SOLVING OPTIMIZATION-CONSTRAINED DIFFERENTIAL EQUATIONS WITH DISCONTINUITY POINTS, WITH APPLICATION TO ATMOSPHERIC CHEMISTRY

CHANTAL LANDRY*, ALEXANDRE CABOUSSAT†, AND ERNST HAIRER‡

Abstract. Ordinary differential equations are coupled with mixed constrained optimization problems when modeling the thermodynamic equilibrium of a system evolving with time. A particular application arises in the modeling of atmospheric particles. Discontinuity points are created by the activation/deactivation of inequality constraints. A numerical method for the solution of optimization-constrained differential equations is proposed by coupling an implicit Runge-Kutta method (RADAU5), with numerical techniques for the detection of the events (activation and deactivation of constraints). The computation of the events is based on dense output formulas, continuation techniques and geometric arguments. Numerical results are presented for the simulation of the time-dependent equilibrium of organic atmospheric aerosol particles, and show the efficiency and accuracy of the approach.

Key words. Initial value problems, Differential-algebraic equations, Constrained optimization, Runge-Kutta methods, Event detection, Discontinuity points, Computational chemistry.

AMS subject classifications. 65L05, 65L06, 65L80, 90C30, 80A30

1. Introduction.

The microscopic modeling of the dynamics and chemical composition of atmospheric aerosol particles is a crucial issue when trying to simulate the global climate forcing in three-dimensional air quality models [24]. The dynamic computation of the gas-particle partitioning and liquid-liquid equilibrium for organic particles introduces a coupling between the thermodynamic equilibrium of the particle and the interactions between the particle and the surrounding gas.

A mathematical model for the computation of the gas-particle partitioning and liquid-liquid equilibrium for organic atmospheric aerosol particles is presented. It couples a system of ordinary differential equations with a mixed constrained global optimization problem. A model problem can be written as follows: for $p, q > 0$, $T > 0$ and $\mathbf{b}_0$ given, find $\mathbf{b} : (0, T) \to \mathbb{R}^p$ and $\mathbf{x} : (0, T) \to \mathbb{R}^q$ satisfying

$$\begin{aligned} \frac{d}{dt}\mathbf{b}(t) &= \mathbf{f}(t, \mathbf{b}(t), \mathbf{x}(t)), \qquad \mathbf{b}(0) = \mathbf{b}_0 \\ \mathbf{x}(t) &= \arg\min_{\bar{\mathbf{x}}} \ \mathcal{G}(\bar{\mathbf{x}}) \\ &\qquad \text{s.t. } \mathbf{c}(\bar{\mathbf{x}}, \mathbf{b}(t)) = 0, \qquad \bar{\mathbf{x}} \geq \mathbf{0}. \end{aligned} \tag{1.1}$$

The first equation in (1.1) represents a stiff nonlinear system of ordinary differential equations where $\mathbf{f}$ is a smooth vector-valued function. The second part of (1.1) corresponds to a global minimization problem subjected to l equality constraints ($\mathbf{c} : \mathbb{R}^q \times \mathbb{R}^p \to \mathbb{R}^l$), and box constraints. The objective function $\mathcal{G}$ is non-convex, nonlinear and uniquely depends on $\mathbf{x}$. The equality constraints can be nonlinear functions.

*Institute of Analysis and Scientific Computing, Ecole Polytechnique Fédérale de Lausanne, Station 8, 1015 Lausanne, Switzerland (chantal.landry@a3.epfl.ch).

†Department of Mathematics, University of Houston, 4800 Calhoun Rd, Houston, Texas 77204-3008, USA (caboussat@math.uh.edu).

‡Section de mathématiques, Université de Genève, 1211 Genève, Switzerland (Ernst.Hairer@unige.ch).

The purpose of this article is to present an efficient numerical method that solves optimization-constrained differential equations like (1.1). The system (1.1) is such that as soon as an inequality constraint is activated or deactivated, the variable $\mathbf{x}$ is "truncated" and loses regularity. The numerical method has to accurately detect and compute the times of activation and deactivation of constraints in order to (i) compute the exact time of phase separation in the particle evolution and (ii) guarantee the accuracy of the numerical approximation of the solution of (1.1).

If the number of active inequality constraints is fixed, the considered system can be associated to a system of differential algebraic equations (DAE), by replacing the minimization problem by its first order optimality conditions. In that case, since the computation of the *global minimum of energy* is required, uniqueness is lost and the solutions may bifurcate between branches of global optima, local optima or saddle-points.

Efficient techniques to solve DAE systems relying on implicit Runge-Kutta methods have been developed in [4, 15, 16]. The determination of activation/deactivation times corresponds to the detection of a discontinuity in the variables $\mathbf{x}$, and requires techniques for *tracking of discontinuities*, or *event detection*. The activation/deactivation of constraints adds/removes algebraic equations from the DAE system.

A review of the detection of events in systems of ordinary differential equations or differential-algebraic equations can be found in [9]. Typically the event is determined by the zero of a state-dependent event function. Several procedures are based on the construction of interpolation polynomials which are used to approximate the event function and on the determination of a root of this approximation (see *e.g.*, [10, 14, 25]). Since the interpolation polynomials are in general less accurate than the solution approximation at the grid points, this procedure may lead to a loss of accuracy for the integration beyond this point. We follow a new strategy that exactly computes the discontinuity point, see [13]. It relies on the insertion of the fractional step size needed to reach the discontinuity as a variable in the set of equations. Following [11, 13], the proposed strategy is as follows:

1. Solution of the regular DAE system with a Runge-Kutta method;
2. Detection of discontinuity points (activation/deactivation of constraints);
3. Computation of the location and time of the discontinuity points;
4. Definition of the new DAE system and restart.

In Section 2, a mathematical model for the simulation of the dynamics of atmospheric particles is introduced, based on optimization-constrained differential equations. The geometric interpretation of the problem as the dynamic computation of the convex envelope of a non-convex function is detailed in Section 3. The numerical solution of the DAE system is presented in Section 4, while numerical methods for the tracking of discontinuities (activation and deactivation of inequality constraints) are detailed in Section 5. In Section 6, numerical results are presented for atmospheric organic particles to illustrate the efficiency and accuracy of the algorithm.

2. Optimization-Constrained Differential Equations. We are interested in the dynamics and chemical phase behavior of atmospheric aerosol particles. A single organic aerosol particle is considered and surrounded by a gas of same chemical composition. The internal composition of the particle satisfies the minimum of its internal energy, by enforcing *phase partitioning* between distinct liquid phases inside the particle [2]. Chemical reactions do not occur and temperature and pressure are

kept constant. The aim of the model is to accurately compute the time evolution of the particle's gas-particle partitioning and phase equilibrium.

Problem (1.1) can be seen as an ODE-constrained optimization problem with an objective function involving sup-norms for instance (see *e.g.* problems arising in control systems theory [26], or in PDE-constrained optimization [19]). The major difficulty resides in the fact that the underlying energy $\mathcal{G}$ is minimized for *a.e.* $t \in (0,T)$ along the trajectory. However, in order to emphasize that the problem is a time-evolutive problem under constraints and take advantage of its physical structure, it is more convenient to consider the optimization problem as a component of the definition of the fluxes of the ODE system.

Let $(0,T)$ be the interval of integration with $T > 0$. Let us denote by $\mathbf{b}(t) \in \mathbb{R}^s$ the composition vector of the s chemical components present in the particle at time $t \in (0,T)$. Let us denote by $p \leq s$ the maximal number of possible liquid phases arising at thermodynamic equilibrium [2] and define $\mathbf{x}_\alpha \in \mathbb{R}^s$ and $y_\alpha \in \mathbb{R}$, for $\alpha = 1,\dots,p$ as the mole-fraction vectors in phase α and the total number of moles in phase α respectively.

The mass transfer between the particle and the surrounding gas is modeled by ordinary differential equations, whereas the phase partitioning inside the particle results from the *global minimization of the Gibbs free energy* of the particle. Thus the problem is: find $\mathbf{b}, \mathbf{x}_\alpha : (0,T) \to \mathbb{R}^s_{++}$ and $y_\alpha : (0,T) \to \mathbb{R}_+$, $\alpha = 1,\dots,p$ satisfying:

$$\begin{aligned}
&\frac{d}{dt}\mathbf{b}(t) = \mathbf{f}(\mathbf{b}(t), \mathbf{x}_1(t), \dots, \mathbf{x}_p(t)), \qquad \mathbf{b}(0) = \mathbf{b}_0 \\
&\{\mathbf{x}_\alpha(t), y_\alpha(t)\}_{\alpha=1}^p = \underset{\{\bar{\mathbf{x}}_\alpha, \bar{y}_\alpha\}_{\alpha=1}^p}{\operatorname{argmin}} \sum_{\alpha=1}^p \bar{y}_\alpha\, g(\bar{\mathbf{x}}_\alpha) \\
&\text{s.t.} \quad \mathbf{e}^T\bar{\mathbf{x}}_\alpha = 1,\ \bar{\mathbf{x}}_\alpha > 0,\ \bar{y}_\alpha \geq 0,\ \alpha = 1,\dots,p, \quad \sum_{\alpha=1}^p \bar{y}_\alpha \bar{\mathbf{x}}_\alpha = \mathbf{b}(t),
\end{aligned} \tag{2.1}$$

where $\mathbf{b}_0$ is a given initial composition-vector and $\mathbf{e} = (1,\dots,1)^T$. The function $g \in \mathcal{C}^\infty(\mathbb{R}^s_{++})$ is the *molar Gibbs free energy* function [2] where $\mathbb{R}_{++}$ denotes the set of positive real numbers. The major property of g is to be a homogeneous function of degree one and satisfying $\lim_{x_i \to 0} \frac{\partial g}{\partial x_i} = -\infty$ and $\mathbf{x}^T \boldsymbol{\nabla} g(\mathbf{x}) = g(\mathbf{x})$, $\forall \mathbf{x} \in \mathbb{R}^s_{++}$. The vector-valued function $\mathbf{f}$ is the flux between the particle and the surrounding media and is a non-linear function of $\mathbf{b}$ and $\mathbf{x}_\alpha$. Actually, the flux $\mathbf{f}$ depends only on the variables $\mathbf{x}_\alpha$ for which the index α is such that $y_\alpha > 0$. It can be expressed as $\mathbf{f} = C(\mathbf{b}(t))\,(\mathbf{b}(t) - D\exp(\nabla g(\mathbf{x}_\alpha(t))))$, for any $\alpha \in \mathcal{A}$, where C is a function of $\mathbf{b}(t)$, D is a constant, both depending on the chemical properties of the aerosol particle. The chemical description of $\mathbf{f}$ can be found *e.g.* in [1, 24].

