Journal Code:				Article ID				Dispatch: 12.03.12	CE: CMLara
O	C	A		2	0	2	6	No. of Pages: 19	ME:

OPTIMAL CONTROL APPLICATIONS AND METHODS
Optim. Control Appl. Meth. (2012)
Published online in Wiley Online Library (wileyonlinelibrary.com). DOI: 10.1002/oca.2026

A quadratically convergent inexact SQP method for optimal control of differential algebraic equations

Boris Houska[1,2,*,†] and Moritz Diehl[1,2]

[1]*Electrical Engineering Department (ESAT), K.U. Leuven, Kasteelpark Arenberg 10, 3001 Leuven, Belgium*
[2]*Optimization in Engineering Center (OPTEC), K.U. Leuven, Kasteelpark Arenberg 10, 3001 Leuven, Belgium*

SUMMARY

In this paper, we present an inexact sequential quadratic programming method in the context of a direct multiple shooting approach for differential algebraic equations. For the case that a numerical integration routine is used to compute the states of a relaxed differential algebraic equation, the computation of sensitivities, with respect to a large number of algebraic states, can become very expensive. To overcome this limitation, the inexact sequential quadratic programming method that we propose in this paper requires neither the computation of any sensitivity direction of the differential state trajectory, with respect to the algebraic states as [10], nor the consistent initialization of the differential algebraic equation, as [7, 8]. We prove the locally quadratic convergence of the proposed method. Finally, we demonstrate the numerical performance of the method by optimizing a distillation column with 82 differential and 122 algebraic states. Copyright © 2012 John Wiley & Sons, Ltd.

Received 2 December 2009; Revised 3 November 2010; Accepted 10 February 2012

KEY WORDS: nonlinear optimal control; inexact SQP methods; differential algebraic equations; DAE relaxation

1. INTRODUCTION

For the numerical solution of nonlinear optimal control problems with an underlying differential algebraic equation (DAE), typically, nonlinear programming (NLP) methods, also called direct approaches, are applied. Here, the DAE is discretized first, and the optimal control problem is transformed into a finite dimensional NLP. Many researchers have developed collocation methods to discretize the dynamic model [1, 2]. This allows the discretization of the DAE exclusively at the level of the NLP but leads to extremely large and sparse optimization problems, whereas the collocation scheme must be adapted to control the discretization error. Alternatively, a single shooting approach, as introduced in [3, 4], or a multiple shooting approach, as introduced in [5, 6], can be applied to discretize the problem on a moderate number of coarse intervals, making use of an integration routine which adaptively discretizes the DAE in an accurate way within these coarse intervals.

In this paper, we concentrate on a particular aspect of direct multiple shooting methods for DAE: as the differential algebraic equation needs to be simulated subsequently during the iterations of the NLP algorithm, it is not efficient to compute consistent initializations in every step of the optimization algorithm. In the simplest form of the multiple shooting methods for DAE [7, 8], this aspect has simply been neglected, leading to more expensive DAE integration phases, where in every step of the

*Correspondence to: Boris Houska, Electrical Engineering Department, K.U. Leuven, Kasteelpark Arenberg 10 (bus 2446), 3001 Leuven, Belgium.
†E-mail: boris.houska@esat.kuleuven.be

Copyright © 2012 John Wiley & Sons, Ltd.

UNCORRECTED PROOF

optimization algorithm and in every multiple shooting node, the set of nonlinear consistency conditions must be solved by Newton's method. However, in [9] and later in [10], this problem has been overcome by the introduction of DAE relaxation, which allows the satisfaction of the consistency conditions only in the optimal solution. This relaxation function weakens the algebraic consistency conditions by introducing slack parameters in such a way that the DAE is always consistent. Actually, it turns out that the sensitivities, with respect to these slack parameters, are only needed in certain directions for the case that partially reduced sequential quadratic programming (PRSQP) methods are used at the top-level of the optimization, as introduced by Leineweber [10]. However, these methods suffer from the fact that the implementation of the PRSQP strategy, and the sensitivity generation of the DAE are deeply intertwined, which makes the implementation complicated.

The main contribution of this paper is divided into two parts. First, a special parameterized relaxation function for the DAE is suggested. This special relaxation function is chosen in such a way that the sensitivity directions of the state trajectory, with respect to the relaxation parameters, vanish in the optimal solution. Second, an inexact SQP method is proposed, which uses this property of the new relaxation function in a systematic way. This inexact SQP method has an interesting property, wherein the approximations of the Hessian and Jacobian matrices become exact within the optimal solution. As discussed in [11, 12], general purpose inexact SQP methods can achieve locally q-superlinear convergence for the case that a suitable update method for the matrix approximations is applied. Although the specialized SQP method proposed in this paper is also inexact, that is the Hessian and the constraint Jacobian in the QPs are only approximated, the q-quadratic convergence properties can be recovered.

In Section 2, we review the direct multiple shooting discretization approach for DAE optimal control problems and introduce the basic notation, as well as the concept of relaxation functions that is needed within the paper. In the next step, we concentrate on a special class of relaxation functions for DAE systems, which are analyzed in Section 3 and for which we can show desirable properties. These properties are used in Section 4, where the inexact SQP method is constructed. Within this section, we discuss the q-quadratic convergence properties of the method. In Section 5, the new approach is applied and successfully tested with both a small-scale toy problem and a large-scale real-world DAE optimization problem arising in the context of optimal control of continuous distillation processes. The latter model, which was validated at a real-world distillation column [13], includes 122 implicit algebraic as well as 82 differential states. We conclude the paper in Section 6.

2. DISCRETIZATION OF DAE OPTIMIZATION PROBLEMS

In this section, we introduce the following standard formulation of the equality constrained DAE optimization problems

$$\begin{aligned} &\underset{x(\cdot),z(\cdot),u(\cdot),p}{\text{minimize}} && J[x(\cdot),z(\cdot),u(\cdot),p] \\ &\text{subject to:} \\ &\forall t \in [0,m]: && \dot{x}(t) = f(t,x(t),z(t),u(t),p) \\ &\forall t \in [0,m]: && 0 = g(t,x(t),z(t),u(t),p) \\ & && 0 = r(x(0),x(T),p) \end{aligned} \tag{1}$$

Here, $x : [0,m] \to \mathbb{R}^{n_x}$ and $z : [0,m] \to \mathbb{R}^{n_z}$ denote differential and algebraic states, respectively, whereas $u : [0,m] \to \mathbb{R}^{n_u}$ is a time-dependent control, and $p \in \mathbb{R}^{n_p}$ a time constant parameter. For ease of notation, we assume that m is a given integer. Note that this can always be achieved by rescaling the time horizon if necessary. The right-hand side functions f, g, and r are assumed to be twice continuously differentiable in all arguments. The function g should additionally satisfy the condition that $\frac{\partial g}{\partial z}$ is regular, that is, the DAE is assumed to have the index 1. Moreover, J denotes an objective functional. Note that, for notational simplicity, the mentioned DAE optimal control problem (1) does not take any inequality constraints into account. However, the following considerations can easily be generalized for the case that we have additional inequalities by using SQP methods [14, 15].

UNCORRECTED PROOF

Copyright © 2012 John Wiley & Sons, Ltd.

DOI: 10.1002/oca

In this paper, we are interested in the case where the mentioned DAE optimization problem is discretized by Bock's direct multiple shooting approach [5]. This means that we discretize our control input u on the finite mesh $0 = t_0 < t_1 < \ldots < t_m = m$ with $t_i = i$ for all $i \in \{0, ..., m\}$, writing the piecewise constant control as $\forall t \in [i, i+1]: \; u(t) := u_i \in \mathbb{R}^{n_u}$ with $i \in \{0, \ldots, m-1\}$. The main idea of multiple shooting is to take not only the discretized control input but also the initial values $s_0 := x(0), \ldots, s_m := x(m)$ at the nodes into the formulation of the discrete NLP—in contrast to single shooting, where only s_0 is regarded as a free variable. More precisely, we define the functions $\hat{X}_0, \hat{X}_1, \ldots, \hat{X}_{m-1} : \mathbb{R}^{n_x} \times \mathbb{R}^{n_u} \times \mathbb{R}^{n_p} \to \mathbb{R}^{n_x}$ to be solutions $\hat{X}_i(s_i, u_i, p) := \hat{x}_i(i+1)$ of the given DAE on the multiple shooting intervals ($i \in \{0, \ldots, m-1\}$)

$$\begin{aligned} \frac{\mathrm{d}}{\mathrm{d}t}\hat{x}_i(t) &= f(t, \hat{x}_i(t), \hat{z}_i(t), u_i, p) \\ 0 &= g(t, \hat{x}_i(t), \hat{z}_i(t), u_i, p) \\ \text{with} \quad \hat{x}_i(0) &= s_i \end{aligned} \tag{2}$$

for all $t \in [i, i+1]$. Note that the functions $\hat{X}_i$ are well defined in their domains, in the sense that the solution of the DAE (2) uniquely exists as we assume that f and g are twice continuously differentiable and $\frac{\partial g}{\partial z}$ regular.[‡]

As the optimal solution for the state x should be continuous, we have to require matching conditions to be satisfied at the multiple shooting nodes. The initial value s_{i+1} associated with the i-th interval should, in the optimal solution, be equal to the end value $\hat{X}_{i-1}(s_{i-1}, u_{i-1}, p)$ of the previous interval. Thus, the matching conditions can be summarized in the form

$$\hat{H}(\alpha) := \begin{pmatrix} s_1 - \hat{X}_0(s_0, u_0, p) \\ s_2 - \hat{X}_1(s_1, u_1, p) \\ \vdots \\ s_m - \hat{X}_{m-1}(s_{m-1}, u_{m-1}, p) \end{pmatrix} = 0 \tag{3}$$

Here, we collect the $(m+1)$ initial values at the nodes, the m control input pieces, as well as the free parameters in the variable

$$\alpha := \left(s_0^T, s_1^T, \ldots, s_m^T, u_0^T, u_1^T, \ldots u_{m-1}^T, p^T\right)^T \in \mathbb{R}^{n_\alpha}$$

where the dimension n_α is given by $n_\alpha := (m+1)n_x + mn_u + n_p$.

In the next step, the objective functional and the boundary constraints must be discretized. For this aim, we introduce the algebraic node values

$$\beta := \left(\beta_0^T, \beta_1^T, \ldots, \beta_m^T\right)^T := \left(z(0)^T, z(1)^T, \ldots, z(m)^T\right)^T$$

These algebraic node values allow us to finally write the discretized version of the optimal control problem (1) in the form

$$\begin{aligned} \underset{\alpha,\beta}{\text{minimize}} \quad & F(\alpha, \beta) \\ \text{subject to:} \quad & G(\alpha, \beta) = 0 \\ & \hat{H}(\alpha) \quad = 0 \end{aligned} \tag{4}$$

Here, the algebraic consistency conditions and the discretized boundary constraint have been summarized in the function $G : \mathbb{R}^{n_\alpha} \times \mathbb{R}^{n_\beta} \to \mathbb{R}^{(m+1)n_z + n_r}$, which is defined as

$$G(\alpha, \beta) := \begin{pmatrix} g(0, s_0, \beta_0, u_0, p) \\ g(1, s_1, \beta_1, u_1, p) \\ \vdots \\ g(m-1, s_{m-1}, \beta_{m-1}, u_{m-1}, p) \\ g(m, s_m, \beta_m, u_{m-1}, p) \\ r(s_0, s_m, p) \end{pmatrix} \tag{5}$$

[‡]For a proof of this uniqueness and existence statement, we refer to [16].

UNCORRECTED PROOF

Copyright © 2012 John Wiley & Sons, Ltd.

DOI: 10.1002/oca

Finally, the function $F : \mathbb{R}^{n_\alpha} \times \mathbb{R}^{n_\beta} \to \mathbb{R}$ represents the discrete version of the objective functional J evaluated at the multiple shooting points. In the following, we assume that F is twice continuously differentiable in all its arguments.

Once the discrete optimization problem (4) is derived, we can, of course, use a standard NLP solver (e.g. an SQP method) to solve this structured nonlinear program. As it was mentioned in Section 1, this strategy has been applied in some approaches of multiple shooting for DAE [7, 8]. However, note that the evaluation of the function $\hat{H}$—or, more precisely, the evaluation of functions $\hat{X}_0, \ldots, \hat{X}_{m-1}$—is typically the most expensive part of the algorithm, as it requires solving m DAEs of the form (2) in each step of the SQP algorithm. Numerical integration routines, which are able to numerically solve these DAE systems (2), usually proceed in two phases. In the first phase, a consistent algebraic initialization point $\beta_i^* \in \mathbb{R}^{n_z}$ is generated which satisfies $\|g(i, s_i, \beta_i^*, u_i, p)\| \leqslant \epsilon$, where $\epsilon \geqslant 0$ is a small constant, depending on how accurate we want to solve the DAE. This first phase can, for example, be performed through a Newton method, as the Jacobian $\frac{\partial g}{\partial z}$ is assumed to be regular. In the second phase, the integration algorithm is started, which can, for example, be based on an implicit Runge–Kutta method [17] or on backward differentiation formula (BDF) methods [18, 19].

