in each interval.

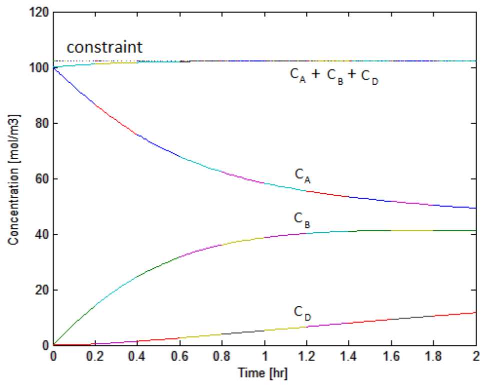
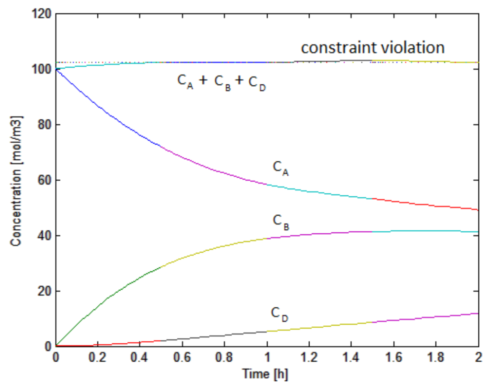


Figure 5. Problem with 2 shots and 2 control functions in each interval.



function consists of two steps, so vector, which represents a solution of each part can have a form
w_i = (initial values of state variables, control functions). In particular, there are

$$w_0 = (C_A(0), C_B(0), C_D(0), u_{01}, u_{02}),$$

and

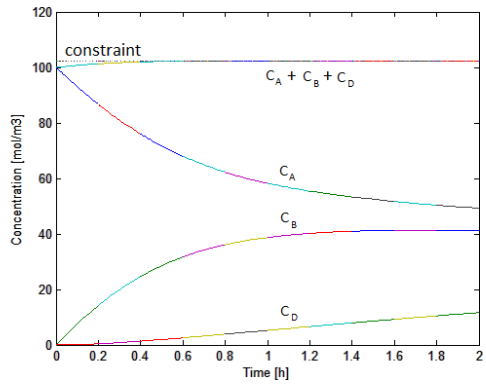
$$w_1 = (C_A(0.5t_f), C_B(0.5t_f), C_D(0.5t_f), u_{11}, u_{12}).$$

Together there are 3 discretized control variables and 2 control functions. Each control function has two parameters. In all there are 7 variables and 4 defect functions: 3 for discretized state variables and one for the final state.

Now we can repeat the Betts' question: What Can Go Wrong? [3]. The algorithm can influence the selection of discretized state variables and control variables. But we not can be sure, that in each interval, when controls variables are changing, the constraints are satisfied. The obtained solution, which is feasible for c-problem, can violate the constraints of basic discretized problem (Figure 5). Then it should be regularized.

An important topic. We have less variables in this problem, but we lose a possibility of parallel computations too.

Figure 6. Problem with 2 shots and 5 control function in each interval.



E. 2 shots and 5 control functions in each interval

In this problem we have 3 discretized state variables too. But we have to consider situation with 10 control functions and 4 defect functions: 3 for discretized state variables and one for the final state. We can have the same problem like in the previous example (Figure 6).

Methods, which have more than 1 control variable in each time interval give good results for BVP's.

F. Comparison of results

The results are summarized in Table V.

We can see, that methods, which have the same number of time intervals as number of control functions give results more similar to the results of the basic problem.

All experiments were performed on the processor Intel(R) Core(TM) i5 CPU 2.67 GHz.

Important questions are the CPU performance time and a parallel computing possiblity. Especially, when shooting method is used, one is able to use more processors. Although m processors were used, the computations performance time would not be m times smaller. Before starting the parallel computing one has to have in mind a communications overhead. Running 2, 3 or 4 local workers as the clients on the same machine needs about 0.35 CPU time.

When can we expect a performance time improvement? The performance time depends on number of functions evaluations and communications time with local workers. If we denote x as CPU time processing on one processor, y - number of function evaluations and n - number of local workers, one can compute number of local workers are needed

$$\frac{1}{n} \cdot x + 0.35 \cdot y \leq x.$$

If $0.35 \cdot y \leq x$, then

$$\frac{x}{x - 0.35 \cdot y} \leq n.$$

In the presented example parallel computing should not improve the performance time. It would be useful in more complex applications.