The first equality constraints in (2.1) are the normalization relations that follow from the definition of the mole-fraction vector $\mathbf{x}_\alpha$. The last equality constraint expresses the mass conservation among the liquid phases. The inequality constraints illustrate the non-negativity of the number of moles in the liquid phases. If $y_\alpha(t) > 0$, the liquid phase α is present at thermodynamic equilibrium in the particle at time t. Otherwise, if $y_\alpha(t) = 0$, the liquid phase α is not present at equilibrium.

System (2.1) couples ordinary differential equations and a mixed constrained global minimization problem, with a non-convex nonlinear objective function. The variables y_α and $\mathbf{x}_\alpha$ lose regularity when one variable $y_\alpha(t) > 0$ vanishes (*activation of an inequality constraint*) or, conversely, when one variable $y_\alpha(t) = 0$ becomes strictly positive (*deactivation of an inequality constraint*). The goal of this article is to present a numerical algorithm for the simulation of (1.1), and (2.1), with an accurate

determination of the activation/deactivation of inequality constraints.

3. Geometric Interpretation. A geometric interpretation of (2.1) is useful to understand the dynamics of the system and design efficient numerical techniques. First let us consider the optimization problem solely with a fixed point $\mathbf{b}$. If $\{y_\alpha, \mathbf{x}_\alpha\}_{\alpha=1}^p$ is the solution of the minimization problem for $\mathbf{b}$, then for any $c > 0$, $\{cy_\alpha, \mathbf{x}_\alpha\}_{\alpha=1}^p$ is the solution of the minimization problem for the point $c\mathbf{b}$. Therefore, without loss of generality, it is assumed that $\mathbf{e}^T\mathbf{b} = 1$ in this section. The hereafter interpretation follows [2] and starts with the projection of the optimization problem on a reduced space of lower dimension.

Let Δ'_s be defined by $\Delta'_s = \{\mathbf{x} \in \mathbb{R}^s | \mathbf{e}^T\mathbf{x} = 1, \mathbf{x} \geq 0\}$ and, for $r = s - 1$, $\Delta_r = \{\mathbf{z} \in \mathbb{R}^r | \mathbf{e}^T\mathbf{z} \leq 1, \mathbf{z} \geq 0\}$. The unit simplex Δ_r can be identified with Δ'_s via the mapping $\Pi : \Delta_r \to \Delta'_s$ such that $\mathbf{z} \to \mathbf{x} = \mathbf{e}_s + Z_e\mathbf{z}$, where $\mathbf{e}_s$ is the canonical basis vector and $Z_e^T = (\mathbf{I}_r, -\mathbf{e})$ with $\mathbf{I}_r$ the $r \times r$ identity matrix. Let $\tilde{g} = g \, o \, \Pi$. Then $\tilde{g}$ belongs to the function space E given by

$$E = \{\tilde{g} \in \mathcal{C}^\infty(\mathrm{int}\Delta_r) \,|\, \tilde{g} \in \mathcal{C}^0(\Delta_r), \, \partial\tilde{g}(\mathbf{z}) = \emptyset \text{ for } \mathbf{z} \in \partial\Delta_r\},$$

where $\partial\tilde{g}(\mathbf{z})$ represents the subdifferential of $\tilde{g}$ at $\mathbf{z}$.

Let P be the projection from $\mathbb{R}^s$ to $\mathbb{R}^r$ defined by $P(x_1, \ldots, x_r, x_s) = (x_1, \ldots, x_r)$, and denote $\mathbf{z}_\alpha = P\mathbf{x}_\alpha$ for $\alpha = 1, \ldots, p$, and $\mathbf{d} = P\mathbf{b}$. The minimization problem in (2.1) is equivalent after projection to

$$\begin{aligned} \min_{\{y_\alpha, \mathbf{z}_\alpha\}_{\alpha=1}^p} \quad & \sum_{\alpha=1}^p y_\alpha \tilde{g}(\mathbf{z}_\alpha), \\ \text{s.t.} \quad & y_\alpha \geq 0, \quad \alpha = 1, \ldots, p, \quad \sum_{\alpha=1}^p y_\alpha \mathbf{z}_\alpha = \mathbf{d}, \quad \sum_{\alpha=1}^p y_\alpha = 1. \end{aligned} \tag{3.1}$$

Since the domain of $\tilde{g}$ is Δ_r, the condition $\mathbf{z}_\alpha \in \Delta_r$ does not need to be included as constraint in (3.1). Problem (3.1) consists of the determination of the convex envelope of $\tilde{g}$ at point $\mathbf{d}$ [2]. The following result is a direct consequence of the Carathéodory's theorem.

THEOREM 3.1. *For every $\mathbf{d} \in \Delta_r$, the minimum of (3.1) is* $\mathrm{conv}\,\tilde{g}(\mathbf{d})$*, the value of the convex envelope of $\tilde{g}$ at $\mathbf{d}$. Moreover, one has* $\mathrm{conv}\tilde{g}(\mathbf{d}) = \sum_{\alpha=1}^p y_\alpha \tilde{g}(\mathbf{z}_\alpha)$ *for some convex combination* $\mathbf{d} = \sum_{\alpha=1}^p y_\alpha \mathbf{z}_\alpha$*,* $\sum_{\alpha=1}^p y_\alpha = 1$*,* $y_\alpha \geq 0$*,* $\alpha = 1, \ldots, p$*. The point* $(y_\alpha, \mathbf{z}_\alpha)_{\alpha=1,\ldots,p} \in \mathbb{R}^{(r+1)p}$ *is called a* phase splitting *of* $\mathbf{d}$*.*

A phase splitting is called *stable* if $y_\alpha > 0$ for all $\alpha = 1, \ldots, p$ and $\mathbf{z}_\alpha$ are distincts. Note that any phase splitting can be transformed into a stable phase splitting by considering the subset $\{\mathbf{z}_\alpha : y_\alpha > 0\}$. Let us define the sets of indices $\mathcal{A} = \{\alpha \in \{1, \ldots, p\} \mid y_\alpha = 0\}$ and $\mathcal{I} = \{\alpha \in \{1, \ldots, p\} \mid y_\alpha > 0\}$. The set $\mathcal{A}$ represents the set of indices of the active constraints, and $\mathcal{I}$ is the set of inactive constraints. Let $p^\mathcal{A}$, resp. $p^\mathcal{I}$, be the cardinal of $\mathcal{A}$, resp. $\mathcal{I}$, such that $p^\mathcal{A} + p^\mathcal{I} = p$. Hence $(y_\alpha^\mathcal{I}, \mathbf{z}_\alpha^\mathcal{I})_{\alpha\in\mathcal{I}}$ is a stable phase splitting of $\mathbf{d}$ if $(y_\alpha, \mathbf{z}_\alpha)_{\alpha=1,\ldots,p}$ is a phase splitting of $\mathbf{d}$.

REMARK 3.1. *In the sequel an exponent $\mathcal{I}$, resp. $\mathcal{A}$, is added to the variables y_α and $\mathbf{x}_\alpha$ to specify that $\alpha \in \mathcal{I}$, resp. $\mathcal{A}$. For instance, the expression $y_\alpha^\mathcal{I}$ stands for all y_α with $\alpha \in \mathcal{I}$. Moreover the notation $\alpha = 1, \ldots, p^\mathcal{I}$ is considered equivalent to $\forall\alpha \in \mathcal{I}$.*

The following result states the existence and uniqueness of the stable phase splitting for a given $\mathbf{d}$ and characterizes the geometrical structure of $\mathrm{conv}\,\tilde{g}(\mathbf{d})$. The proof of this result can be found in [22].

THEOREM 3.2. *There exists a residual set R of E such that for any function $\tilde{g} \in R$, every $\mathbf{d} \in \Delta_r$ has a unique stable phase splitting. More precisely, there exists a unique $(p^{\mathcal{I}}-1)$-simplex $\sum(\mathbf{d}) = \text{conv}\ (\mathbf{z}_1^{\mathcal{I}}, \ldots, \mathbf{z}_{p^{\mathcal{I}}}^{\mathcal{I}})$ with $p^{\mathcal{I}} \leq s$ such that $\text{conv}\, \tilde{g}(\mathbf{d}) = \sum_{\alpha=1}^{p^{\mathcal{I}}} y_\alpha^{\mathcal{I}} \tilde{g}(\mathbf{z}_\alpha^{\mathcal{I}})$ with the barycentric representation $\mathbf{d} = \sum_{\alpha \in \mathcal{I}} y_\alpha^{\mathcal{I}} \mathbf{z}_\alpha^{\mathcal{I}}$, $\sum_{\alpha \in \mathcal{I}} y_\alpha^{\mathcal{I}} = 1$ and $y_\alpha^{\mathcal{I}} > 0$, $\forall \alpha \in \mathcal{I}$.*

For a given $\mathbf{d} \in \text{int}\Delta_r$, the $(p^{\mathcal{I}}-1)$-simplex $\sum(\mathbf{d})$ is called the *phase simplex of* $\mathbf{d}$. The domain Δ_r can be separated in different areas according to the size of all possible phase simplexes, and is called a *phase diagram*.