However, if we regard optimization methods for DAE, it is usually not efficient to compute a consistent initialization in every step of the optimization algorithm. The key idea to avoid in the first phase is to relax the DAE with the aim to simulate the differential algebraic equation on the basis of a nonconsistent initial value. This strategy has originally been developed in [9] and was refined in [10]. Here, relaxation means that the original algebraic condition in (2) is (for all $i \in \{1, \ldots, m-1\}$) replaced by a modified equation of the form

$$0 = g(t, x_i(t), z_i(t), u_i, p) - \vartheta\,(\gamma_i, t - i) \tag{6}$$

for all $t \in [i, i+1]$, where $\vartheta : \mathbb{R}^{n_z} \times [0,1] \to \mathbb{R}^{n_z}$ is a relaxation function and

$$\gamma_i := g(i, s_i, \beta_i, u_i, p)$$

This relaxation function is required to satisfy the conditions

$$\forall \gamma \in \mathbb{R}^{n_z} : \quad \vartheta(\gamma, 0) = \gamma \tag{7}$$

$$\forall \tau \in [0,1] : \quad \vartheta(0, \tau) = 0 \tag{8}$$

such that (6) is, by construction, satisfied at $t = i$. In addition, if the algebraic consistency condition $\gamma_i = g(i, s_i, \beta_i, u_i, p) = 0$ holds at the i-th shooting node, the condition (8) guarantees that the function ϑ vanishes on the corresponding shooting interval, such that the relaxed condition (6) coincides with the original algebraic condition.

In [9], the function

$$\vartheta_1(\gamma, \tau) := \gamma \quad \text{for all } (\gamma, \tau) \in \mathbb{R}^{n_z} \times [0,1] \tag{9}$$

was chosen as a relaxation function. Whereas in [18] and in [10], it is reported that the function

$$\vartheta_2(\gamma, \tau) := \gamma\, \exp(-\delta\tau) \quad \text{for all } (\gamma, \tau) \in \mathbb{R}^{n_z} \times [0,1] \tag{10}$$

with the empirical value $\delta = 5$, works better in practice. One of the two main contributions of this paper is suggesting a third choice for the function ϑ, which will be defined in (16). In Sections 3 and 4, we will discuss the advantages and desirable properties of this new relaxation function in comparison to the existing choices (9) and (10). However, let us first address what the introduction of a relaxation function changes with regard to the discretization of the continuous optimal control problem. For this aim, we consider the relaxed DAEs on the multiple shooting intervals ($i \in \{0, \ldots, m-1\}$)

$$\begin{aligned} \frac{\mathrm{d}}{\mathrm{d}t} x_i(t) &= f(t, x_i(t), z_i(t), u_i, p) \\ 0 &= g(t, x_i(t), z_i(t), u_i, p) - \vartheta(\gamma_i, t - i) \\ \text{with} \quad x_i(0) &= s_i \end{aligned} \tag{11}$$

UNCORRECTED PROOF

Copyright © 2012 John Wiley & Sons, Ltd.

DOI: 10.1002/oca

for all $t \in [i, i+1]$. Analogous to the functions $\hat{X}_0, \hat{X}_1, \ldots, \hat{X}_{m-1}$, we define the functions $X_0, X_1, \ldots, X_{m-1} : \mathbb{R}^{n_x} \times \mathbb{R}^{n_z} \times \mathbb{R}^{n_u} \times \mathbb{R}^{n_p} \to \mathbb{R}^{n_x}$ to be the solutions

$$X_i(s_i, \gamma_i, u_i, p) := x_i(i+1)$$

of the relaxed DAE system (11). Moreover, we define a function H by

$$H(\alpha, \Gamma) := \begin{pmatrix} s_1 - X_0(s_0, \gamma_0, u_0, p) \\ s_2 - X_1(s_1, \gamma_1, u_1, p) \\ \vdots \\ s_m - X_{m-1}(s_{m-1}, \gamma_{m-1}, u_{m-1}, p) \end{pmatrix} \tag{12}$$

Here, we us the definition

$$\Gamma := \left(\gamma_0^T, \gamma_1^T, \ldots, \gamma_{m-1}^T, \gamma_m^T, \gamma_r^T\right)^T := G(\alpha, \beta) \tag{13}$$

as a notation for the components of the function G. Now, the matching conditions for the relaxed DAE can be summarized as

$$H(\alpha, G(\alpha, \beta)) = 0 \tag{14}$$

The associated discretized optimization problem takes the form

$$\begin{aligned} \underset{\alpha,\beta}{\text{minimize}} \quad & F(\alpha, \beta) \\ \text{subject to:} \quad & 0 = G(\alpha, \beta) \\ & 0 = H(\alpha, G(\alpha, \beta)) \end{aligned} \tag{15}$$

Note that the optimization problems (4) and (15) are equivalent by construction:

Lemma 2.1
If the relaxation function ϑ satisfies $\vartheta(0, \tau) = 0$ for all $\tau \in [0, 1]$, the optimization problems (4) and (15) are equivalent. That is, a point (α^*, β^*) is an optimal point of the problem (4) if, and only if, it is an optimal point of the problem (15).

Proof
If (α^*, β^*) is a feasible point of one of the optimization problems, it must satisfy $G(\alpha^*, \beta^*) = 0$; that is, the consistency condition $\gamma_i^* = g(t_i, s_i^*, \beta_i^*, u_i^*, p^*) = 0$ holds for all $i \in \{1, \ldots, m-1\}$. Thus, the relaxation function ϑ satisfies $\vartheta(\gamma_i^*, \tau) = \vartheta(0, \tau) = 0$ for all $\tau \in [0, 1]$, that is the relaxation vanishes at an optimal point such that the relaxed and the original DAE exactly coincide. Consequently, we have $\hat{H}(\cdot) \equiv H(\cdot, G(\alpha^*, \beta^*)) \equiv H(\cdot, 0)$ and the optimization problems (4) and (15) must be equivalent. □

The advantage of the function H in comparison to the function $\hat{H}$ is that a single evaluation is cheaper, as the simulation of the relaxed DAE (11) does not require a first phase in which a consistent initial value is computed. However, the function H does not only depend on α but implicitly also on the variable β which enters via the variable $\Gamma = G(\alpha, \beta)$. Thus, assuming that H is differentiable, the computation of the Jacobian of H with respect to Γ can be expensive—especially if we have many algebraic states. In order to overcome this problem, in [10], the use of PRSQP methods has been proposed to solve the relaxed problem (15). These methods have the advantage of only needing the $n_\alpha + 1$ directional derivatives of H per SQP step—in comparison to needing the $n_\alpha + n_\beta$ directional derivatives of H for the whole Jacobian which is needed within the full-space SQP method. However, PRSQP methods suffer from the fact that the simulation and derivative generation of the DAE system and the optimization routine itself are deeply intertwined [10], which makes the implementation complex and less attractive from a programming point of view. Q5

The first contribution of this paper is the introduction of a novel relaxation function ϑ defined by (16). The desirable properties (cf. Section 3) of this special relaxation function will allow us to

UNCORRECTED PROOF

Copyright © 2012 John Wiley & Sons, Ltd.

DOI: 10.1002/oca

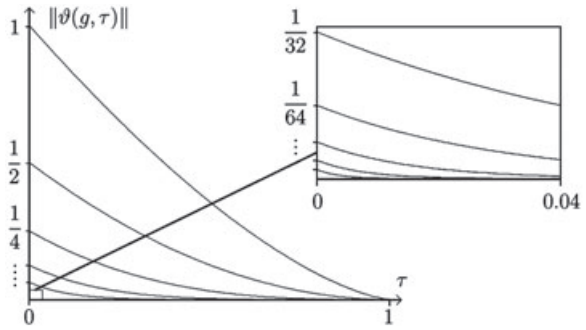


Figure 1. The function $\|\vartheta(\gamma,\tau)\|$ depending on τ for several values of $\|\gamma\|$ (with the values $\|\gamma\| \in \{1, \frac{1}{2}, \frac{1}{4}, \ldots, \frac{1}{256}\}$) for cases $a = \frac{1}{2}$ and $b = 1$.

avoid the computation of the derivatives of H with respect to β. That is, we only need n_α derivatives of H per SQP step, whereas the optimization and DAE simulation, together with the derivative generation, can be implemented in a modular way. In the following sections, we will explain this idea step by step.

3. PROPERTIES OF THE NEW RELAXATION FUNCTION

In this section, we are interested in the analysis of the relaxed DAE (11) under the assumption that the relaxation function $\vartheta : \mathbb{R}^{n_z} \times [0,1] \to \mathbb{R}^{n_z}$ has the form

$$\vartheta(\gamma,\tau) := \begin{cases} \gamma\,(1-\tau)^{\frac{a+b\|\gamma\|}{\|\gamma\|}} & \text{if } \|\gamma\| \neq 0 \\ 0 & \text{otherwise} \end{cases} \tag{16}$$

for all $(\gamma,\tau) \in \mathbb{R}^{n_z} \times [0,1]$. Here, $a > 0$ and $b \geq 1$ are positive design parameters, whereas $\|\cdot\|$ denotes the Euclidean norm.[§] In Figure 1, the norm $\|\vartheta(\gamma,\tau)\|$ is plotted as a function of the variable $\tau \in [0,1]$ for several values of $\|\gamma\|$ by using the choice $a = \frac{1}{2}$ and $b = 1$.

Lemma 3.1
The relaxation function ϑ, which is defined by (16), has the following properties:

(*i*) The function ϑ is continuous on its domain $\mathbb{R}^{n_z} \times [0,1]$ and satisfies the fundamental relations (7) and (8).

(*ii*) The function ϑ is differentiable with respect to τ, and the associated derivative $\dot{\vartheta} := \frac{\mathrm{d}\vartheta}{\mathrm{d}\tau}$ satisfies

$$\forall \tau \in [0,1]: \quad \left\| \dot{\vartheta}(\gamma,\tau) \right\| \leq a + \mathbf{O}(\|\gamma\|) \tag{17}$$

(*iii*) On the domain $(\mathbb{R}^{n_z} \setminus \{0\}) \times [0,1]$, the function ϑ is twice differentiable with respect to γ, and the associated derivatives $\vartheta' := \frac{\mathrm{d}\vartheta}{\mathrm{d}\gamma}$ and $\vartheta'' := \frac{\mathrm{d}^2\vartheta}{\mathrm{d}\gamma^2}$ satisfy

$$\begin{aligned} \int_0^1 \left\| \vartheta'(\gamma,\tau) \right\| \mathrm{d}\tau &\leq \mathbf{O}(\|\gamma\|)\,, \\ \int_0^1 \left\| \vartheta'(\gamma,\tau) \right\|^2 \mathrm{d}\tau &\leq \mathbf{O}(\|\gamma\|)\,, \\ \text{and} \quad \int_0^1 \left\| \vartheta''(\gamma,\tau) \right\| \mathrm{d}\tau &\leq \mathbf{O}(1) \end{aligned} \tag{18}$$

for all $\gamma \in \mathbb{R}^{n_z} \setminus \{0\}$.

[§]In the following, we will use the notation $\|\cdot\|$ not only for the Euclidean norm on $\mathbb{R}^{n_z}$ but also for the induced matrix or tensor norms.

UNCORRECTED PROOF

Copyright © 2012 John Wiley & Sons, Ltd.