Table V
COMPARISON OF RESULTS

Problem		Results			
shots	control functions	CPU time	fun eval	c-prob	basic prob
2	1	14.2429	303	11.6416	11.6536
4	1	80.2313	1623	11.6998	11.6998
2	2	23.7434	522	11.7815	11.6867
10	1	701.6457	8148	11.7160	11.7160
2	5	104.8015	1391	11.8278	11.6885

V. CONCLUSION

Modified multiple shooting algorithms for the optimization of complex DAE systems are proposed. They are aimed at the determination of a suboptimal feasible solution of a high practical meaning. To this end an initial feasible solution is found by the c-conversion of the basic discretized multipoint problem and the analysis of the results of the consecutive shots of the system trajectory. The employment of the feasible-SQP approach dealing with compatible QP subproblems is quaranteed. The optimized solution can be regularized by the bound-constrained trust-region and inexact Newton method to ensure a high degree of its applicability.

REFERENCES

[1] S. Bellavia and B. Morini, "Subspace trust-region methods for large bound-constrained nonlinear equations," *SIAM Journal on Numerical Analysis*, vol.44., pp.1535-1555, 2006.

[2] S. Bellavia, M. Macconi, and B. Morini, "An affine scaling trust-region approach to bound-constrained nonlinear systems," *Applied Numerical Mathematics*, vol. 44, pp. 257-280, 2003.

[3] J.T. Betts, *Practical Methods for Optimal Control and Estimation Using Nonlinear Programming*. Philadelphia: SIAM, 2010.

[4] L.T. Biegler, *Nonlinear Programming. Concepts, Algorithms, and Chemical Processes*. Philadelphia: SIAM, 2010.

[5] K.E. Brenan, S.L. Campbell, and L.R. Petzold, *Numerical Solution of Initial-Value Problems in Differential-Algebraic Equations*. Philadelphia: SIAM, 1996.

[6] L.Chen, Y. Wang, and G. He, "A feasible active set QP-free method for nonlinear programming," *SIAM Journal on Optimization*, vol.17, pp. 401-429, 2006.

[7] E. Gelman and J. Mandel, "On multilevel iterative methods for optimization problems," *Mathematical Programming*, vol. 48, pp. 1-17, 1990.

[8] J. Herskovits, "A two-stage feasible directions algorithm for nonlinear constrained optimization," *Mathematical Programming*, vol.36, pp.19-38, 1986.

[9] Y. J. Huang, G. V. Reklaitis, V. Venkatasubramanian, "Model decomposition based method for solving general dynamic optimization problems," *Computers and Chemical Engineering*, vol. 26, pp. 863-873, 2002

[10] J.-B. Jian, C-M. Tang, Q.-J. Hu, H.-Y. Zheng, "A feasible descent SQP algorithm for general constrained optimization without strict complementarity," *Journal of Computational and Applied Mathematics*, vol. 180, pp. 391-412, 2005. pp.1531-1542, 1986.

[11] D.B. Leineweber, I. Bauer, H.G. Bock, J.P. Schlöder, "An efficient multiple shooting based reduced SQP strategy for large-scale dynamic process optimization. Part 1: theoretical aspect," *Computers and Chemical Engineering*, vol. 27, pp. 157-166, 2003.

[12] D.Q. Mayne and E. Polak, "Feasible directions algorithms for optimization problems with equality and inequality constraints," *Mathematical Programming*, vol.11, pp.67-80, 1976.

[13] R. Pytlak, "Optimal control of differential-algebraic equations of higher index, Part 1: First-order approximations," *J. Optimiz. Theory Appl.*, vol. 134, pp. 61-75, 2007.

[14] M.J. Śmietański, "Inexact quasi-Newton global convergent method for solving constrained nonsmooth equations," *International Journal of Computer Mathematics*, vol. 84, pp.1757-1770, 2007.

[15] Z. Zhu, W. Zhang, and Z. Geng, "A feasible SQP method for nonlinear programming," *Applied Mathematics and Computation*, vol. 215, pp.3956-3969, 2010