The *Gibbs tangent plane criterion* (see *e.g.* [20]) states that a $(p^{\mathcal{I}}-1)$-simplex $\sum(\mathbf{d}) = \text{conv}\ (z_1^{\mathcal{I}}, \ldots, z_{p^{\mathcal{I}}}^{\mathcal{I}})$ is a phase simplex if and only if there exist multipliers $\eta \in \mathbb{R}^r$ and $\gamma \in \mathbb{R}$ such that

$$\boldsymbol{\nabla} \tilde{g}(\mathbf{z}_\alpha^{\mathcal{I}}) + \eta = \mathbf{0}, \quad \forall \alpha \in \mathcal{I}, \tag{3.2}$$

$$\tilde{g}(\mathbf{z}_\alpha^{\mathcal{I}}) + \eta^T \mathbf{z}_\alpha^{\mathcal{I}} + \gamma = 0, \quad \forall \alpha \in \mathcal{I}, \tag{3.3}$$

$$\tilde{g}(\mathbf{z}) + \eta^T \mathbf{z} + \gamma \geq 0, \quad \forall \mathbf{z} \in \Delta_r. \tag{3.4}$$

Geometrically, the affine hyperplane tangent to the graph of $\tilde{g}$ at $(\mathbf{z}_\alpha^{\mathcal{I}}, \tilde{g}(\mathbf{z}_\alpha^{\mathcal{I}}))$, $\forall \alpha \in \mathcal{I}$ lies entirely below the graph of $\tilde{g}$. This hyperplane is called the *supporting tangent plane*.

A point $\mathbf{d} \in \text{int}\ \Delta_r$ is said to be a *single-phase point* if and only if $\text{conv}\, \tilde{g}(\mathbf{d}) = \tilde{g}(\mathbf{d})$; the following result holds:

THEOREM 3.3. *Consider $\mathbf{d} \in \text{int}\Delta_r$ and $\sum(\mathbf{d}) = \text{conv}(\mathbf{z}_1^{\mathcal{I}}, \ldots, \mathbf{z}_{p^{\mathcal{I}}}^{\mathcal{I}})$ the phase simplex of $\mathbf{d}$. Then for all $\alpha \in \mathcal{I}$, $\mathbf{z}_\alpha^{\mathcal{I}} \in \text{int}\Delta_r$ and $\text{conv}\, \tilde{g}(\mathbf{z}_\alpha^{\mathcal{I}}) = \tilde{g}(\mathbf{z}_\alpha^{\mathcal{I}})$.*

The graph of g (and therefore of $\tilde{g}$) depends on the chemical components present in the aerosol, but is always composed of $r+1$ convex regions lying in the neighborhood of the vertices of Δ_r. For organic aerosols the maximum number of convex regions is equal to s. Let us consider the case of an aerosol made of 2 chemical components. Thereby $s = 2$, $r = 1$ and Δ_r is the interval $[0, 1]$. A generic representation of $\tilde{g}$ is given in Figure 3.1. For the points $\mathbf{d}$ considered on the left and right graphs, the value of the convex envelope of $\tilde{g}$ at point $\mathbf{d}$ is equal to the value of $\tilde{g}$ at point $\mathbf{d}$ and $\text{conv}\, \tilde{g}(\mathbf{d}) = \tilde{g}(\mathbf{z}_\alpha)$. This implies that the stable phase splitting of $\mathbf{d}$ is given by $(y, \mathbf{z})$ with $p^{\mathcal{I}} = 1$, $\mathbf{z} = \mathbf{d}$ and $y = 1$, and that $\mathbf{d}$ is a single-phase point.

On the central graph of Figure 3.1 the convex envelope of $\tilde{g}$ considered at points $\mathbf{d}$ is no longer superposed with $\tilde{g}$ but follows the segment given by $[\tilde{g}(\mathbf{z}_1), \tilde{g}(\mathbf{z}_2)]$. Hence the minimum of (3.1) is given by $\text{conv}\, \tilde{g}(\mathbf{d}) = y_1 \tilde{g}(\mathbf{z}_1) + y_2 \tilde{g}(\mathbf{z}_2)$, the stable phase splitting of $\mathbf{d}$ is $(y_1, \mathbf{z}_1, y_2, \mathbf{z}_2)$ with $p^{\mathcal{I}} = 2$ and $y_1 + y_2 = 1$, and the phase simplex of $\mathbf{d}$ is equal to $\text{conv}\ (\mathbf{z}_1, \mathbf{z}_2)$ where the vertices $\mathbf{z}_1$ and $\mathbf{z}_2$ are single-phase points.

Each single-phase point is associated to a convex region of $\tilde{g}$. We denote by $\Delta_{r,\alpha}$ the part of Δ_r that corresponds to the convex region of $\tilde{g}$ associated to $\mathbf{z}_\alpha$, and by $\Delta'_{s,\alpha}$ the image of $\Delta_{r,\alpha}$ through Π. The sizes of the convex regions of the energy function g cover many orders of magnitude (see [2])

In Figure 3.1 the supporting tangent plane is drawn for all considered $\mathbf{d}$. It can be observed that every hyperplane lies below the graph of $\tilde{g}$ as the Gibbs tangent plane criterion states. When $\mathbf{d}$ is a single-phase point, the tangent plane is in contact with $\tilde{g}$ at the point $(\mathbf{z}, \tilde{g}(\mathbf{z}))$ solely. When the phase simplex of $\mathbf{d}$ is given by $\text{conv}\ (\mathbf{z}_1, \mathbf{z}_2)$ the tangent plane touches $\tilde{g}$ at $(\mathbf{z}_1, \tilde{g}(\mathbf{z}_1))$ and $(\mathbf{z}_2, \tilde{g}(\mathbf{z}_2))$.

Let us consider the case where $\mathbf{b}$ (and therefore $\mathbf{d}$) evolves in time. The points $\mathbf{b}(t)$ are no longer supposed to be normalized. According to previous theory the points $\frac{1}{\mathbf{e}^T \mathbf{b}(t)} \mathbf{b}(t)$ lie in Δ_s and the points $\mathbf{d}(t)$ represent the projection of $\frac{1}{\mathbf{e}^T \mathbf{b}(t)} \mathbf{b}(t)$ onto the

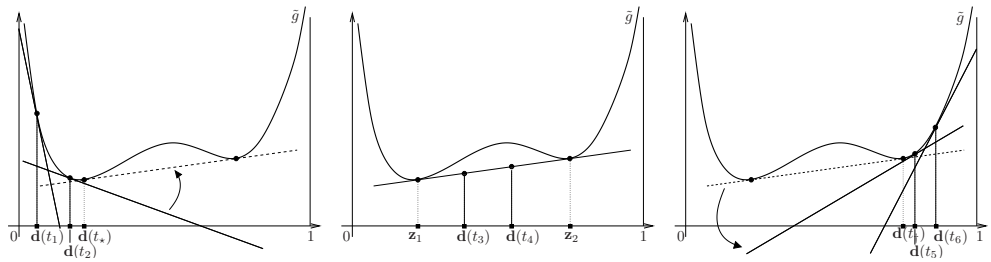

FIG. 3.1. *Geometric representation of the dynamic computation of the convex envelope. For a sequence of times $t_1 < t_2 < t_\star < t_3 < t_4 < t_\dagger < t_5 < t_6$, the vector $\mathbf{d}(t)$ moves from left to right. The supporting tangent plane follows the tangential slope at point $\mathbf{d}(t)$. Deactivation occurs at time $t_\star$ when the tangent plane (dashed line) touches the graph of $\tilde{g}$; Activation occurs at time $t_\dagger$ when the tangent plane (dashed line) gets released from the graph of $\tilde{g}$.*

simplex Δ_r. The time evolution of $\mathbf{b}$ is governed by the differential equation of (2.1) and requires the time-dependent computation of the stable phase simplex $\sum(\mathbf{d}(t))$. The activation/deactivation of constraints therefore corresponds to a change of dimension of the corresponding phase simplex $\sum(\mathbf{d}(t))$. In particular, the deactivation of a constraint can be interpreted as a *new tangential contact between the supporting tangent plane and the graph of the function $\tilde{g}$.*

Figure 3.1 shows the motion of the supporting tangent plane in one dimension of space, when the point $\mathbf{b}$ goes from left to right. When the tangent plane becomes in contact with the right convex region, one constraint is deactivated and the phase simplex' size increases by one ($p^{\mathcal{I}} = 1$ becomes $p^{\mathcal{I}} = 2$). Reciprocally, when the tangent plane leaves contact with the graph of $\tilde{g}$, the size of the phase simplex decreases by one ($p^{\mathcal{I}} = 2$ becomes $p^{\mathcal{I}} = 1$ again).

REMARK 3.2. *Even if we work with g and the variables $\mathbf{x}_\alpha$ and $\mathbf{b}$, it is more convenient to represent to projections $\tilde{g}$, $\mathbf{z}_\alpha$ and $\mathbf{d}$. For that reason the figures in the remainder of this article always illustrate $\tilde{g}$ and the projected variables $\mathbf{z}_\alpha$ and $\mathbf{d}$, but the notations g, $\mathbf{x}_\alpha$ and $\mathbf{b}$ are kept in the text and on the forthcoming figures.*

4. Numerical Method for Differential-Algebraic Equations. This section is devoted to the solution of (1.1), resp. (2.1), with a fixed number of active inequality constraints. In this case, the regularity of the variables $\mathbf{b}(t)$ and $\mathbf{x}(t)$ is guaranteed and one can prove the local existence and uniqueness of a continuously differentiable solution (following *e.g.* [23]).