DOI: 10.1002/oca

Proof
The statement (*i*) follows immediately from the definition of ϑ. Let us define the exponent κ by

$$\kappa := \frac{a + b\|\gamma\|}{\|\gamma\|} \geqslant b \geqslant 1$$

We can compute the slope of the function ϑ by direct computation

$$\left\| \dot{\vartheta}(\gamma, \tau) \right\| = \left\{ \begin{array}{ll} (a + b\|\gamma\|)(1-\tau)^{\kappa-1} & \text{if } \|\gamma\| \neq 0 \\ 0 & \text{otherwise} \end{array} \right\} \tag{19}$$

Obviously, we have

$$\left\| \dot{\vartheta}(\gamma, \tau) \right\| \leqslant a + b\|\gamma\| = a + \mathbf{O}(\|\gamma\|) \tag{20}$$

for all $\tau \in [0, 1]$, which shows the statement (*ii*). Moreover, the differentiability of ϑ on the domain $(\mathbb{R}^{n_z} \setminus \{0\}) \times [0, 1]$ is a rather trivial consequence of its definition (16). Explicitly, ϑ' can be written as

$$\vartheta'(\gamma, \tau) = \left[1 - \frac{a\gamma\gamma^T \log(1-\tau)}{\|\gamma\|^3} \right] (1-\tau)^{\kappa} \tag{21}$$

where we understand the above right-hand side expression at the point $\tau = 1$ in the limit sense, which is justified because the inequality $\kappa \geqslant 1$ guarantees that the limit

$$\lim_{t \to 1} \; \log(1-\tau)(1-\tau)^{\kappa} = 0 \tag{22}$$

exists. It is a simple exercise to show that, for all integers $n \in \{0, 1, 2\}$, the following relation holds (for all $\kappa \geqslant 1$):

$$\int_0^1 \log(1-\tau)^n (1-\tau)^{\kappa} \, \mathrm{d}\tau = (-1)^n \frac{1}{(1+\kappa)^{n+1}} \tag{23}$$

Now, we use the formula (23) once with $n = 0$ and once with $n = 1$ to find

$$\begin{aligned} \int_0^1 \|\vartheta'(\gamma, \tau)\| \, \mathrm{d}\tau &= \int_0^1 \left[1 - \frac{a}{\|\gamma\|} \log(1-\tau) \right] (1-\tau)^{\kappa} \, \mathrm{d}\tau \\ &= \frac{1}{\kappa+1} + \frac{a}{\|\gamma\|} \frac{1}{(\kappa+1)^2} \leqslant \frac{2}{a}\|\gamma\| \end{aligned} \tag{24}$$

Similarly, we compute the integral

$$\begin{aligned} \int_0^1 \|\vartheta'(\gamma, \tau)\|^2 \, \mathrm{d}\tau &= \int_0^1 \left[1 - \frac{a}{\|\gamma\|} \log(1-\tau) \right]^2 (1-\tau)^{2\kappa} \, \mathrm{d}\tau \\ &= \frac{1}{2\kappa+1} + 2\frac{a}{\|\gamma\|} \frac{1}{(2\kappa+1)^2} + \frac{a^2}{\|\gamma\|^2} \frac{1}{(2\kappa+1)^3} \leqslant \frac{9}{8a}\|\gamma\| \end{aligned} \tag{25}$$

where we have used the formula (23) once for $n = 0$, once for $n = 1$, and once for $n = 2$, as well as $\kappa \geqslant \frac{a}{\|\gamma\|}$. Finally, we compute

$$\begin{aligned} \int_0^1 \|\vartheta''(\gamma, \tau)\| \, \mathrm{d}\tau &\leqslant \int_0^1 \frac{a^2}{\|\gamma\|^3} \log(1-\tau)^2 (1-\tau)^{\kappa} \, \mathrm{d}\tau - \int_0^1 \frac{3a}{\|\gamma\|^2} \log(1-\tau)(1-\tau)^{\kappa} \, \mathrm{d}\tau \\ &= \frac{a^2}{\|\gamma\|^3} \frac{1}{(\kappa+1)^3} + \frac{3a}{\|\gamma\|^2} \frac{1}{(\kappa+1)^2} \leqslant \frac{4}{a} \end{aligned} \tag{26}$$

which leads to the statement (*iii*). □

UNCORRECTED PROOF

Copyright © 2012 John Wiley & Sons, Ltd.

DOI: 10.1002/oca

Remark 3.1
The statement (*ii*) of the above Lemma is important, as the slope of ϑ will influence the performance of the numerical integration algorithm applied to the associated relaxed DAE. Whenever the residual $\|\gamma\|$ is sufficiently small, which means, in our context, that the algebraic consistency condition is approximately satisfied, the estimate (17) guarantees that this slope can directly be influenced by the design parameter a.

Recall that the solutions of the relaxed differential algebraic equations (11) are denoted by $X_i(s_i, \gamma_i, u_i, p)$. In the following, we analyze these functions X_i in more detail under the assumption that the relaxation function ϑ satisfies the properties (*i*) and (*iii*) of the above Lemma.

Theorem 3.1
Let $i \in \{0, ..., m-1\}$, s_i, u_i, and p be given, the functions f and g be twice continuously differentiable, and $\frac{\partial g}{\partial z}$ be regular. If the function ϑ satisfies the properties (*i*) and (*iii*) of Lemma 3.1, then the function X_i, which is defined to be the solution of the DAE system (11), is differentiable with respect to γ_i and its Jacobian $X_i' := \frac{\partial X_i}{\partial \gamma_i}$ is locally Lipschitz continuous. Moreover, we have $X_i'(s_i, 0, u_i, p) = 0$.

Proof
Let us pick $\overline{\gamma} \in \mathbb{R}^{n_z}$. For theoretical purposes, we eliminate the algebraic states of the system (11) in a neighborhood of $\overline{\gamma}$. More precisely, the implicit function theorem guarantees the existence of a bounded open neighborhood $D \subset \mathbb{R}^{n_z}$ with $\overline{\gamma} \in D$ and a twice continuously differentiable function $h : [i, i+1] \times \mathbb{R}^{n_x} \times \mathbb{R}^{n_z} \times \mathbb{R}^{n_u} \times \mathbb{R}^{n_p} \to \mathbb{R}^{n_x}$, such that the differential state x_i satisfies the ODE

$$\forall t \in [i, i+1]: \quad \dot{x}_i(t) = h(t, x_i(t), \vartheta(\gamma_i, t-i), u_i(t), p) \quad \text{with} \quad x_i(i) = s_i \tag{27}$$

for all $\gamma_i \in D$. Here, we have used the assumption that f and g are twice continuously differentiable in all arguments, whereas $\frac{\partial g}{\partial z} \neq 0$ is regular. Note that the differential equation (27) can also be written in its integral form

$$x_i(t) = s_i + \int_i^{i+1} h(\tau, x_i(\tau), \vartheta(\gamma_i, \tau - i), u_i(\tau), p)\, \mathrm{d}\tau \tag{28}$$

for all $\gamma_i \in D$. Let us define the set $D^* := \{\gamma \in D \mid \gamma \neq 0\}$. The property (*iii*) of Lemma 3.1 guarantees that the function ϑ is at least on the domain D^*, differentiable with respect to γ_i. Consequently, the derivative function $x_i' := \frac{\partial x_i}{\partial \gamma_i}$ exists on D^* and can also be written in its integral form

$$\begin{aligned} x_i'(t) = &\int_i^t \frac{\partial h}{\partial x}(\tau, x_i(\tau), \vartheta(\gamma_i, \tau - i), u_i(\tau), p)\, x_i'(\tau)\, \mathrm{d}\tau \\ &+ \int_i^t \frac{\partial h}{\partial \vartheta}(\tau, x(\tau), \vartheta(\gamma_i, \tau - i), u_i(\tau), p)\, \vartheta'(\gamma_i, \tau)\, \mathrm{d}\tau \end{aligned}$$

for all $\gamma_i \in D^*$ and all $t \in [i, i+1]$. As the function h is twice continuously differentiable and as we have assumed that D is bounded, we can always find local Lipschitz constants $H_x, H_\vartheta > 0$, such that

$$\|x_i'(t)\| \leq \int_i^t H_x \|x_i'(\tau)\|\, \mathrm{d}\tau + H_\vartheta \int_i^t \|\vartheta'(\gamma_i, \tau)\|\, \mathrm{d}\tau \tag{29}$$

for all $\gamma_i \in D^*$. Now, an application of the integral form of Gronwall's Lemma yields

$$\|x_i'(i+1)\| \leq H_\vartheta\, e^{H_x} \int_0^1 \|\vartheta'(\gamma_i, \tau)\|\, \mathrm{d}\tau \tag{30}$$

that is, using property (*iii*) of Lemma 3.1 leads to the estimate

$$\|X_i'(s_i, \gamma_i, u_i, p)\| = \|x_i'(i+1)\| \leq \mathbf{O}(\|\gamma_i\|) \tag{31}$$

for all $\gamma_i \in D^*$.

UNCORRECTED PROOF

Copyright © 2012 John Wiley & Sons, Ltd.

DOI: 10.1002/oca

For the case $0 \notin D$, we have $D^* = D$, that is, we know already that X is differentiable on the set D. Otherwise, that is, for the case $0 \in D$, we use (28) in combination with Gronwall's Lemma to conclude that there exists a constant $C < \infty$ with

$$\begin{aligned}\|X_i(s_i, \gamma, u_i, p) - X_i(s_i, 0, u_i, p)\| &\leq C \max_{\tau\in[0,1]} \|\vartheta(\gamma, \tau) - \vartheta(0, \tau)\| \\ &= C \max_{\tau\in[0,1]} \|\vartheta(\gamma, \tau)\| \leq C\|\gamma\| \end{aligned} \tag{32}$$

for all $\gamma \in D$. Here, we have used the property (i) of Lemma 3.1, which guarantees that we have $\vartheta(0, \tau) = 0$ for all $\tau \in [0, 1]$.

Let $(\gamma^n) \in D^*$ be a sequence with

$$\lim_{n\to\infty} \gamma^n = 0$$

Now, it is guaranteed by the estimates (31) and (32) that the function value sequences $X_i(s_i, \gamma^n, u_i, p)$ and $X_i'(s_i, \gamma^n, u_i, p)$ both converge uniformly to $X_i(s_i, 0, u_i, p) \in \mathbb{R}^{n_x}$ and $0 \in \mathbb{R}^{n_x \times n_z}$, respectively, if n tends to infinity. Thus, we can conclude that the function X_i' exists on the whole set D and that we have $X_i'(s_i, 0, u_i, p) = 0$. Moreover, as the point $\overline{\gamma}$, around which the open set D has been constructed, was arbitrary, we can transfer our statement to the whole domain $\mathbb{R}^{n_z}$, that is X_i' exists everywhere, and we have $X_i'(s_i, 0, u_i, p) = 0$.

It remains to be shown that the function X_i' is locally Lipschitz continuous. For this aim, we start again with the integral representation (28) for $\gamma \in D^*$. Differentiating twice with respect to γ, taking the norm, and applying Gronwall's Lemma show that there exist constants $C_0, C_1, C_2, C_3 < \infty$ with

$$\begin{aligned}\|X_i''(s_i, \gamma_i, u_i, p)\| \leq C_0 &+ C_1 \int_0^1 \|\vartheta'(\gamma_i, \tau)\| \,\mathrm{d}\tau + C_2 \int_0^1 \|\vartheta'(\gamma_i, \tau)\|^2 \,\mathrm{d}\tau \\ &+ C_3 \int_0^1 \|\vartheta''(\gamma_i, \tau)\| \,\mathrm{d}\tau \end{aligned} \tag{33}$$

for all $\gamma \in D^*$. Here, $X'' := \frac{\partial^2 X}{\partial y_0^2}$ denotes the second derivative of X with respect to y_0. This second derivative exists on D^* as the second derivative $\vartheta'' := \frac{\partial^2 \vartheta}{\partial y_0^2}$ does exist on this domain. Using Property (iii) from Lemma 3.1, we conclude that there must be a constant $M < \infty$, such that

$$\forall \gamma_i \in D^*: \quad \|X_i''(s_i, \gamma_i, u_i, p)\| \leq M \tag{34}$$

In other words, the derivative X_i'' exists on D^* and is uniformly bounded on this set. This means that X_i' is a continuous function on D, whose derivative exists almost everywhere and is uniformly bounded. Hence, X_i' is locally Lipschitz continuous on D. With the same argument, we can continue this statement to the whole $\mathbb{R}^{n_z}$, that is, X_i' exists everywhere and is locally Lipschitz continuous. □

Remark 3.2

Properties (i) and (iii) from Lemma 3.1 do not uniquely characterize the relaxation function ϑ. The above theorem holds for all relaxation functions ϑ which satisfy these two properties. However, in this paper, we concentrate on the choice defined in (16), which turned out to work well in practice as it will also later be discussed in Section 5.

Remark 3.3

For the case that the relaxation function ϑ is defined by (16), it is worthwhile to discuss that the estimate (31) has the explicit form

$$\|X_i'(s_i, \gamma_i, u_i, p)\| \leq \frac{2H_\vartheta \, e^{H_x}}{a} \|\gamma_i\| \tag{35}$$

UNCORRECTED PROOF

Copyright © 2012 John Wiley & Sons, Ltd.

DOI: 10.1002/oca

This can be seen by using the estimate (24). The estimate (35) gives us an idea on how the function X_i' behaves with respect to the design parameter a. Indeed, choosing a very large a will lead to a small norm of X_i'. However, we should also recall that the slope of the function ϑ at the point $t = 0$ has, in Proposition 3.1, been estimated by

$$\left\| \dot{\vartheta}(\gamma, \tau) \right\| \leqslant a + \mathbf{O}(\|\gamma\|) \tag{36}$$

Thus, if we choose a very large a, the slope of ϑ cannot be guaranteed to be small anymore, which might lead to small steps taken by a numerical integration routine that is used to numerically solve the relaxed DAE system (11) on the basis of an adaptive step size control.