In [6], (1.1) has been solved with a monolithic first order implicit Euler scheme. A fixed-point approach, together with a classical Crank-Nicolson scheme for the ordinary differential part has been used in [7], to obtain a second order accurate scheme. A new approach is presented here and based on the fifth-order accurate RADAU5 method [16], where discontinuities are treated using the ideas of [13]. In this way there is no loss in accuracy when passing through a discontinuity (cf. Section 5 below). For a robust and reliable simulation a certain accuracy is required, and experience shows that order two (as for Crank-Nicolson) is often too low. Our choice of an implicit Runge-Kutta method is further motivated by the fact that the differential equation is stiff. Explicit integrators would suffer from severe step size restrictions.

By replacing the minimization problem by its first order optimality conditions (Karush-Kuhn-Tucker (KKT) conditions), (1.1) becomes

$$\begin{aligned}
\frac{d}{dt}\mathbf{b}(t) &= \mathbf{f}(t,\mathbf{b}(t),\mathbf{x}(t)),\ \mathbf{b}(0)=\mathbf{b}_0,\\
\mathbf{0} &= \nabla\mathcal{G}(\mathbf{x}(t)) + \nabla_{\mathbf{x}}\mathbf{c}(\mathbf{x}(t),\mathbf{b}(t))\boldsymbol{\lambda}(t) - \boldsymbol{\theta}(t),\\
\mathbf{0} &= \mathbf{c}(\mathbf{x}(t),\mathbf{b}(t)),\\
0 &= x_i\theta_i,\ \forall i=1,\ldots,q,
\end{aligned} \tag{4.1}$$

together with $x_i \geq 0$, $\theta_i \geq 0$, $i=1,\ldots,q$, where $\boldsymbol{\lambda}(t)$ are the multipliers associated to the equality constraints, and $\boldsymbol{\theta}(t) = (\theta_1(t),\ldots,\theta_q(t))$ are the multipliers associated to the inequality constraints $\mathbf{x} \geq \mathbf{0}$. Let $\mathbf{Y}^T(t) = \Big(\mathbf{b}^T(t), \mathbf{x}^T(t), \boldsymbol{\lambda}^T(t), \boldsymbol{\theta}(t)\Big)$ be the N-vector, $N = p+q+l+q$ that contains all the unknowns of (4.1). When discarding the inequalities $x_i \geq 0$ and $\theta_i \geq 0$, the system (4.1) can be written as

$$\mathbf{M}\frac{d\mathbf{Y}}{dt}(t) = \mathbf{F}(\mathbf{Y}(t)), \quad \mathbf{M} = \begin{pmatrix} \mathbf{I} & \mathbf{0} \\ \mathbf{0} & 0 \end{pmatrix} \tag{4.2}$$

where the function $\mathbf{F}$ is the right hand side of (4.1) and $\mathbf{I}$ is the $p \times p$ identity matrix. Under the second order necessary conditions corresponding to the optimal problem in (1.1), the linear independence constraint qualification (LICQ), and the strict complementarity conditions [21], (4.2) is a DAE system of index 1 that is solvable.

When considering (2.1) in particular, this minimization problem consists of the computation of the convex envelope [2]. If a constraint $\bar{\alpha}$ is active (*i.e.* if $y_{\bar{\alpha}}(t) = 0$), then the variables $y_{\bar{\alpha}}$ and $\mathbf{x}_{\bar{\alpha}}$ are removed from the optimization algorithm without affecting the solution. This step is necessary to ensure that the DAE system remains solvable, of index 1 and similar to (4.2) [2]. When considering only the inactive constraints, (2.1) becomes:

$$\begin{aligned}
&\frac{d}{dt}\mathbf{b}(t) = \mathbf{f}(\mathbf{b}(t),\mathbf{x}_\alpha^{\mathcal{I}}(t)), \qquad \mathbf{b}(0)=\mathbf{b}_0,\\
&\{y_\alpha^{\mathcal{I}}(t), \mathbf{x}_\alpha^{\mathcal{I}}(t)\}_{\alpha\in\mathcal{I}(t)} = \arg\min_{\{\bar{y}_\alpha,\bar{\mathbf{x}}_\alpha\}_{\alpha\in\mathcal{I}(t)}} \sum_{\alpha\in\mathcal{I}(t)} \bar{y}_\alpha\, g(\bar{\mathbf{x}}_\alpha),\\
&\text{s.t.} \qquad \mathbf{e}^T\bar{\mathbf{x}}_\alpha = 1,\ \bar{\mathbf{x}}_\alpha > 0,\ \bar{y}_\alpha > 0,\ \alpha \in \mathcal{I}(t), \qquad \sum_{\alpha\in\mathcal{I}(t)} \bar{y}_\alpha\bar{\mathbf{x}}_\alpha = \mathbf{b}(t).
\end{aligned} \tag{4.3}$$

The solution of (2.1) is then equivalent to the solution of (4.3), together with $y_\alpha(t) = 0, \forall\alpha \in \mathcal{A}(t)$. The particularity is that the variables $\mathbf{x}_\alpha^{\mathcal{A}}$ do not appear in (4.3) and therefore are not updated in the computation of the convex envelope. The sole condition on $\mathbf{x}_\alpha^{\mathcal{A}}$ is the normalization constraint $\mathbf{e}^T\mathbf{x}_\alpha^{\mathcal{A}} = 1$.

Let $\boldsymbol{\lambda} \in \mathbb{R}^s$ and $\zeta_\alpha \in \mathbb{R}$, $\alpha \in \mathcal{I}(t)$ be the Lagrangian multipliers associated to the equality constraints in (4.3). We replace the minimization problem by its first order optimality (KKT) conditions. By using the homogeneity property of g, one can show that the variable ζ_α equals to 0 when $\alpha \in \mathcal{I}(t)$. With some algebra, (4.3) becomes

$$\begin{aligned}
\frac{d}{dt}\mathbf{b}(t) &= \mathbf{f}(\mathbf{b}(t),\mathbf{x}_\alpha^{\mathcal{I}}(t)), && \mathbf{b}(0)=\mathbf{b}_0,\\
\mathbf{0} &= \boldsymbol{\nabla} g(\mathbf{x}_\alpha(t)) + \boldsymbol{\lambda}(t), && \alpha\in\mathcal{I}(t),\\
0 &= \mathbf{e}^T\mathbf{x}_\alpha(t) - 1, && \alpha\in\mathcal{I}(t),\\
\mathbf{0} &= \sum_{\alpha\in\mathcal{I}(t)} y_\alpha(t)\,\mathbf{x}_\alpha(t) - \mathbf{b}(t).
\end{aligned} \tag{4.4}$$

The second equation means that the gradient of g at the points $\mathbf{x}_\alpha$, $\alpha \in \mathcal{I}$, is always equal to $-\boldsymbol{\lambda}$ and consequently $\boldsymbol{\nabla} g(\mathbf{x}_\alpha) = \boldsymbol{\nabla} g(\mathbf{x}_\beta)$, $\forall \alpha, \beta \in \mathcal{I}$.

Replacing a non-convex optimization problem by its first order optimality conditions does not necessarily guarantee the global optimality of the solution. For the particular case of (2.1), sufficient conditions to obtain a global minimum to the *pointwise* optimization problem have been given in [2] and in the references therein. As long as the number of active constraints remains constant, the optimum at each time t is in a neighborhood of the solution at another time in the near future. Thus it is a good initial guess for any Newton method. By continuation, the model is therefore able to track a branch of global minima provided that the trajectory started with the global minimum at time $t = 0$.

Let $\mathbf{Y}^T(t) = \Big(\mathbf{b}^T(t), \mathbf{x}_1^{\mathcal{I},T}(t), \ldots, \mathbf{x}_{p^\mathcal{I}}^{\mathcal{I},T}(t), y_1^\mathcal{I}(t), \ldots, y_{p^\mathcal{I}}^\mathcal{I}(t), \boldsymbol{\lambda}^T(t)\Big)$ be a N-vector, $N = s + sp^\mathcal{I} + p^\mathcal{I} + s$, that contains all the unknowns of (4.4). The system (4.4) can therefore be written again as (4.2), where the function $\mathbf{F}$ is the right hand side of (4.4) and the matrix $\mathbf{I}$ is the $s \times s$ identity matrix.

The system (4.2) is completed by the initial condition $\mathbf{Y}(0) = \mathbf{Y}^0$. The first s components of $\mathbf{Y}^0$ (related to the variable $\mathbf{b}$) are given by the initial condition $\mathbf{b}_0$ in (4.4). The initial value of the (algebraic) variables $\mathbf{x}_\alpha, y_\alpha$ and $\boldsymbol{\lambda}$ must satisfy the usual *consistency conditions*, and are obtained as the solution of the minimization problem in (2.1) for a given concentration-vector $\mathbf{b}_0$. As proposed in [2], there is a consistent solution, that is obtained with a primal-dual interior-point method.

The system (4.2) is a system of differential-algebraic equations of index one, that couples the differential variable $\mathbf{b}$ and the algebraic variables $(\mathbf{x}_\alpha^\mathcal{I}, y_\alpha^\mathcal{I}, \boldsymbol{\lambda})$. Such systems are widely studied in the literature (see *e.g.* [5, 13, 15, 16]). A 3-stage implicit Runge-Kutta method RADAU5 of order 5 [15, 16] is used here for the solution of (4.2).

Let $\mathbf{b}^n$, $\mathbf{x}_\alpha^n$, y_α^n, $\boldsymbol{\lambda}^n$ and $\mathbf{Y}^n$ be approximations of $\mathbf{b}(t_n)$, $\mathbf{x}_\alpha(t_n)$, $y_\alpha(t_n)$, $\boldsymbol{\lambda}(t_n)$ and $\mathbf{Y}(t_n)$, respectively, at time t_n. With the notations of (4.2), a q-stage implicit Runge-Kutta method is defined by

$$\mathbf{M}(\mathbf{Z}^i - \mathbf{Y}^n) = h_n \sum_{j=1}^{q} a_{ij}\, \mathbf{F}(\mathbf{Z}^j), \quad i = 1, \ldots, q \tag{4.5}$$

$$\mathbf{M}(\mathbf{Y}^{n+1} - \mathbf{Y}^n) = h_n \sum_{j=1}^{q} c_j\, \mathbf{F}(\mathbf{Z}^j), \tag{4.6}$$

where $\{a_{ij}\}$ and $\{c_j\}$ are given prescribed coefficients, and $h_n = t_{n+1} - t_n$. For stiffly accurate methods such as RADAU5, $c_i = a_{qi}$ for $i = 1, \ldots, q$. The numerical solution of (4.5)-(4.6) is then given by $\mathbf{Y}^{n+1} = \mathbf{Z}^q$ at each time step. Relation (4.5) forms a nonlinear system of equations for the internal stages values $\mathbf{Z}^i$, $i = 1, \ldots, q$. Details concerning the implementation of RADAU5 methods can be found in [15, 16]. At each time step, the initialization of the Newton method for the solution of (4.5)-(4.6) with the global optimum at the previous time step encourages the computation of a branch of global optima, as long as the number of active constraints does not change.

Since this Runge-Kutta method is a collocation method, it provides a cheap numerical approximation to $\mathbf{Y}(t_n + \theta h_n)$ for the whole integration interval $0 \le \theta \le 1$. The *dense output approximation* (collocation polynomial) computed at the nth step t_n is denoted by $\mathbf{U}^n(t_n + \theta h_n)$. The collocation method based on Radau points is of order $2q-1$, and the dense output of order q. The error $\mathbf{U}^n(t_n + \theta h_n) = \mathbf{Y}(t_n + \theta h_n)$ is therefore composed of the global error at t_n plus the local error contribution which

is bounded by $\mathcal{O}((h_n)^{q+1})$. In the sequel, the dense output formula for specific components of $\mathbf{Y}$ are used and the corresponding component is specified by its index. For instance, the dense output for the variables $y_{\bar{\alpha}}$ at t_n is denoted by $\mathbf{U}^n_{y_{\bar{\alpha}}}(t_n + \theta h_n)$ for $\theta \in [0, 1]$.

As soon as the set of inactive constraints is fixed, the RADAU5 algorithm is used. It yields the full order of accuracy (here, order 5) as long as the solution is sufficiently regular. To guarantee this regularity, the step sizes are chosen carefully, so that instants of discontinuity exactly coincide with points of the grid. An algorithmic realization is presented in Section 5. The coupling of this algorithm with an efficient procedure to compute any change in the set of inactive constraints allows to track the activation/deactivation of constraints that correspond to discontinuity points. It also allows to avoid the bifurcation between branches of local and global minima that may arise when the activation or deactivation of a constraint is not accurately computed.

5. Tracking of Discontinuity Points. When an inequality constraint is activated or deactivated, the variables $y^{\mathcal{I}}_\alpha$ and $\mathbf{x}^{\mathcal{I}}_\alpha$ can lose their regularity (typically when $y^{\mathcal{I}}_\alpha$ is truncated to zero, its first derivative is discontinuous at the truncation point). These discontinuity points have to be detected with accuracy [10, 11, 13, 14], although the time at which the discontinuities occurs is not known in advance.

Following [11, 13], methods for the tracking of discontinuity points consist of two steps: (i) the detection of the time interval $[t_n, t_{n+1}]$ that contains the event; (ii) the accurate computation of the event time. This two-steps procedure applied to (1.1), and (2.1) resp., is detailed in the next sections.

5.1. Detection of Discontinuity Points. At each time step t_n, the detection of the activation/deactivation of a constraint is achieved by checking on the sign of a particular quantity. In the sequel, the cases of the activation or the deactivation of a constraint are distinguished.

The Case of the Activation of an Inequality Constraint. This case corresponds to the determination of the minimal time for the transition $x_i > 0 \to x_i = 0$ in (1.1). When the number of active constraints is fixed and (4.2) is solved with the RADAU5 method, the positiveness constraints on the variables $\mathbf{x}$ is temporarily relaxed. The criterion to detect the presence of the activation of an inequality constraint is therefore to check at each time step t_{n+1} if there exists an index $i = 1, \ldots, q$ such that $x^n_i > 0$ and $x^{n+1}_i \leq 0$.

For the particular case of (2.1), the activation of an inequality constraint corresponds to the minimal time t (*discontinuity time*) such that the transition $y_\alpha(t) > 0 \to y_\alpha(t) = 0$ occurs. When the number of active constraints is fixed, the variables y_α may take negative values (which is a nonsense from a chemical point of view since the quantity y_α represents a number of moles). The criterion to detect the presence of the activation of an inequality constraint is therefore to check at each time step t_{n+1} if

$$\exists\, \bar{\alpha} \in \mathcal{I}(t_{n+1}) \quad \text{such that} \quad y^n_{\bar{\alpha}} > 0 \quad \text{and} \quad y^{n+1}_{\bar{\alpha}} < 0. \tag{5.1}$$

In that case, there exists a time $\tau \in (t_n, t_{n+1})$ for which the inequality constraint is activated. Results about the RADAU5 method [13] ensures that activation of constraints are not missed.

The Case of the Deactivation of an Inequality Constraint. This case corresponds to the determination of the minimal time for the transition $\mathbf{x}_i = 0 \to \mathbf{x}_i >$

0 in (1.1). By strict complementarity condition, this is equivalent to working with the dual variables θ_i, and looking for the minimal time for the transition $\theta_i > 0 \to \theta_i = 0$. The criterion to detect the presence of the deactivation of an inequality constraint is therefore to check at each time step t_{n+1} if there exists an index $i = 1, \ldots, q$ such that $\theta_i^n > 0$ and $\theta_i^{n+1} \leq 0$.

For our particular problem, the variables θ_i do not appear explicitly. A deactivation occurs when there exists an index $\bar{\alpha} \in \mathcal{A}$ such that $y_{\bar{\alpha}}(t) = 0 \to y_{\bar{\alpha}}(t) > 0$. However, the variables $y_{\bar{\alpha}}$ and $\mathbf{x}_{\bar{\alpha}}$, $\bar{\alpha} \in \mathcal{A}$ do not appear in (4.4) or (4.2) (the only condition on $\mathbf{x}_{\bar{\alpha}}$ is the normalization condition $\mathbf{e}^T\mathbf{x}_{\bar{\alpha}} = 1$). The criterion to "add" such variables into (4.2) for the next time step is therefore independent of the solution of the differential-algebraic system at the previous time step.

As described in Section 3 and illustrated in Figure 5.1 (left), the deactivation of a constraint occurs when the supporting tangent plane to the energy function g becomes tangent to a new point on the graph of the function. The point $\mathbf{b}$ considered in Figure 5.1 (left) is a single-phase point and the supporting tangent plane lies below the graph. Suppose that $\mathbf{b}$ moves to the right until the area where both inequality constraints are deactivated and the deactivation of the second constraint does not occur. In that case $\mathbf{b}$ remains a single-phase point and the supporting tangent plane is still defined by $(\mathbf{b}, g(\mathbf{b}))$. Such a situation is represented in Figure 5.1 (right). The tangent plane crosses the graph of g in that case and the Gibbs tangent plane criterion is therefore not satisfied. This fact is the indicator for the deactivation of an inequality constraint.

Since the function g is known only point-wise, the intersection between the supporting tangent plane and the graph of g cannot be computed analytically. However, it is not necessary to compute this intersection, but only to find one point $(\mathbf{x}, g(\mathbf{x}))$ situated below the tangent plane. Let us sign the distance between $(\mathbf{x}, g(\mathbf{x}))$ and the supporting tangent plane in such a way that the distance is said to be positive if $(\mathbf{x}, g(\mathbf{x}))$ lies above the tangent plane, and negative if $(\mathbf{x}, g(\mathbf{x}))$ is below the tangent plane. The points for which the distance can be negative are situated in the convex areas associated to the active constraints $\Delta'_{s,\alpha}$, $\alpha \in \mathcal{A}$. Since there is no condition on $\mathbf{x}_\alpha^{\mathcal{A}}$ except $\mathbf{e}^T\mathbf{x}_\alpha^{\mathcal{A}} - 1 = 0$, let us define $\mathbf{x}_\alpha^{\mathcal{A}}$ such that $(\mathbf{x}_\alpha^{\mathcal{A}}, g(\mathbf{x}_\alpha^{\mathcal{A}}))$ is situated at minimal distance from the supporting tangent plane. If we denote by $d^n(\mathbf{x})$ the signed distance between $(\mathbf{x}, g(\mathbf{x}))$ and the supporting tangent plane at time t_n, then the criterion to detect the presence of the deactivation of an inequality constraint is to check at each time step t_{n+1} if

$$\exists\, \bar{\alpha} \in \mathcal{A}(t_{n+1}) \;\text{ such that }\; d^n(\mathbf{x}_{\bar{\alpha}}^n) > 0 \;\text{ and }\; d^{n+1}(\mathbf{x}_{\bar{\alpha}}^{n+1}) < 0, \tag{5.2}$$

where $\mathbf{x}_{\bar{\alpha}}^n$, $\mathbf{x}_{\bar{\alpha}}^{n+1} \in \Delta'_{s,\bar{\alpha}}$ are the points that respectively minimize $d^n(\cdot)$ and $d^{n+1}(\cdot)$ in the convex area $\Delta'_{s,\bar{\alpha}}$. This distance corresponds to the dual variable $\boldsymbol{\theta}$ in the generic problem (4.1). In that case, there exists a time $\tau \in (t_n, t_{n+1})$ for which the inequality constraint is deactivated.