In the next step, we discuss a generalization of Theorem 3.1 for derivatives of the function X_i, with respect to the variables $\alpha_i := (s_i^T, u_i^T, p^T)^T$, which enter the DAE as the initial value, the control input, and the parameter, respectively.

Corollary 3.1
Requiring the same assumptions as in Theorem 3.1, the function $X_{i,\alpha_i} := \frac{\partial X_i}{\partial \alpha_i}$ is differentiable with respect to γ_i, and its Jacobian $X'_{i,\alpha_i} := \frac{\partial^2 X_i}{\partial \alpha_i \partial \gamma_i}$ is a locally Lipschitz continuous function. Moreover, we have $X'_{i,\alpha_i}(s_i, 0, u_i, p) = 0$.

Proof
The proof of this corollary is almost analogous to the proof of Theorem 3.1. We start again with the integral form (28) for $\gamma_i \in D^*$ and differentiate with respect to γ_i and α_i

$$\begin{aligned} \frac{\partial}{\partial \alpha_i} x_i'(t) = &\int_i^t \frac{\partial h}{\partial x} \frac{\partial}{\partial \alpha_i} x_i'(\tau)\, \mathrm{d}\tau + \int_i^t \left[\frac{\partial^2 h}{\partial x \partial \alpha_i} + \frac{\partial^2 h}{\partial x^2} \frac{\partial x_i}{\partial \alpha_i} \right] x_i'(\tau)\, \mathrm{d}\tau \\ &+ \int_i^t \left[\frac{\partial^2 h}{\partial \vartheta \partial \alpha_i} + \frac{\partial^2 h}{\partial \vartheta \partial x} \frac{\partial x_i}{\partial \alpha_i} \right] \vartheta'(\gamma_i, \tau)\, \mathrm{d}\tau \end{aligned} \tag{37}$$

for all $\gamma_i \in D^*$ and all $t \in [i, i+1]$. Thus, as we have $\|x_i'(\tau)\| \leqslant H_\vartheta e^{H_x} \int_i^\tau \|\vartheta'(\gamma_i, \tau')\|\, \mathrm{d}\tau'$ for all $\tau \in [i, i+1]$, we can find local Lipschitz constants H_x and $H_{\vartheta,\alpha}$, such that

$$\left\| \frac{\partial}{\partial \alpha_i} x_i'(t) \right\| \leqslant \int_i^t H_x \left\| \frac{\partial}{\partial \alpha_i} x_i'(t) \right\| \mathrm{d}\tau + H_{\vartheta,\alpha} \int_i^t \|\vartheta'(\gamma_i, \tau)\| \mathrm{d}\tau \tag{38}$$

Now, the integral form of Gronwall's Lemma yields

$$\|X'_{i,\alpha_i}(s_i, \gamma_i, u_i, p)\| = \left\| \frac{\partial}{\partial \alpha_i} x_i'(i+1) \right\| \leqslant H_{\vartheta,\alpha} e^{H_x} \int_0^1 \|\vartheta'(\gamma_i, \tau)\|\, \mathrm{d}\tau \leqslant \mathbf{O}(\|\gamma_i\|) \tag{39}$$

for $\gamma_i \in D^*$. For the case $0 \in D$, we use the Lipschitz relation

$$\|X_{i,\alpha_i}(s_i, \gamma_i, u_i, p) - X_{i,\alpha_i}(s_i, 0, u_i, p)\| \leqslant \mathbf{O}(\|\gamma_i\|) \tag{40}$$

such that we can apply the same argumentation as in the Theorem 3.1. For every sequence $(\gamma^n) \in D^*$ with $\lim_{n\to\infty} \gamma^n = 0$, the function value sequences $X_{i,\alpha_i}(s_i, \gamma^n, u_i, p)$ and $X'_{i,\alpha_i}(s_i, \gamma^n, u_i, p)$ converge for $n \to \infty$ uniformly to $X_{i,\alpha_i}(s_i, 0, u_i, p)$ and 0, respectively. Thus, the function X_{i,α_i} is differentiable on D, and we have $X'_{i,\alpha_i}(s_i, 0, u_i, p) = 0$. This statement can be continued to the whole domain.

It remains to show the local Lipschitz continuity of the function X'_{i,α_i}. Again, the argumentation is analogous to Theorem 3.1. First, we differentiate the integral form (37) once more, with respect to γ_i, take the norm on both sides, apply the integral form of Gronwall's Lemma, and find the estimate

$$\begin{aligned} \|X''_{i,\alpha_i}(s_i, \gamma_i, u_i, p)\| \leqslant\ & C_{\alpha,0} + C_{\alpha,1} \int_0^1 \|\vartheta'(\gamma_i, \tau)\|\, \mathrm{d}\tau + C_{\alpha,2} \int_0^1 \|\vartheta'(\gamma_i, \tau)\|^2\, \mathrm{d}\tau \\ &+ C_{\alpha,3} \int_0^1 \|\vartheta''(\gamma_i, \tau)\|\, \mathrm{d}\tau < \infty \end{aligned} \tag{41}$$

UNCORRECTED PROOF

Copyright © 2012 John Wiley & Sons, Ltd.

DOI: 10.1002/oca

for some constants $C_{\alpha,0}, C_{\alpha,1}, C_{\alpha,2}, C_{\alpha,3} < \infty$ and for all $\gamma_i \in D^*$. Obviously, we can now apply a completely analogous argumentation as in Theorem 3.1 to show that the function X'_{i,α_i} is locally Lipschitz continuous. □

4. INEXACT SQP METHODS FOR DAE SYSTEMS

In this section, we come back to the question how to numerically solve the discretized optimization problem (15) of the form

$$\begin{array}{ll} \underset{\alpha,\beta}{\text{minimize}} & F(\alpha,\beta) \\ \text{subject to:} & 0 = G(\alpha,\beta) \\ & 0 = H(\alpha, G(\alpha,\beta)) \end{array} \tag{42}$$

Recall that F and G are twice continuously differentiable functions, where G is defined by (5), whereas F is the discretized objective. The function H has, in Section 2, been defined within (12).

Proposition 4.1
Let the relaxation function ϑ satisfy the assumptions of Theorem 3.1. Then, the following statements hold:

(*i*) The function H is totally differentiable with respect to the arguments (α,β), and the associated total derivative functions

$$\frac{\mathrm{d}H}{\mathrm{d}\alpha} := \frac{\partial H}{\partial \alpha} + \frac{\partial H}{\partial G}\frac{\partial G}{\partial \alpha} \quad \text{and} \quad \frac{\mathrm{d}H}{\mathrm{d}\beta} := \frac{\partial H}{\partial G}\frac{\partial G}{\partial \beta}$$

are locally Lipschitz continuous. In addition, we have $\frac{\mathrm{d}H}{\mathrm{d}\alpha}(\alpha,0) = \frac{\partial H}{\partial \alpha}(\alpha,0)$, as well as $\frac{\mathrm{d}H}{\mathrm{d}\beta}(\alpha,0) = 0$.

(*ii*) The derivative of $H(\cdot, G(\cdot,\cdot))$ with respect to its first argument, denoted by $\frac{\partial H}{\partial \alpha}$, is totally differentiable in (α,β), and the total derivative function $\frac{\mathrm{d}}{\mathrm{d}(\alpha,\beta)}\frac{\partial H}{\partial \alpha}$ is locally Lipschitz continuous. In addition, we have $\frac{\mathrm{d}}{\mathrm{d}\alpha}\frac{\partial H}{\partial \alpha}(\alpha,0) = \frac{\partial^2 H}{\partial \alpha^2}(\alpha,0)$, as well as $\frac{\mathrm{d}}{\mathrm{d}\beta}\frac{\partial H}{\partial \alpha}(\alpha,0) = 0$.

Proof
We use the notation $X := (X_0^T, \ldots, X_{m-1}^T)^T$. The derivative of H can now be reduced to the derivatives of X, as we have

$$\frac{\mathrm{d}H}{\mathrm{d}\alpha} = \frac{\partial H}{\partial \alpha} + \sum_{i=0}^{m-1} \frac{\partial H}{\partial \gamma_i}\frac{\mathrm{d}\gamma_i}{\mathrm{d}\alpha} = \frac{\partial H}{\partial \alpha} - \sum_{i=0}^{m-1} \frac{\partial X}{\partial \gamma_i}\frac{\mathrm{d}\gamma_i}{\mathrm{d}\alpha} \tag{43}$$

as well as

$$\frac{\mathrm{d}H}{\mathrm{d}\beta} = \sum_{i=0}^{m-1} \frac{\partial H}{\partial \gamma_i}\frac{\mathrm{d}\gamma_i}{\mathrm{d}\beta} = -\sum_{i=0}^{m-1} \frac{\partial X}{\partial \gamma_i}\frac{\mathrm{d}\gamma_i}{\mathrm{d}\beta} \tag{44}$$

where the existence and local Lipschitz continuity of the derivatives $\frac{\partial X}{\partial \gamma_i}$ is for all indices $i \in \{0,\ldots,m-1\}$ guaranteed by Theorem 3.1. In particular, Theorem 3.1 guarantees that the derivatives of the form $\frac{\partial X}{\partial \gamma_i}$ in (43) and (44) vanish at every consistent initialization point, that is, we have $\frac{\mathrm{d}H}{\mathrm{d}\alpha}(\alpha,0) = \frac{\partial H}{\partial \alpha}(\alpha,0)$, as well as $\frac{\mathrm{d}H}{\mathrm{d}\beta}(\alpha,0) = 0$, and Statement (*i*) is proven.

In the next step, we use Corollary 3.1 to show that also the Lipschitz continuous derivatives

$$\frac{\mathrm{d}}{\mathrm{d}\alpha}\frac{\partial H}{\partial \alpha} = \frac{\partial^2 H}{\partial \alpha^2} - \sum_{i=0}^{m-1} \frac{\partial^2 X}{\partial \alpha \partial \gamma_i}\frac{\mathrm{d}\gamma_i}{\mathrm{d}\alpha} \tag{45}$$

$$\text{and} \quad \frac{\mathrm{d}}{\mathrm{d}\beta}\frac{\partial H}{\partial \alpha} = -\sum_{i=0}^{m-1} \frac{\partial^2 X}{\partial \alpha \partial \gamma_i}\frac{\mathrm{d}\gamma_i}{\mathrm{d}\beta} \tag{46}$$

UNCORRECTED PROOF

Copyright © 2012 John Wiley & Sons, Ltd.

DOI: 10.1002/oca

exist. In particular, it follows from Corollary 3.1 that we have $\frac{\mathrm{d}}{\mathrm{d}\alpha}\frac{\partial H}{\partial \alpha}(\alpha,0) = \frac{\partial^2 H}{\partial \alpha^2}(\alpha,0)$ as well as $\frac{\mathrm{d}}{\mathrm{d}\beta}\frac{\partial H}{\partial \alpha}(\alpha,0) = 0$, which leads to Statement (*ii*). □

Note that an important consequence of the previous proposition is that all functions in the optimization problem (42) are continuously differentiable, that is, we can formulate first order KKT conditions.

In the following, we regard the optimization problem (42) independent of our DAE context. Although we might, of course, still have relaxed DAE's in mind, we will from now on only require that the functions F and G are twice continuously differentiable, whereas the function H can be any function that satisfies Properties (*i*) and (*ii*) of Proposition 4.1.