While results about the RADAU5 method [13] ensure that the activations of constraints are not missed, the detection of the deactivation of constraints relies on an external, *dual*, argument, and there is no theoretical result that provides such a guarantee. For a given supporting tangent plane, the algorithm presented in Section 5.2 allows to determine the signed distance, together with the point $\mathbf{x}_\alpha^{\mathcal{A}}$ that satisfies the minimal distance. The accurate computation of the signed distance, and therefore of the criterion (5.2) is actually the only lack of complete robustness and reliability in the algorithm. However, numerical experiments will show that this particular point

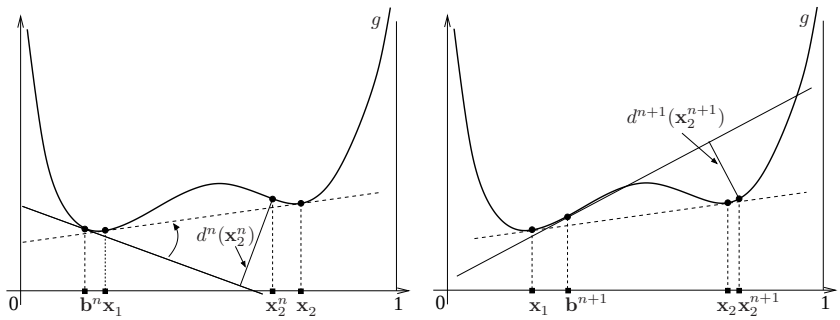

FIG. 5.1. *Deactivation of an inequality constraint: minimal distance criterion. Left: the tangent plane of g at $(\mathbf{b}^n, g(\mathbf{b}^n))$ lies under g; Right: without detection of a deactivation of a constraint, the tangent plane crosses the curve g.*

can be controlled.

5.2. Computation of the Minimal Distance Criterion. Let us determine first the equation describing the supporting tangent plane and the distance between the plane and any points $(\mathbf{x}, g(\mathbf{x}))$, $\mathbf{x} \in \mathbb{R}^s_{++}$. As described in Section 3 the supporting tangent plane is the affine hyperplane tangent to the graph of g at the points $(\mathbf{x}_\alpha, g(\mathbf{x}_\alpha))$, $\alpha \in \mathcal{I}$. Since $\boldsymbol{\nabla} g(\mathbf{x}_\alpha) = \boldsymbol{\nabla} g(\mathbf{x}_\beta)$, $\forall \alpha, \beta \in \mathcal{I}$, the normal vector to the tangent plane is uniquely determined. The supporting tangent plane is then defined by the set of points $(\mathbf{x}, x_{s+1}) \in \mathbb{R}^s_+ \times \mathbb{R}$ satisfying

$$\boldsymbol{\nabla} g(\mathbf{x}_\alpha)^T(\mathbf{x}_\alpha - \mathbf{x}) + x_{s+1} - g(\mathbf{x}_\alpha) = 0,$$

where $\mathbf{x}_\alpha$ is any point for which $\alpha \in \mathcal{I}$.

Since $\boldsymbol{\nabla} g(\mathbf{x})^T\mathbf{x} = g(\mathbf{x})$, $\forall \mathbf{x} \in \mathbb{R}^s_{++}$ the definition of the hyperplane is reduced to $-\boldsymbol{\nabla} g(\mathbf{x}_\alpha)^T\mathbf{x} + x_{s+1} = 0$. The vector $\mathbf{x}_\alpha$ being solution of the differential-algebraic system (4.4), the relation $\boldsymbol{\lambda} = -\boldsymbol{\nabla} g(\mathbf{x}_\alpha)$ holds and the above equation becomes $\boldsymbol{\lambda}^T\mathbf{x} + x_{s+1} = 0$. The signed distance of any point $(\mathbf{x}, g(\mathbf{x}))$ to the tangent plane is thus given by $d(\mathbf{x}) = (\boldsymbol{\lambda}^T\mathbf{x} + g(\mathbf{x}))/\|\mathbf{n}\|_2$, where $\mathbf{n} = \left(-\boldsymbol{\lambda}^T, -1\right)^T$. We consider in practice the signed distance, again denoted by d, defined by $d(\mathbf{x}) = \boldsymbol{\lambda}^T\mathbf{x} + g(\mathbf{x})$.

Hence at each time step t_{n+1} of the time discretization algorithm, and for all active constraints $\alpha \in \mathcal{A}$, the computation of the point $\mathbf{x}_\alpha^{\mathcal{A},n+1} \in \Delta'_{s,\alpha}$ situated at minimal distance from the tangent plane is given by the solution of the following minimization problem

$$\mathbf{x}_\alpha^{\mathcal{A},n+1} = \arg\min_{\mathbf{x}\in\Delta'_{s,\alpha}} d(\mathbf{x}) = \arg\min_{\mathbf{x}\in\Delta'_{s,\alpha}} \boldsymbol{\lambda}^{n+1,T}\mathbf{x} + g(\mathbf{x}), \tag{5.3}$$

where $\boldsymbol{\lambda}^{n+1}$ is solution of the system (4.4) at time t_{n+1}.

The distance function d possesses several local minima. Each $\mathbf{x}_\alpha^{\mathcal{I}}$ realizes a local minimum such that $d(\mathbf{x}_\alpha^{\mathcal{I}}) = 0$, while $\mathbf{x}_\alpha^{\mathcal{A}}$ realizes a local minimum in $\Delta'_{s,\alpha}$. The determination of $\mathbf{x}_\alpha^{\mathcal{A}}$ corresponds to finding the point located in $\Delta'_{s,\alpha}$ that realizes the local minima of the distance function. The value of the objective function $d(\mathbf{x}_\alpha^{\mathcal{A}})$ indicates if deactivation occurs.

In the minimization problem (5.3) we search for $\mathbf{x}$ in $\Delta'_{s,\alpha}$. One way to characterize $\Delta'_{s,\alpha}$ is to impose a constraint to (5.3) that expresses the positive-definiteness

of the Hessian matrix $\boldsymbol{\nabla}^2 g(\mathbf{x})$. We consider here the minimization problem where the sole constraint on $\mathbf{x}$ is $\mathbf{e}^T\mathbf{x} - 1 = 0$ and the constraint $\mathbf{x} \in \Delta_{s,\alpha}^{'}$ is imposed weakly. This relaxed problem is defined as follows

$$\mathbf{x}_\alpha^{\mathcal{A},n+1} = \arg\min_{\mathbf{x}\in\mathbb{R}^s} \boldsymbol{\lambda}^{n+1,T}\mathbf{x} + g(\mathbf{x}), \quad \text{s.t.} \quad \mathbf{e}^T\mathbf{x} - 1 = 0. \tag{5.4}$$

The KKT conditions relative to (5.4) lead to the nonlinear system:

$$\boldsymbol{\nabla} g(\mathbf{x}) + \boldsymbol{\lambda}^{n+1} + \zeta\mathbf{e} = \mathbf{0}, \qquad \mathbf{e}^T\mathbf{x} - 1 = 0, \tag{5.5}$$

where $\zeta \in \mathbb{R}$ is a Lagrangian multiplier associated to the equality constraint $\mathbf{e}^T\mathbf{x} - 1$. The unknowns are $\mathbf{x}$ and ζ, and the size of (5.5) is $s+1$, which is small by opposition to the optimization problem arising in (4.3). However the small nonlinear system (5.5) has to be solved at each time step and for all $\alpha \in \mathcal{A}$.

Problem (5.5) is solved with a Newton method and the corresponding Newton system reads:

$$\begin{pmatrix} \boldsymbol{\nabla}^2 g(\mathbf{x}) & \mathbf{e} \\ \mathbf{e}^T & 0 \end{pmatrix} \begin{pmatrix} \mathbf{p}_\mathbf{x} \\ p_\zeta \end{pmatrix} = -\begin{pmatrix} \boldsymbol{\nabla} g(\mathbf{x}) + \boldsymbol{\lambda}^{n+1} + \zeta\mathbf{e} \\ \mathbf{e}^T\mathbf{x} - 1 \end{pmatrix}, \tag{5.6}$$

where $\mathbf{p}_\mathbf{x}$ and p_ζ are the increments corresponding to the variables $\mathbf{x}$ and ζ.

LEMMA 5.1. *If $\mathbf{x}$ belongs to a convex region of g, (5.6) is solvable.*

Proof. If $\mathbf{x}$ remains in a convex region of g, $\boldsymbol{\nabla}^2 g(\mathbf{x})$ is symmetric positive definite and the inertia theorem (see *e.g.* [12]) allows to conclude that the matrix of (5.6) is invertible. □

Following Lemma 5.1, the numerical algorithm for the solution of (5.6) must pay attention to building a sequence of iterates that remains in the convex region $\Delta_{s,\alpha}^{'}$. The initial guess of the Newton method is given either in a neighborhood of the vertices of the simplex $\Delta_s^{'}$ (as the initial guesses of the interior-point method described in [2]), or by the last iterate obtained in the convex region at the previous time step.

For each iterate $\mathbf{x}^i$ of the Newton sequence, the sequence is re-initialized at $\mathbf{x}^{i-1}$ if the Hessian $\boldsymbol{\nabla}^2 g(\mathbf{x}^i)$ is not positive definite or if the point $\mathbf{x}^i$ goes out of the simplex. The Newton increments are controled with a step-size algorithm in order to ensure that the iterates remain in the convex region $\Delta_{s,\alpha}^{'}$ and that the Hessian remains positive definite. More precisely, let c_{thres} be a given threshold that corresponds to an approximation of the distance between convex regions; if $\|(\mathbf{p}_\mathbf{x}, p_\zeta)^T\|_2 \geq c_{\text{thres}}$, then the Newton iterates are computed as

$$\begin{pmatrix} \mathbf{x}^{i+1} \\ \zeta^{i+1} \end{pmatrix} = \begin{pmatrix} \mathbf{x}^i \\ \zeta^i \end{pmatrix} + \alpha_i \begin{pmatrix} \mathbf{p}_\mathbf{x} \\ p_\zeta \end{pmatrix}, \quad \alpha_i = \frac{c_{\text{thres}}}{\|(\mathbf{p}_\mathbf{x}, p_\zeta)^T\|_2} \in (0,1]; \tag{5.7}$$

otherwise the new iterate of Newton $(\mathbf{x}^{i+1}, \zeta^{i+1})^T$ is computed with $\alpha_i = 1$. This procedure is only activated when $\det(\boldsymbol{\nabla}^2 g(\mathbf{x}^i)) \leq \delta$, where $\delta \simeq 10^0 - 10^2$ captures "flat" regions (by comparison, $\det(\boldsymbol{\nabla}^2 g(\mathbf{x}^i))$ can be as big as 10^{15} for such systems). The points $\mathbf{x}^i$ lying in the simplex $\Delta_s^{'}$, the parameter c_{thres} is initialized to 0.1. Since the distance between the convex areas could be smaller than 0.1, the value of c_{thres} can be updated at each time step by computing the minimal distance between all $\mathbf{x}_\alpha$, $\alpha = 1, \ldots, p$.

Figure 5.2 illustrates the influence of the modification of the increments given by (5.7) for the case $r = 1$. A 2-components chemical system composed of 1-hexacosanol and pinic acid is considered. The simplex Δ_1 is the segment $[0,1]$ (0 meaning 100%

of pinic acid in the system). In this example the inactive constraint is situated on the right and the active constraint is on the left.

The distance function d is represented with a bold curve, and the derivative ∇d with a dashed curve, while the tangent lines for the determination of the next iterate in the Newton method are in black straight lines. The black squares correspond to the successive Newton iterates, $\mathbf{x}^0$ being the starting point. The black circles are therefore the successive values $g(\mathbf{x}^k)$, $k = 0, \ldots, i, i+1, \ldots$. In this example d contains only one minimum that is $d(\mathbf{x}_2^{\mathcal{I},n+1})$ and the minimizer of d on $\Delta_{1,1}^{'}$ is the right edge of $\Delta_{1,1}^{'}$ where the Hessian of g becomes singular.

Figure 5.2 (left) shows the minimizing sequence obtained with the Newton method without the adaptive step-length (5.7). The iterate $\mathbf{x}^{i+1}$ leaves the convex area $\Delta_{1,1}^{'}$ and jumps to the convex area of the inactive constraint because the Newton system is ill-conditioned around $\mathbf{x}^i$. Consequently the sequence converges to the global minimizer and $\mathbf{x}_1^{\mathcal{A},n+1}$ is falsely set to $\mathbf{x}_2^{\mathcal{I},n+1}$.

Figure 5.2 (right) illustrates the convergence of the sequence with step-length modification. The iterate $\mathbf{x}^{i+1}$ is modified by (5.7) and its new value falls in the area where the Hessian is not positive definite. The Newton method is then stopped and $\mathbf{x}_1^{\mathcal{A},n+1}$ is set to $\mathbf{x}^i$ which is situated near the local minimizer.

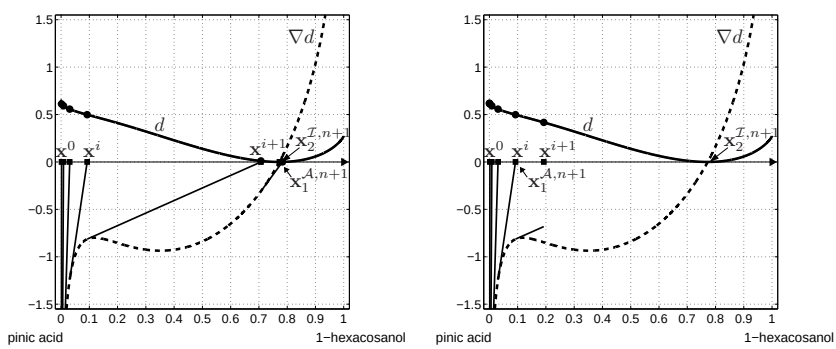

FIG. 5.2. *Left: Steps of the Newton algorithm for the computation of the point at minimal distance to the tangent plane without the criterion on the increment ($\alpha_i = 1$). Right: same but with the criterion on the increment.*

At each iteration of the Newton method the distance is computed and the algorithm is stopped if the distance is negative. Otherwise the algorithm stops if the stopping criterion on the Euclidean norm of the residuals is smaller than a fixed tolerance, or if a maximal number of iterations K is reached.

The converged iterate of the Newton method serves as the initial guess of the Newton method for the next time step, *i.e.* a classical continuation method for the computation of the point at minimal distance of the tangent plane is used (see *e.g.* [3, 8]). The algorithm for the computation of the minimal distance is summarized as follows:

ALGORITHM 5.1. *At each time step t_{n+1} and for each inequality constraint such that $\alpha \in \mathcal{A}(t_n)$, initialize $\mathbf{x}^0 = \mathbf{x}_\alpha^n$ and $\zeta^0 = \zeta^n$. Then, for $i = 1, \ldots, K$*

(i) Build and solve the system (5.6) to obtain $\mathbf{p}_\mathbf{x}^i$ and p_ζ^i.

(ii) Compute $\mathbf{x}^i = \mathbf{x}^{i-1} + \alpha_i \mathbf{p}_\mathbf{x}^i$ and $\zeta^i = \zeta^{i-1} + \alpha_i p_\zeta^i$ with (5.7).

(iii) If $\boldsymbol{\nabla}^2 g(\mathbf{x}^i)$ is not positive definite, or $\mathbf{x}^i$ does not belong to the simplex $\Delta_s^{'}$, or if Newton does not converge, STOP and set $\mathbf{x}_\alpha^{n+1} = \mathbf{x}^{i-1}$.

(iv) If the distance to the supporting tangent plane is negative, if the stopping criterion is satisfied, or if $i = K$, STOP. If $\mathbf{x}^i$ is not colinear to another $\mathbf{x}_\alpha$, then set $\mathbf{x}_\alpha^{n+1} = \mathbf{x}^i$; otherwise, set $\mathbf{x}_\alpha^{n+1} = \mathbf{x}^0$.

5.3. Computation of the Discontinuity Point. Let us assume in the following that an inequality constraint is activated/deactivated in the time interval $[t_n, t_{n+1}]$. The computation of the exact time of discontinuity follows [13] and introduces the partial time step as an (unknown) additional variable, together with the additional event function equation.

Let us denote by W the function describing the event location. This function depends directly on the dense output $\mathbf{U}^n$ defined on the interval $[t_n, t_{n+1}]$. Let us denote by $\tau \in [t_n, t_{n+1}]$ the time $\tau = t_n + h_n$, which is the root of the function W. The problem corresponds therefore to finding $(\mathbf{Y}^{n+1}, h_n)$, satisfying:

$$M(\mathbf{Z}^i - \mathbf{Y}^n) = h_n \sum_{j=1}^{q} a_{ij}\, \mathbf{F}(\mathbf{Z}^j), \quad \forall i = 1, \ldots, q, \tag{5.8}$$

$$M(\mathbf{Y}^{n+1} - \mathbf{Y}^n) = h_n \sum_{j=1}^{q} c_j\, \mathbf{F}(\mathbf{Z}^j), \tag{5.9}$$

$$W(\mathbf{U}^n(t_n + h_n)) = 0. \tag{5.10}$$

Following [13], a *splitting algorithm* is advocated, that couples the RADAU5 algorithm together with a bisection method. It is summarized as follows.

ALGORITHM 5.2. *At each time step t_n such that an activation/deactivation is detected in $[t_n, t_{n+1}]$, consider the system (5.8)-(5.10) and solve it as follows:*

(i) compute $(h_n)_0 = \theta h_n$ as the root of $W(\mathbf{U}^n(t_n + \theta h_n)) = 0$, where $\mathbf{U}^n(t)$ is the dense output obtained from the solution of (5.8)-(5.9);

(ii) for $k = 0, 1, \ldots$ until convergence

(a) solve (5.8)-(5.9) with $h_n = (h_n)_k$; this yields a dense output $\mathbf{U}_k^n(t_n + \theta(h_n)_k)$ for $\theta \in [0, 1]$;

(b) with $\mathbf{U}^n$ replaced by $\mathbf{U}_k^n$, compute $(h_n)_{k+1}$ with a bisection method applied to (5.10);

(iii) terminate the iterations with a step of (5.8)-(5.9).

The convergence criterion is based on the difference between 2 successive step lengths $(h_n)_k$, i.e. $|(h_n)_{k+1} - (h_n)_k| < \varepsilon$, where ε is a given prescribed tolerance.

The addition of the time step as an unknown in (5.8)-(5.10) [13] allows to avoid the numerical error due to the dense output formula and to recover the full accuracy of the method. Furthermore the choice of the splitting algorithm for the solution of (5.8)-(5.10) allows for a simple implementation. The event function W is defined explicitly in the sequel for the cases of an activation and a deactivation.