Lemma 4.1
Let F and G be continuously differentiable, while H satisfies the properties (*i*) and (*ii*) of Proposition 4.1. Let (α,β) be a minimizer of the problem (42) at which the matrix

$$\begin{pmatrix} \frac{\mathrm{d}G}{\mathrm{d}\alpha}(\alpha,\beta) & \frac{\mathrm{d}G}{\mathrm{d}\beta}(\alpha,\beta) \\ \frac{\partial H}{\partial \alpha}(\alpha, G(\alpha,\beta)) & 0 \end{pmatrix} \tag{47}$$

has full rank. Then, (α,β) is a KKT point, and there exist multipliers $\lambda \in \mathbb{R}^{n_G}$ and $\mu \in \mathbb{R}^{n_H}$, such that the following conditions are satisfied:

$$0 = \frac{\mathrm{d}F}{\mathrm{d}\alpha}(\alpha,\beta) + \lambda^T \frac{\mathrm{d}G}{\mathrm{d}\alpha}(\alpha,\beta) + \mu^T \frac{\partial H}{\partial \alpha}(\alpha, G(\alpha,\beta)) \tag{48}$$

$$0 = \frac{\mathrm{d}F}{\mathrm{d}\beta}(\alpha,\beta) + \lambda^T \frac{\mathrm{d}G}{\mathrm{d}\beta}(\alpha,\beta) \tag{49}$$

$$0 = G(\alpha,\beta) \tag{50}$$

$$0 = H(\alpha, G(\alpha,\beta)) \tag{51}$$

Proof
If the point (α,β) is a minimizer of the problem (42), it must satisfy the feasibility condition $G(\alpha,\beta) = 0$. Consequently, we can use Properties (*i*) and (*ii*) of Proposition 4.1 to conclude that the matrix

$$\begin{pmatrix} \frac{\mathrm{d}G}{\mathrm{d}\alpha}(\alpha,\beta) & \frac{\mathrm{d}G}{\mathrm{d}\beta}(\alpha,\beta) \\ \frac{\mathrm{d}H}{\mathrm{d}\alpha}(\alpha, G(\alpha,\beta)) & \frac{\mathrm{d}H}{\mathrm{d}\beta}(\alpha, G(\alpha,\beta)) \end{pmatrix} = \begin{pmatrix} \frac{\mathrm{d}G}{\mathrm{d}\alpha}(\alpha,\beta) & \frac{\mathrm{d}G}{\mathrm{d}\beta}(\alpha,\beta) \\ \frac{\partial H}{\partial \alpha}(\alpha, G(\alpha,\beta)) & 0 \end{pmatrix}$$

has full rank, that is, the linear independence constraint qualification is satisfied at the minimizer (α,β). Thus, (α,β) is a KKT point. The corresponding first order necessary conditions are equivalent to the conditions (48)–(51). This can be verified by using Properties (*i*) and (*ii*) of Proposition 4.1 again. □

Remark 4.1
An important observation in the above Lemma is that an evaluation of KKT conditions (48)–(51) does not require the computation of the derivative of H with respect to β. The algorithm which is proposed in the following will make use of this observation.

In order to numerically determine KKT point $w := (\alpha,\beta,\lambda,\mu)$, satisfying the conditions (48)–(51), we plan to apply a Newton-type method. For this aim, we start at an initial point w^0, generating iterates of the form $w^{k+1} := w^k + A(w^k)^{-1}\Phi(w^k)$, where the function $\Phi : \mathbb{R}^{n_w} \to \mathbb{R}^{n_w}$ (with $n_w := n_\alpha + n_\beta + n_G + n_H$) is defined as

$$\forall w \in \mathbb{R}^{n_w}: \quad \Phi(w) := \begin{pmatrix} \frac{\mathrm{d}F}{\mathrm{d}\alpha}(\alpha,\beta)^T + \frac{\mathrm{d}G}{\mathrm{d}\alpha}(\alpha,\beta)^T\lambda + \frac{\partial H}{\partial \alpha}(\alpha, G(\alpha,\beta))^T\mu \\ \frac{\mathrm{d}F}{\mathrm{d}\beta}(\alpha,\beta)^T + \frac{\mathrm{d}G}{\mathrm{d}\beta}(\alpha,\beta)^T\lambda \\ G(\alpha,\beta) \\ H(\alpha, G(\alpha,\beta)) \end{pmatrix} \tag{52}$$

UNCORRECTED PROOF

Copyright © 2012 John Wiley & Sons, Ltd.

DOI: 10.1002/oca

whereas the matrix-valued function $A : \mathbb{R}^{n_w} \to \mathbb{R}^{n_w \times n_w}$ is defined as

$$A(w) := \begin{pmatrix} \frac{\mathrm{d}^2 L(\alpha,\beta,\lambda)}{\mathrm{d}\alpha^2} + \frac{\partial^2 \mu^T H}{\partial \alpha^2}(\alpha, G(\alpha,\beta)) & \frac{\mathrm{d}^2 L(\alpha,\beta,\lambda)}{\mathrm{d}\alpha \mathrm{d}\beta} & \frac{\mathrm{d}G(\alpha,\beta)}{\mathrm{d}\alpha}^T & \frac{\partial H}{\partial \alpha}(\alpha, G(\alpha,\beta))^T \\ \frac{\mathrm{d}^2 L(\alpha,\beta,\lambda)}{\mathrm{d}\beta \mathrm{d}\alpha} & \frac{\mathrm{d}^2 L(\alpha,\beta,\lambda)}{\mathrm{d}\beta^2} & \frac{\mathrm{d}G(\alpha,\beta)}{\mathrm{d}\beta}^T & 0 \\ \frac{\mathrm{d}G(\alpha,\beta)}{\mathrm{d}\alpha} & \frac{\mathrm{d}G(\alpha,\beta)}{\mathrm{d}\beta} & 0 & 0 \\ \frac{\partial H}{\partial \alpha}(\alpha, G(\alpha,\beta)) & 0 & 0 & 0 \end{pmatrix} \tag{53}$$

for all $w \in \mathbb{R}^{nW}$, where we use the notation

$$L(\alpha,\beta,\lambda) := F(\alpha,\beta) + \lambda^T G(\alpha,\beta)$$

For the moment, we assume here that the matrix $A(w)$ is always invertible, such that the iterations are well defined. But, we will discuss later, in more detail, under which conditions this can be guaranteed.

Lemma 4.2
Let F and G be twice continuously differentiable, while H satisfies Properties (i) and (ii) of Proposition 4.1. The function Φ is a differentiable function, and its Jacobian $\Phi' := \frac{\partial \Phi}{\partial w}$ is locally Lipschitz continuous on $\mathbb{R}^{n_w}$. Moreover, for every point $w^* \in \mathbb{R}^{n_w}$, which satisfies $\Phi(w^*) = 0$, we have $\Phi'(w^*) = A(w^*)$.

Proof
The question of whether the function Φ is differentiable is obviously reduced to the question of whether the derivatives

$$\frac{\mathrm{d}H}{\mathrm{d}\beta} \quad \text{and} \quad \frac{\mathrm{d}}{\mathrm{d}\beta}\frac{\partial H}{\partial \alpha} \tag{54}$$

exist. As the existence and local Lipschitz continuity of both terms is guaranteed by Properties (i) and (ii) of Proposition 4.1, Φ is differentiable, and its Jacobian is locally Lipschitz continuous. Now, we have

$$\Phi'(w^*) - A(w^*) = \begin{pmatrix} 0 & \mu^T \frac{\mathrm{d}}{\mathrm{d}\beta}\frac{\partial H}{\partial \alpha}(\alpha^*, G(\alpha^*,\beta^*)) & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & \frac{\mathrm{d}H}{\mathrm{d}\beta}(\alpha^*, G(\alpha^*,\beta^*)) & 0 & 0 \end{pmatrix} = 0 \tag{55}$$

as we have $G(\alpha^*,\beta^*) = 0$ at every point w^*, which satisfies $\Phi(w^*) = 0$. □

Remark 4.2
The above Newton-type method can of course also be interpreted as an SQP-type method. As this method is neither based on exact Hessians nor on exact constraint Jacobians, it can be regarded as an inexact SQP method. Note that the approximation of the Hessian and the constraint Jacobian has the advantage that no derivatives of the function H with respect to β are needed, which is especially beneficial in the case that we have a large number of algebraic states in the DAE. Moreover, the mentioned SQP formulation can, in an obvious way, be transferred to the case that we also have inequalities.

The final step of this section is to discuss the local convergence properties of the suggested inexact SQP method.

Theorem 4.1
Let F and G be twice continuously differentiable, while H satisfies Properties (i) and (ii) of Proposition 4.1. If the discretized DAE optimization problem (42) has a strict local minimum at the primal dual KKT point w^*, whereas A is a regular function, then there exists an open neighbor-

UNCORRECTED PROOF

Copyright © 2012 John Wiley & Sons, Ltd.

DOI: 10.1002/oca

hood $\mathcal{N}(w^*)$ around $w^* \in \mathcal{N}(w^*)$, such that for all initial points $w^0 \in \mathcal{N}(w^*)$, the sequence (w^k) converges q-quadratically to w^*.

Proof
The proof is based on the standard argumentation for Newton methods. Let w^* be a point that satisfies $\Phi(w^*) = 0$. Now, we have, for all $k \in \mathbb{N}$,

$$\begin{aligned} w^{k+1} - w^* &= w^k - w^* - A(w^k)^{-1}\Phi(w^k) \\ &= w^k - w^* - A(w^k)^{-1}\int_0^1 \Phi'(w^* + s)(w^k - w^*)\,\mathrm{d}s \\ &= A(w^k)^{-1}\int_0^1 \Big[A(w^k) - \Phi'(w^* + s)\Big]\,\mathrm{d}s \end{aligned} \tag{56}$$

We know from Lemma 4.2 that Φ' is locally Lipschitz continuous and $\Phi'(w^*) = A(w^*)$, that is, there exists a constant $L < \infty$, such that for all $s \in [0, 1]$, the inequality

$$\|A(w^k)^{-1}\Big[A(w^k) - \Phi'(w^* + s(w^k - w^*))\Big](w^k - w^*)\| \leqslant L\,\|w^k - w^*\|^2$$

is satisfied. Thus, we find with (56) that

$$\|w^{k+1} - w^*\| \quad \leqslant \quad L\,\|w^k - w^*\|^2 \tag{57}$$

which shows the q-quadratic convergence of the method and leads immediately to the statement of the theorem. □

Remark 4.3

(*i*) The mentioned theorem assumes that the matrix-valued function A is regular. However, it would also be enough to require that $A(w^*)$ is regular, as this implies that there is a neighborhood of w^* in which A is regular. Moreover, it can easily be seen that $A(w^*)$ must be regular if the original unrelaxed optimization problem (4) satisfies the second-order sufficient condition for a minimum at w^*, whereas the constraints satisfy the linear independence constraint qualification. (This follows immediately from the equivalence relations $\hat{H}(\cdot) \equiv H(\cdot, \beta^*)$, $\frac{\partial}{\partial\alpha}\hat{H}(\cdot) \equiv H_\alpha(\cdot, \beta^*)$, and $\frac{\partial^2}{\partial\alpha^2}\hat{H}(\cdot) \equiv H_{\alpha.\alpha}(\cdot, \beta^*)$.)

(*ii*) The above theorem requires that the functions F and G are twice continuously differentiable, while H satisfies Properties (*i*) and (*ii*) of Proposition 4.1. Note that these requirements are satisfied, if F, the original DAE right-hand side functions f and g, as well as the constraint function r are twice continuously differentiable, whereas the relaxation function ϑ satisfies Properties (*i*) and (*iii*) of Lemma 3.1.

(*iii*) Note that the proposed inexact SQP algorithm does not require the computation of any derivatives of the functions X_i with respect to the algebraic node variables β_i. Hence, the proposed inexact SQP algorithm is beneficial if we have many algebraic states.

(*iv*) One of the main advantages of the inexact SQP method, which is proposed here, is that it is easy to implement. We can simply take an existing SQP method, combe it with our favorite DAE integrator by replacing the algebraic conditions with the relaxed version, and stack the consistency conditions to the equality constraints. At the place where we would usually call the integrator to compute the derivatives of the differential states with respect to the variables γ_i, we pretend that we have never introduced any relaxation, ignoring this dependency. The mentioned Theorem guarantees that this way of dealing with DAEs does not destroy the q-quadratic convergence of the SQP method.

(*v*) Note that several variations of the previous method are possible. For example, instead of exactly computing the Hessians, we could replace them with an approximation on the basis of BFGS updates.

UNCORRECTED PROOF

Copyright © 2012 John Wiley & Sons, Ltd.

DOI: 10.1002/oca

(*vi*) Finally, we compare the above inexact SQP algorithm with the PRSQP method for DAE's, developed by Leineweber [10]. The PRSQP method is an exact SQP method, but the computation of the state sensitivities and the SQP algorithm are deeply intertwined, such that at the end, only

$$n_x + n_u + n_p + 1$$

forward directions are needed in every step of the optimization algorithm. The inexact SQP method proposed here requires the computation of $n_x + n_u + n_p$ forward derivatives of the state trajectory in every step, that is, we need 1 direction less, which is only a minor advantage. The main advantage of the inexact SQP method proposed here is that it can easily be implemented. In fact, there exist a lot of open, as well as commercial, SQP implementations, and there is also a number of DAE integrators available. As the PRSQP method requires deep intertwining of the SQP routine and the sensitivity generation, it is not so easily possible to couple or exchange existing software modules. In contrast, the inexact SQP method proposed here can deal with existing integrators and SQP methods, as it is explained previously.

(*vii*) Note that the mentioned method can easily be combined with the Lifted Newton methods [20] for the case that we have both many differential and many algebraic states but only a moderate number of control inputs and parameters.

5. NUMERICAL TEST EXAMPLES

In this section, we address the question whether the proposed inexact SQP method also works in practice. In the previous section, we have proven the theoretical properties of the proposed method, summarized in Theorem 4.1, which states that we can expect q-quadratic convergence of the method. However, we should also ask some critical questions that have only been partially addressed by our theoretical results: how should we choose the design parameters a and b associated with the relaxation function ϑ

$$\vartheta(\gamma,\tau) := \begin{cases} \gamma\,(1-\tau)^{\frac{a+b\|\gamma\|}{\|\gamma\|}} & \text{if } \|\gamma\| \neq 0 \\ 0 & \text{otherwise} \end{cases}$$

which has been defined in (16)? Does the choice of a influence the step size control of an integration routine? How does the inexact SQP algorithm behave if we choose a to be too small? In order to understand the relevance of these questions, recall estimates (35) and (36). From these estimates, we only know that the Lipschitz constant of the functions X_i' at a KKT point is small if a is large, whereas the slope of the function ϑ, with respect to the time t, is expected to be small for small as. Intuitively, we expect that the Lipschitz constant of the functions X_i' will influence the convergence behavior of the SQP method, whereas the slope of the function ϑ might have an influence on the step size control of the integration routine.

In order to give at least an empirical answer to these questions, we will test the inexact SQP method by applying it first, in Section 5.1, to a small toy example in order to understand the behavior of the method. Second, we apply the method in Section 5.2 to a large real-world optimal control problem for a distillation column.

5.1. A small toy example

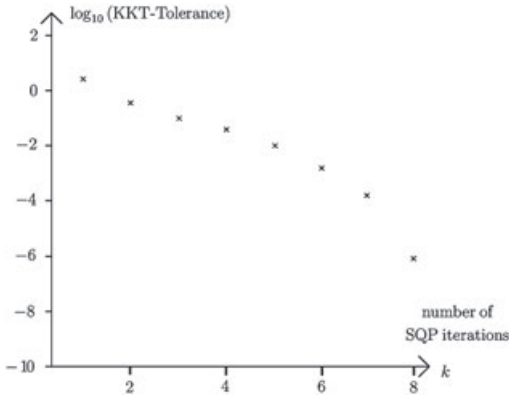
As a simple example for a DAE optimization, we consider the following optimal control problem

$$\begin{array}{lrcl} \underset{x(\cdot),z(\cdot),u(\cdot)}{\text{minimize}} & & & \int_0^5 x(t)^2 + 3u(t)^2\,\mathrm{d}t \\ \text{subject to:} & & & \\ \forall t \in [0,5]: & \dot{x}(t) & = & x(t)\left(\frac{1}{2}x(t)-1\right) + u(t) + \frac{1}{2}z(t) \\ \forall t \in [0,5]: & 0 & = & z(t) + e^{z(t)} + x(t) - 1 \\ & 0 & = & x(0) - 1 \end{array} \tag{58}$$

UNCORRECTED PROOF

Copyright © 2012 John Wiley & Sons, Ltd.

DOI: 10.1002/oca



k	1	2	3	...	7	8
KKT-Tol:	$3.67 \cdot 10^{0}$	$3.46 \cdot 10^{-1}$	$6.00 \cdot 10^{-2}$	$\cdots$	$1.79 \cdot 10^{-4}$	$7.59 \cdot 10^{-7}$

Q8 Figure 2. The KKT tolerance (KKT-Tol) associated with iterations $w^{k+1} = w^k - A(w^k)^{-1}\Phi(w^k)$ of the inexact SQP method applied to the DAE optimal control problem (58).

In this example, we have only 1 differential and 1 algebraic state. However, the differential and the algebraic equation are nonlinear. For our numerical test, we choose 10 multiple shooting intervals. Moreover, we use the ACADO BDF integrator [21], which is based on a backward differentiation formula (BDF) combined with a higher order diagonal implicit Runge–Kutta starter based on the algorithmic ideas in [18]. We also use the ACADO implementation of multiple shooting SQP methods.

Using the notation from the previous section, we start the Newton or SQP method with $w^0 = 0$, that is, all states and controls are simply initialized with 0. As a stopping criterion, we require the KKT tolerance to be less than 10^{-6}. Now, we use $a = \frac{1}{2}$ and $b = 1$ within the relaxation function (16). For these settings, the inexact SQP method converges rapidly within 8 iterations.

The corresponding minimum value of the objective is 0.49581. Here, it should be remarked that we did not use any globalization technique. That is, the full-step method was convergent in the previous example, although the starting guess $w^0 = 0$ is actually quite far from the optimal solution. For the case that the method is not convergent in the full-step mode, it is, for example, possible to apply trust region methods. However, such globalization techniques for inexact SQP methods are not in the scope of this paper, and we refer to [22] for further reading.

In this paper, we are rather interested in a discussion on how the method behaves with regard to the design parameter a in the relaxation function. Thus, we test the inexact SQP method for several values of a.

T1 In Table I, the total CPU time of the inexact SQP iterations until convergence is listed. This CPU time is, in our example, dominated by the time that is needed for the simulation of the DAE, as well as for first and second order sensitivity generation, whereas the solution of the sub-QPs takes usually less than 1 ms. If we choose a to be too large, the slope (36) of the relaxation is too high, such that the BDF integration routine takes many steps at the start of each interval, which is equivalent to a Phase 1 step satisfying the algebraic consistency conditions first before the step size can be increased. However, as soon as we choose a that is less than 10, we observe that the overall CPU time improves, whereas we need the same number of SQP iterations. The CPU time achieves its minimum if we choose the value, for a, of approximately 0.5. If we reduce a further, the approximations of the Jacobian and the Hessian become less accurate, such the CPU time increases again because of the fact that we need more SQP iterations. It is interesting to observe that, even for $a = 0$, the method is still convergent. In fact, even if we choose a to be too small, the method is still faster than an exact SQP method, which uses an expensive Phase 1 step in the integration routine (realized here via a large relaxation parameter a).

UNCORRECTED PROOF

Copyright © 2012 John Wiley & Sons, Ltd.

DOI: 10.1002/oca

Table I. The number of SQP iterations and the associated total CPU time of the inexact SQP method for several values of the design parameter a.

a	# SQP iterations	total CPU time
50	7	204 ms
10	7	203 ms
5	7	181 ms
1	7	155 ms
0.5	8	137 ms
0.1	10	149 ms
0.05	11	170 ms
0	12	181 ms

5.2. *Optimal control of a distillation column*

In this section, we consider a DAE model for a distillation column taken from the literature [13]. This DAE model has 82 differential states, 122 algebraic states, as well as 2 time dependent control inputs. In this paper, we will not restate all the equations that are needed to build up the model of the distillation column, but we will just mention that the first 42 differential states represent molar methanol concentrations in the reboiler and condenser of the distillation column, whereas the remaining 40 differential states are the molar tray holdups. The liquid and vapor molar fluxes (40 each) together with the 42 temperatures of the reboiler, the 40 trays, and the condenser are the algebraic state vectors, that is we have $n_z = 40 + 40 + 42 = 122$ algebraic states. However, for the details of the model, we refer to [13]. The optimal control problem of our interest is also taken from [13]. We consider a least square problem of the form

$$\begin{aligned} \underset{x(\cdot),z(\cdot),u(\cdot)}{\text{minimize}} \quad & \int_0^T \left\| \widetilde{T} z(t) - T_{\text{ref}} \right\|^2 + \| R(u(t) - u_{\text{S}}) \|^2 \, \mathrm{d}t \\ \text{subject to:} \quad & \\ \forall t \in [0,T]: \quad & \dot{x}(t) = f(t, x(t), z(t), u(t)) \\ \forall t \in [0,T]: \quad & 0 = g(t, x(t), z(t), u(t)) \\ & x(0) = x_0 \end{aligned} \tag{59}$$

where T_{ref} is the reference temperature, at which we would like to operate the system, $\widetilde{T}$ a projection matrix that filters the temperatures out of the algebraic state vector that we would like to penalize, u_{S} a control set point, and x_0 a given initial state (e.g. from a measurement). Note that this problem is exactly coinciding with the optimization problem (7.22) in [13]. Here, we chose $T = 1000\,\text{s}$ to be the time horizon, whereas the 6 multiple shooting discretization intervals are also chosen as suggested in [13].

We employ the same numerical settings as for the toy example from the previous section, that is we use the ACADO toolkit, BDF integrator, and multiple shooting SQP implementation [21] to implement the inexact SQP method for the distillation column problem (59). The problem turns out to be only mildly nonlinear, such that the method converges for $a = 0.5$ in 5 iterations, which corresponds to approximately 30 s of computation time.

In this example, the choice of the parameter a did not influence the number of SQP iterations, that is we always need 5 SQP iterations (Table II). However, for $a = 0.5$ or smaller values of a, the BDF integration routine needs approximately 119 integrator steps per simulation and intervals, whereas values $a \gg 1$ lead to approximately 145 steps, that is we save approximately 20% of computation time in comparison to an exact SQP method with Phase 1 step. T2 Q9

Summarizing our observations, the inexact SQP algorithm proposed in this paper is surprisingly robust with respect to the choice of the design parameter a. Choosing a close to 0.5 led to significant savings in terms of computation time.

UNCORRECTED PROOF

Copyright © 2012 John Wiley & Sons, Ltd.

DOI: 10.1002/oca

Table II. The KKT tolerance associated with the iterations of the inexact SQP method applied to a real-world optimal control problem with 82 differential and 122 algebraic states.

k	1	2	3	4	5
KKT-tol:	$2.252 \cdot 10^{0}$	$4.230 \cdot 10^{-1}$	$7.34 \cdot 10^{-3}$	$2.32 \cdot 10^{-5}$	$1.52 \cdot 10^{-9}$

KKT-tol, KKT tolerance.

6. CONCLUSIONS AND OUTLOOK

In this paper, we have presented an inexact SQP method for DAE optimization problems, which is designed for the case that many implicit algebraic states occur in the DAE system. The contribution of this paper is presented in two parts. First, we have introduced a special relaxation function for differential algebraic equations with desirable properties, as discussed in Theorem 3.1 and Corollary 3.1. Using this relaxation function in the context of DAE optimal control leads to optimization problems of the form (42). Second, we have proposed an inexact SQP method for optimization problems of the form (42), provided that the constraint function H satisfies the properties of Proposition 4.1. Although this SQP method is inexact, in the sense that both the Jacobian and Hessian matrices are only approximated, the q-quadratic convergence of the proposed method has been proven within Theorem 4.1. Finally, the method has been applied to a small toy example as well as to a nontrivial optimal control problem, namely the optimization of a distillation column with 82 differential and 122 implicit algebraic states. The numerical method proposed in this paper turned out to be reliable and efficient in solving these problems.

ACKNOWLEDGEMENTS

This research is supported by the Research Council KUL: CoE EF/05/006 Optimization in Engineering(OPTEC), IOF-SCORES4CHEM, GOA/10/009 (MaNet), GOA/10/11; several PhD/postdoc and fellow grants; the Flemish Government: FWO; PhD/postdoc grants, projects G.0452.04, G.0499.04, G.0211.05, G.0226.06, G.0321.06, G.0302.07, G.0320.08, G.0558.08, G.0557.08, G.0588.09,G.0377.09; research communities (ICCoS, ANMMM, MLDM); IWT; PhD Grants, Belgian Federal Science Policy Office: IUAP P6/04; EU: ERNSI; FP7-HDMPC, FP7-EMBOCON, Contract Research: AMINAL. Other supporters include the Helmholtz-viCERP and COMET-ACCM.

UNCORRECTED PROOF

REFERENCES

1. Biegler LT. Solution of dynamic optimization problems by successive quadratic programming and orthogonal collocation. *Computers and Chemical Engineering* 1984; **8**:243–248.
2. Schulz VH. Reduced SQP methods for large-scale optimal control problems in DAE with application to path planning problems for satellite mounted robots. *PhD thesis*, Universität Heidelberg, 1996.
3. Kraft D. On converting optimal control problems into nonlinear programming problems. In *Computational Mathematical Programming*, Vol. F15, Schittkowski K (ed.), NATO ASI. Springer, 1985; 261–280.
4. Sargent RWH, Sullivan GR. The development of an efficient optimal control package. In *Proceedings of the 8th IFIP Conference on Optimization Techniques (1977), Part 2*, Stoer J (ed.). Springer: Heidelberg, 1978.
5. Bock HG, Plitt KJ. A multiple shooting algorithm for direct solution of optimal control problems. In *Proceedings 9th IFAC World Congress Budapest*. Pergamon Press, 1984; 243–247.
6. Plitt KJ. Ein superlinear konvergentes Mehrzielverfahren zur direkten Berechnung beschränkter optimaler Steuerungen. *Master's thesis*, Universität Bonn, 1981.
7. Hinsberger H. Ein direktes Mehrzielverfahren zur Lösung von Optimalsteuerungsproblemen mit grossen, differential–algebraischen Gleichungssystemen und Anwendungen aus der Verfahrenstechnik. *PhD thesis*, Technische Universität Clausthal, 1998.
8. Hinsberger H, Miesbach S, Pesch HJ. Optimal temperature control of semibatch polymerization reactors. In *Scientific Computing in Chemical Engineering*, Keil F, Mackens W, Voss H, Werther J (eds). Springer: Heidelberg, 1996.
9. Bock HG, Eich E, Schlöder JP. Numerical solution of constrained least squares boundary value problems in differential-algebraic equations. In *Numerical Treatment of Differential Equations*, Strehmel K (ed.). Teubner: Leipzig, 1988.
10. Leineweber DB. *Efficient Reduced SQP Methods for the Optimization of Chemical Processes Described by Large Sparse DAE Models*, Fortschritt-Berichte VDI Reihe 3, Verfahrenstechnik, Vol. 613. VDI Verlag: Düsseldorf, 1999.

Copyright © 2012 John Wiley & Sons, Ltd.

DOI: 10.1002/oca

11. Diehl M, Walther A, Bock HG, Kostina E. An adjoint-based SQP algorithm with quasi-Newton Jacobian updates for inequality constrained optimization. *Optimization Methods and Software* 2009. Accepted for publication. **Q14**
12. Wirsching L. An SQP algorithm with inexact derivatives for a direct multiple shooting method for optimal control problems. *Master's thesis*, University of Heidelberg, 2006.
13. Diehl M. Real-time optimization for large scale nonlinear processes. *PhD thesis*, Universität Heidelberg, 2001. Available from: http://www.ub.uni-heidelberg.de/archiv/1659/. **Q15**
14. Boggs PT, Tolle JW. Sequential quadratic programming. *Acta Numerica* 1995:1–51. **Q16**
15. Büskens C, Maurer H. SQP-methods for solving optimal control problems with control and state constraints: adjoint variables, sensitivity analysis and real-time control. *Journal of Computational and Applied Mathematics* 2000; **120**:85–108.
16. Rheinboldt WC. Differential-algebraic systems as differential equations on manifolds. *Math of Comput* 1984; **43**:473–482. **Q17**
17. Hairer E, Wanner G. RADAU5 - an implicit Runge–Kutta code. Report, Université de Genève, Dept. de mathématiques, Genève, 1988.
18. Bauer I. Numerische Verfahren zur Lösung von Anfangswertaufgaben und zur Generierung von ersten und zweiten Ableitungen mit Anwendungen bei Optimierungsaufgaben in Chemie und Verfahrenstechnik. *PhD thesis*, Universität Heidelberg, 1999.
19. Curtiss CF, Hirschfelder JO. Integration of stiff equations. *Proc Nat Acad Sci* 1952; **38**:235–243.
20. Albersmeyer J, Diehl M. The lifted Newton method and its application in optimization. *SIAM Journal on Optimization* 2010; **20**(3):1655–1684.
21. Houska B, Ferreau HJ, Diehl M. ACADO toolkit–an open source framework for Automatic Control and Dynamic Optimization. *Optimal Control Applications and Methods* 2010, in print. DOI: 10.1002/oca.939. **Q18**
22. Walther A, Biegler L. Numerical experiments with an inexact Jacobian trust-region algorithm. *Computational Optimization and Applications* 2007. DOI: 10.1007.

UNCORRECTED PROOF

Copyright © 2012 John Wiley & Sons, Ltd.

Author Query Form

Journal: Optimal Control Applications and Methods

Article: oca_2026

Dear Author,

During the copyediting of your paper, the following queries arose. Please respond to these by annotating your proofs with the necessary changes/additions.

- If you intend to annotate your proof electronically, please refer to the E-annotation guidelines.
- If you intend to annotate your proof by means of hard-copy mark-up, please refer to the proof mark-up symbols guidelines. If manually writing corrections on your proof and returning it by fax, do not write too close to the edge of the paper. Please remember that illegible mark-ups may delay publication.

Whether you opt for hard-copy or electronic annotation of your proofs, we recommend that you provide additional clarification of answers to queries by entering your answers on the query sheet, in addition to the text mark-up.

UNCORRECTED PROOF

Query No.	Query	Remark
Q1	AUTHOR: Please check that authors and their affiliations are correct.	
Q2	AUTHOR: As per journal instruction, reference citations are not allowed in the abstract. Please delete the citations and reword the sentence.	
Q3	AUTHOR: Reference citations have been renumbered according to its order of appearance in the text. Please check and confirm that there are no mistakes.	
Q4	AUTHOR: A running head short title was not supplied; please check if this one is suitable and, if not, please supply a short title of up to 60 characters that can be used instead.	
Q5	AUTHOR: "These methods have the advantage that..." This sentence has been reworded for clarity. Please check and confirm it is correct.	
Q6	AUTHOR: 'Figure 3' and 'Table 5.1' have been changed to 'Figure 1' and 'Table I', respectively. Please check if this is correct.	
Q7	AUTHOR: Please define KKT.	
Q8	AUTHOR: Figure 2 has not been mentioned in the text. Please cite the figure in the relevant place in the text. Also, please define/explain the relevance of the use of k in the figure.	

Q9	AUTHOR: Table 2 was not cited in the text. An attempt has been made to insert the table into a relevant point in the text - please check that this is OK. If not, please provide clear guidance on where it should be cited in the text. Also, please define/explain the relevance of the use of k in Table II.	
Q10	AUTHOR: Please check Reference 3 if correct and provide the city location of publisher.	
Q11	AUTHOR: Please provide the city location of publisher for Reference 5.	
Q12	AUTHOR: Please provide page range for Reference 8.	
Q13	AUTHOR: Please check Reference 10 if correct.	
Q14	AUTHOR: If this reference has now been published online, please add relevant year/DOI information. If this reference has now been published in print, please add relevant volume/issue/page/year information.	
Q15	AUTHOR: Please provide accessed date for Reference 13.	
Q16	AUTHOR: Please provide volume number for Reference 14.	
Q17	AUTHOR: Please provide full journal title for References 16 and 19.	
Q18	AUTHOR: If this reference has now been published in print, please add relevant volume/issue/page/year information.	

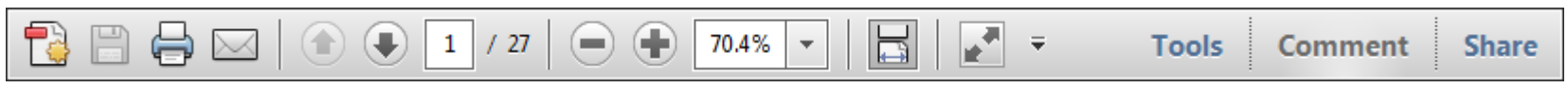
UNCORRECTED PROOF

WILEY-BLACKWELL

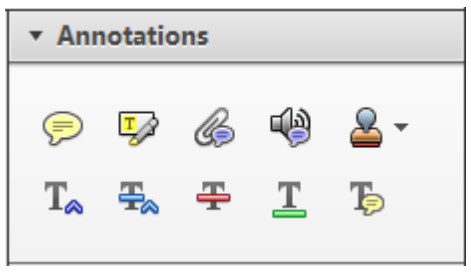
USING e-ANNOTATION TOOLS FOR ELECTRONIC PROOF CORRECTION

Required software to e-Annotate PDFs: Adobe Acrobat Professional or Adobe Reader (version 8.0 or above). (Note that this document uses screenshots from Adobe Reader X)
The latest version of Acrobat Reader can be downloaded for free at: http://get.adobe.com/reader/

Once you have Acrobat Reader open on your computer, click on the Comment tab at the right of the toolbar:



This will open up a panel down the right side of the document. The majority of tools you will use for annotating your proof will be in the Annotations section, pictured opposite. We've picked out some of these tools below:



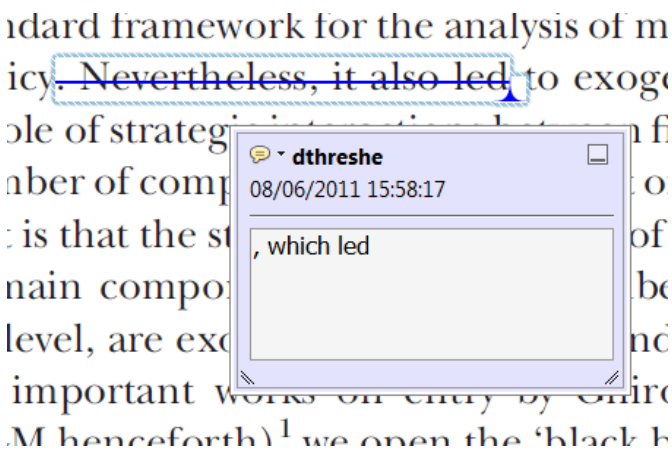
1. Replace (Ins) Tool – for replacing text.



Strikes a line through text and opens up a text box where replacement text can be entered.

How to use it

- Highlight a word or sentence.
- Click on the Replace (Ins) icon in the Annotations section.
- Type the replacement text into the blue box that appears.



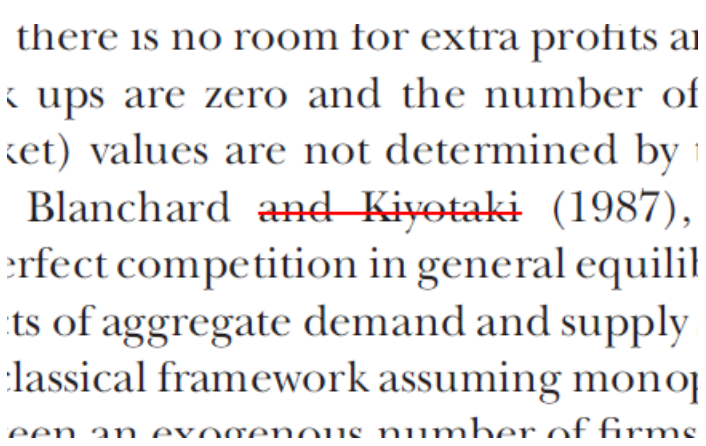
2. Strikethrough (Del) Tool – for deleting text.



Strikes a red line through text that is to be deleted.

How to use it

- Highlight a word or sentence.
- Click on the Strikethrough (Del) icon in the Annotations section.



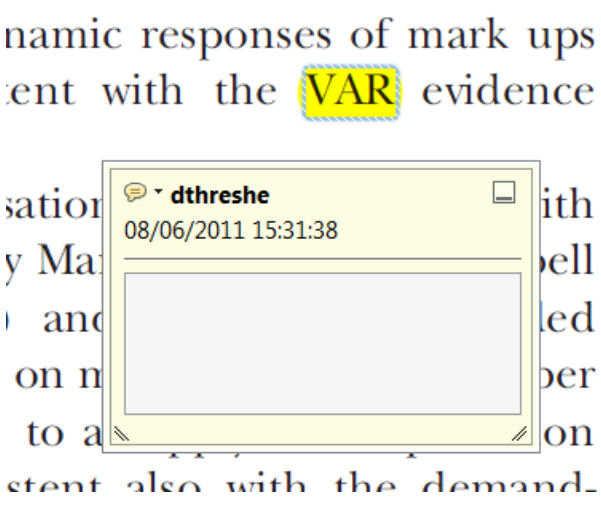
3. Add note to text Tool – for highlighting a section to be changed to bold or italic.



Highlights text in yellow and opens up a text box where comments can be entered.

How to use it

- Highlight the relevant section of text.
- Click on the Add note to text icon in the Annotations section.
- Type instruction on what should be changed regarding the text into the yellow box that appears.



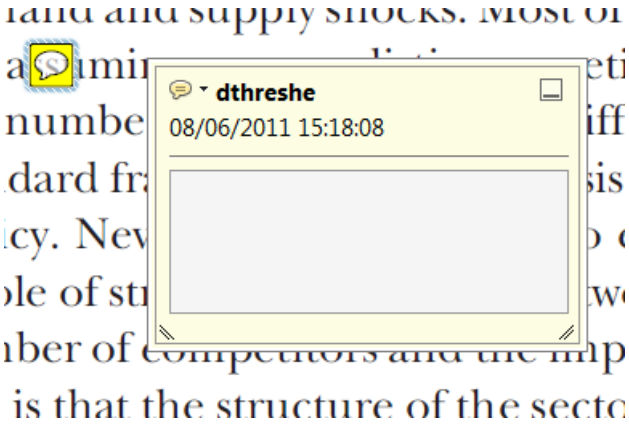
4. Add sticky note Tool – for making notes at specific points in the text.



Marks a point in the proof where a comment needs to be highlighted.

How to use it

- Click on the Add sticky note icon in the Annotations section.
- Click at the point in the proof where the comment should be inserted.
- Type the comment into the yellow box that appears.



WILEY-BLACKWELL

USING e-ANNOTATION TOOLS FOR ELECTRONIC PROOF CORRECTION

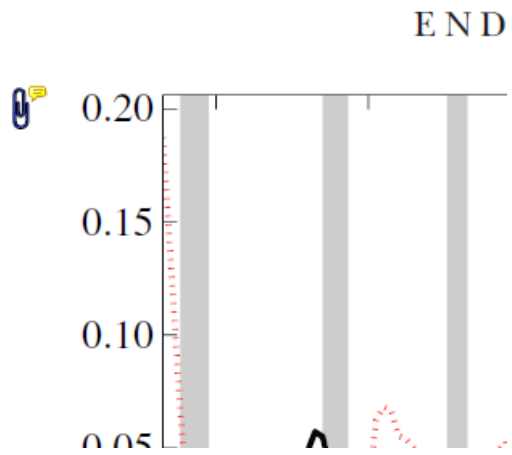
5. Attach File Tool – for inserting large amounts of text or replacement figures.



Inserts an icon linking to the attached file in the appropriate pace in the text.

How to use it

- Click on the Attach File icon in the Annotations section.
- Click on the proof to where you'd like the attached file to be linked.
- Select the file to be attached from your computer or network.
- Select the colour and type of icon that will appear in the proof. Click OK.



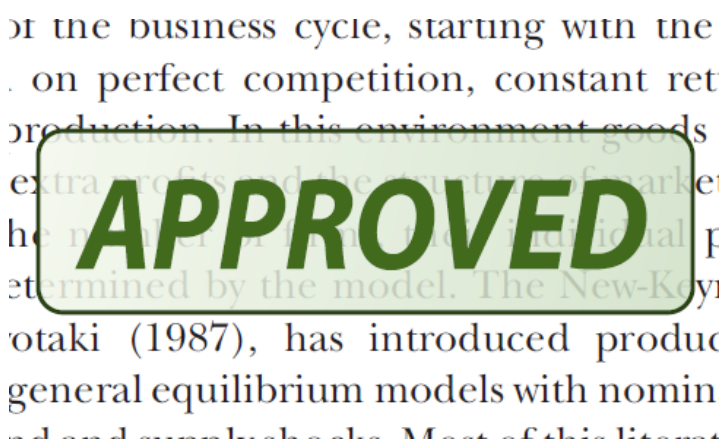
6. Add stamp Tool – for approving a proof if no corrections are required.

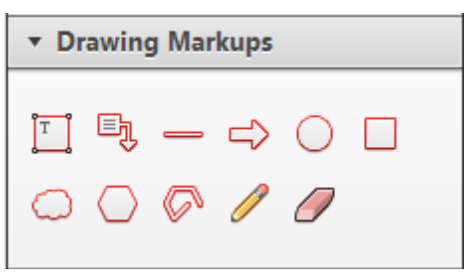


Inserts a selected stamp onto an appropriate place in the proof.

How to use it

- Click on the Add stamp icon in the Annotations section.
- Select the stamp you want to use. (The Approved stamp is usually available directly in the menu that appears).
- Click on the proof where you'd like the stamp to appear. (Where a proof is to be approved as it is, this would normally be on the first page).



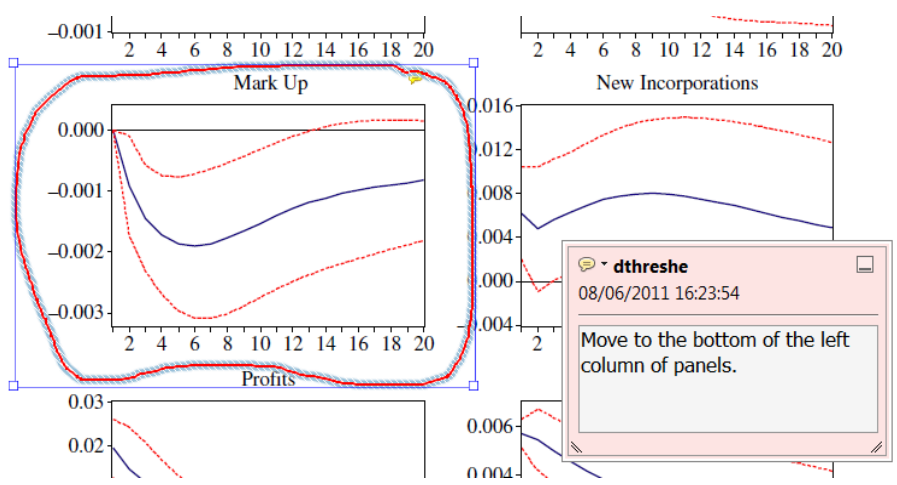


7. Drawing Markups Tools – for drawing shapes, lines and freeform annotations on proofs and commenting on these marks.

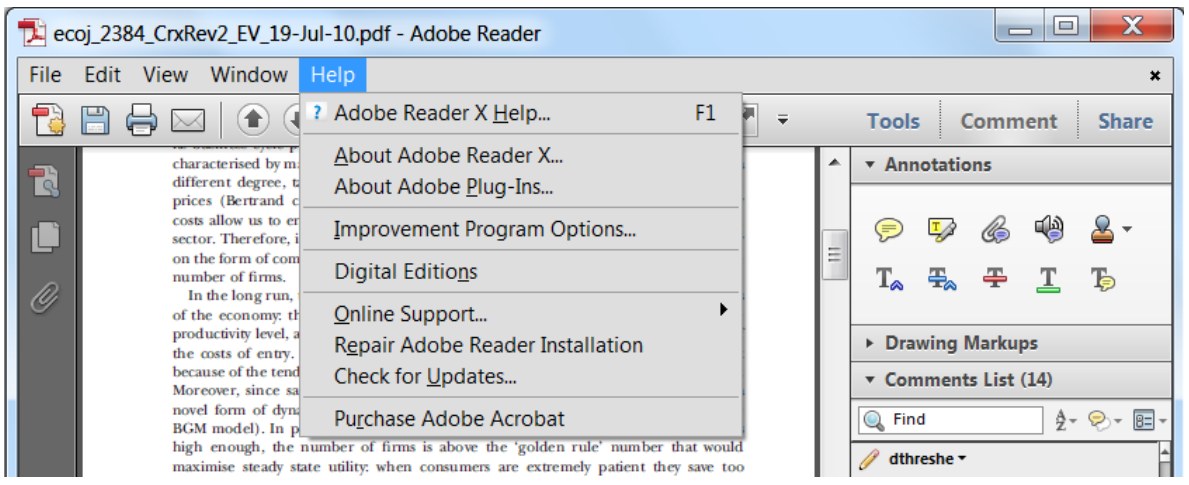
Allows shapes, lines and freeform annotations to be drawn on proofs and for comment to be made on these marks..

How to use it

- Click on one of the shapes in the Drawing Markups section.
- Click on the proof at the relevant point and draw the selected shape with the cursor.
- To add a comment to the drawn shape, move the cursor over the shape until an arrowhead appears.
- Double click on the shape and type any text in the red box that appears.



For further information on how to annotate proofs, click on the Help menu to reveal a list of further options:



WILEY AUTHOR DISCOUNT CLUB

We would like to show our appreciation to you, a highly valued contributor to Wiley's publications, by offering a **unique 25% discount** off the published price of any of our books*.

All you need to do is apply for the **Wiley Author Discount Card** by completing the attached form and returning it to us at the following address:

The Database Group (Author Club)
John Wiley & Sons Ltd
The Atrium
Southern Gate
Chichester
PO19 8SQ
UK

Alternatively, you can **register online** at www.wileyeurope.com/go/authordiscount
Please pass on details of this offer to any co-authors or fellow contributors.

After registering you will receive your Wiley Author Discount Card with a special promotion code, which you will need to quote whenever you order books direct from us.

The quickest way to order your books from us is via our European website at:

http://www.wileyeurope.com

Key benefits to using the site and ordering online include:
- Real-time SECURE on-line ordering
- Easy catalogue browsing
- Dedicated Author resource centre
- Opportunity to sign up for subject-orientated e-mail alerts

Alternatively, you can order direct through Customer Services at:
cs-books@wiley.co.uk, or call +44 (0)1243 843294, fax +44 (0)1243 843303

So take advantage of this great offer and return your completed form today.

Yours sincerely,



Verity Leaver
Group Marketing Manager
author@wiley.co.uk

UNCORRECTED PROOFS

*TERMS AND CONDITIONS

This offer is exclusive to Wiley Authors, Editors, Contributors and Editorial Board Members in acquiring books for their personal use. There must be no resale through any channel. The offer is subject to stock availability and cannot be applied retrospectively. This entitlement cannot be used in conjunction with any other special offer. Wiley reserves the right to amend the terms of the offer at any time.

REGISTRATION FORM
For Wiley Author Club Discount Card

To enjoy your 25% discount, tell us your areas of interest and you will receive relevant catalogues or leaflets from which to select your books. Please indicate your specific subject areas below.

Accounting [ ]
- Public [ ]
- Corporate [ ]

Chemistry [ ]
- Analytical [ ]
- Industrial/Safety [ ]
- Organic [ ]
- Inorganic [ ]
- Polymer [ ]
- Spectroscopy [ ]

Encyclopedia/Reference [ ]
- Business/Finance [ ]
- Life Sciences [ ]
- Medical Sciences [ ]
- Physical Sciences [ ]
- Technology [ ]

Earth & Environmental Science [ ]

Hospitality [ ]

Genetics [ ]
- Bioinformatics/ Computational Biology [ ]
- Proteomics [ ]
- Genomics [ ]
- Gene Mapping [ ]
- Clinical Genetics [ ]

Medical Science [ ]
- Cardiovascular [ ]
- Diabetes [ ]
- Endocrinology [ ]
- Imaging [ ]
- Obstetrics/Gynaecology [ ]
- Oncology [ ]
- Pharmacology [ ]
- Psychiatry [ ]

Non-Profit [ ]

Architecture [ ]

Business/Management [ ]

Computer Science [ ]
- Database/Data Warehouse [ ]
- Internet Business [ ]
- Networking [ ]
- Programming/Software Development [ ]
- Object Technology [ ]

Engineering [ ]
- Civil [ ]
- Communications Technology [ ]
- Electronic [ ]
- Environmental [ ]
- Industrial [ ]
- Mechanical [ ]

Finance/Investing [ ]
- Economics [ ]
- Institutional [ ]
- Personal Finance [ ]

Life Science [ ]

Landscape Architecture [ ]

Mathematics [ ]
Statistics [ ]

Manufacturing [ ]

Materials Science [ ]

Psychology [ ]
- Clinical [ ]
- Forensic [ ]
- Social & Personality [ ]
- Health & Sport [ ]
- Cognitive [ ]
- Organizational [ ]
- Developmental & Special Ed [ ]
- Child Welfare [ ]
- Self-Help [ ]

Physics/Physical Science [ ]

Please complete the next page /

UNCORRECTED PROOFS

I confirm that I am (*delete where not applicable):

a **Wiley** Book Author/Editor/Contributor* of the following book(s):
ISBN:
ISBN:

a **Wiley** Journal Editor/Contributor/Editorial Board Member* of the following journal(s):

SIGNATURE: …… Date: ……

PLEASE COMPLETE THE FOLLOWING DETAILS IN BLOCK CAPITALS:

TITLE: (e.g. Mr, Mrs, Dr) …… FULL NAME: ……

JOB TITLE (or Occupation): ……

DEPARTMENT: ……

COMPANY/INSTITUTION: ……

ADDRESS: ……

……

TOWN/CITY: ……

COUNTY/STATE: ……

COUNTRY: ……

POSTCODE/ZIP CODE: ……

DAYTIME TEL: ……

FAX: ……

E-MAIL: ……

YOUR PERSONAL DATA
We, John Wiley & Sons Ltd, will use the information you have provided to fulfil your request. In addition, we would like to:

1. Use your information to keep you informed by post of titles and offers of interest to you and available from us or other Wiley Group companies worldwide, and may supply your details to members of the Wiley Group for this purpose.
   [ ] Please tick the box if you do **NOT** wish to receive this information

2. Share your information with other carefully selected companies so that they may contact you by post with details of titles and offers that may be of interest to you.
   [ ] Please tick the box if you do **NOT** wish to receive this information.

E-MAIL ALERTING SERVICE
We also offer an alerting service to our author base via e-mail, with regular special offers and competitions. If you **DO** wish to receive these, please opt in by ticking the box [ ].

If, at any time, you wish to stop receiving information, please contact the Database Group (databasegroup@wiley.co.uk) at John Wiley & Sons Ltd, The Atrium, Southern Gate, Chichester, PO19 8SQ, UK.

TERMS & CONDITIONS
This offer is exclusive to Wiley Authors, Editors, Contributors and Editorial Board Members in acquiring books for their personal use. There should be no resale through any channel. The offer is subject to stock availability and may not be applied retrospectively. This entitlement cannot be used in conjunction with any other special offer. Wiley reserves the right to vary the terms of the offer at any time.

PLEASE RETURN THIS FORM TO:
Database Group (Author Club), John Wiley & Sons Ltd, The Atrium, Southern Gate, Chichester, PO19 8SQ, UK author@wiley.co.uk
Fax: +44 (0)1243 770154

UNCORRECTED PROOFS