The Case of the Activation of a Constraint. A constraint is activated if there exists $\bar{\alpha} \in \mathcal{I}(t_n)$ such that corresponds to $y_{\bar{\alpha}}^n > 0$ and $y_{\bar{\alpha}}^{n+1} < 0$. Hence a natural definition for W is $W(\mathbf{U}^n(t_n + h_n)) = \mathbf{U}_{y_{\bar{\alpha}}}^n(t_n + h_n)$ where $\mathbf{U}_{y_{\bar{\alpha}}}^n$ is the component of $\mathbf{U}^n$ relative to the variable $y_{\bar{\alpha}}$.

The Case of the Deactivation of a Constraint. When there exists $\bar{\alpha} \in \mathcal{A}(t_n)$ such that the distance between $(\mathbf{x}_{\bar{\alpha}}^{n+1}, g(\mathbf{x}_{\bar{\alpha}}^{n+1}))$ and the supporting tangent plane defined by the normal vector $\boldsymbol{\lambda}^{n+1}$ is negative, set

$$\begin{aligned} W(\mathbf{U}^n(t_n + h_n)) &= g(\mathbf{x}_{\bar{\alpha}}^{\boldsymbol{\lambda}}(t_n + h_n)) + \mathbf{U}_{\boldsymbol{\lambda}}^{n,T}(t_n + h_n)\, \mathbf{x}_{\bar{\alpha}}^{\boldsymbol{\lambda}}(t_n + h_n) \\ &\quad \text{with } \mathbf{e}^T \mathbf{x}_{\bar{\alpha}}^{\boldsymbol{\lambda}}(t_n + h_n) - 1 = 0, \end{aligned} \tag{5.11}$$

where $\mathbf{U}^n_{\boldsymbol{\lambda}}$ is the subvector of $\mathbf{U}^n$ relative to the variable $\boldsymbol{\lambda}$ and $\mathbf{x}^{\boldsymbol{\lambda}}_{\bar{\alpha}}$ is the point that minimizes the distance to the supporting tangent plane defined by the normal vector $\mathbf{U}^n_{\boldsymbol{\lambda}}(t_n+h_n)$. The expression of W resumes the definition of the distance d, but unlike in (5.4) $\mathbf{U}^n_{\boldsymbol{\lambda}}(t_n+h_n)$ is also an unknown in (5.11). Hence during the bisection steps of Algorithm 5.2 for each $\mathbf{U}^n_{\boldsymbol{\lambda}}(t_n+\theta(h_n)_k)$ the minimization problem (5.4) is solved with $\mathbf{U}^n_{\boldsymbol{\lambda}}(t_n+\theta(h_n)_k)$ instead of $\boldsymbol{\lambda}^{n+1}$ in order to determine $\mathbf{x}^{\boldsymbol{\lambda}}_{\bar{\alpha}}$.

After the computation of the activation or deactivation time, all variables in $\mathbf{Y}$ are reinitialized to their value at time $t=\tau$ thanks to (iii) in Algorithm 5.2. The differential-algebraic system (4.4) (or (4.2)) is then updated by moving the index $\bar{\alpha}$ from the set $\mathcal{I}(\tau)$ into the set $\mathcal{A}(\tau)$ or vice-versa. The complete algorithm is summarized as follows:

ALGORITHM 5.3 (Summary of Complete Algorithm). *For a fixed number of active inequality constraints, solve (4.2) with the RADAU5 algorithm. At each time step t_{n+1}:*

(i) Verify if one (or several) inactive constraint has to be activated. If so, stop RADAU5 and compute the activation time τ with the Algorithm 5.2.

(ii) Verify if one (or several) active constraint has to be deactivated. If so, stop RADAU5 and compute the deactivation time τ with the Algorithm 5.2.

(iii) Determine the minimal time τ among all events detected, update the set of active constraints and the new size of (4.2). Restart the time-discretization scheme RADAU5 at $t=\tau$.

In the case when several events appear during the same time interval, the method detects the event with the smallest event time, compute the corresponding event, and restart the time-stepping procedure before computing the second event during the next time step. The adaptive time step procedure allows to avoid (as much as possible) the presence of several events in one time step.

REMARK 5.1. *When $r>1$, the computation of the point satisfying the minimal distance to the supporting tangent plane in Algorithm 5.1 depends on the topology of the energy function g. In order to improve the robustness of the algorithm and avoid to miss a deactivation time, the number of active constraints obtained by the RADAU5 algorithm may be compared with the number of actual active constraints computed by using the interior-point method described in [2]. The robust version of the algorithm returns back a few time steps when a mismatch is detected.*

6. Numerical Results. Numerical results are presented for various space dimensions r (corresponding to a chemical system of $r+1=s$ components). Graphical results are given for low dimensions, while the computational cost of the algorithm is studied for larger dimensions. The numerical parameters typically used are as follows: $\delta=10$, $K=7$, $c_{\text{thres}}=0.1$, $\varepsilon=10^{-7}$ and for the RADAU5 method the absolute and relative error tolerances are respectively equal to 10^{-13} and 10^{-7}.

6.1. Numerical Results in One Dimension. The chemical system composed of pinic acid ($C_9H_{14}O_4$) and 1-hexacosanol ($C_{26}H_{54}O$) at temperature 298.15 [K] and pressure 1 [atm] is considered ($r+1=2$) as an example of two-components system.

Figure 6.1 (left) shows the time evolution of the vector $\mathbf{b}$ on the phase diagram Δ_1. For more visibility the approximations $\mathbf{b}^n$ are lying on an axis situated just above the phase diagram. The approximations are represented by grey diamonds for the region where one inequality constraint is inactive, and black diamonds are for the regions where both constraints are inactive. The initial point $\mathbf{b}^0$ is situated in the left convex region of the phase diagram and one constraint is inactive ($y_1>0$ and $y_2=0$), then $\mathbf{b}^n$ moves from left to right. The corresponding iterates $g(\mathbf{b}^n)$, moving

on the convex envelope of g, and the corresponding supporting tangent planes are also represented.

The time evolution of $\mathbf{b}^n$, $n = 0, 1, \ldots$ with their distinction between grey and black follows the phase diagram correctly. First approximations are single-phase points, and the corresponding tangent planes are tangent to the curve g at only one point and lie below g. When $\mathbf{b}$ comes closer to the deactivation the tangent planes come near a second contact point with g. At the moment of the deactivation the supporting plane is tangent to g at 2 points ($\mathbf{x}_1 = 0.0665672398$ and $\mathbf{x}_2 = 0.463349192$). These two points are accurate approximations of the points situated at the boundaries of the area on Δ_1 where both constraints are inactive. A zoomed-in view of the deactivation on g is proposed in Figure 6.1 (middle). After the deactivation the points $g(\mathbf{b}^n)$ follow the convex envelope of g. Furthermore the tangent planes touch g at two points and are superposed with the convex envelope of g. Figure 6.1 (right) illustrates the time evolution of y_1 and y_2, and exhibits a discontinuity of the derivatives at time $t = 0.3725$[s] when the second inequality constraint is deactivated, for *both* of the variables.

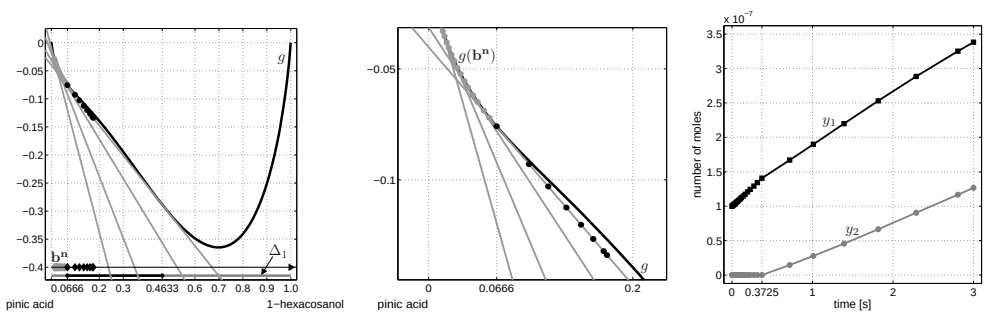

FIG. 6.1. *Deactivation of an inequality constraint for a two-components system. Left: evolution of* $\mathbf{b}$*, the corresponding supporting tangent plane evolves until making contact with the graph of* g*. Middle: zoomed-in view of deactivation on* g*, the points* $g(\mathbf{b})$ *follow the convex envelope of* g*. Right: evolution of* y_1 *and* y_2*, exhibiting discontinuities in the derivatives at the deactivation time.*

Figure 6.2 uses the same notations as in Figure 6.1 to illustrate the time evolution of $\mathbf{b}$ (left), and y_1 and y_2 (right) when one inequality constraint is activated, namely when $\mathbf{b}$ moves from the middle of Δ_1 to the extreme right of the phase diagram. The point $\mathbf{b}^n$ for which the activation occurs is situated on the frontier of Δ_1 between the area of 2 inactive constraints and the one with only one inactive constraint. After the activation, the tangent planes get released from g and remain below g.

Figure 6.3 finally illustrates the difficulty in computing the minimal distance between the graph of g and the supporting tangent plane. The distance function d is represented by a black curve, whereas the iterations of the Newton method are black diamonds and denoted by $\mathbf{x}^i$. The left figure shows the distance function when $\mathbf{b}$ is far away from the deactivation. In this instance, the distance function is convex and admits one unique (global) minimum, namely the contact point of the supporting tangent plane and the graph of the function. The Newton sequence is stopped since $\mathbf{x}^{i+1}$ goes out of its convex region, and $\mathbf{x}^{\mathcal{A},far} = \mathbf{x}^i$. When $\mathbf{b}$ gets closer from the discontinuity time, at time t_n the distance function is stretched and a local minimum appears (see middle figure). The Newton sequence described in Algorithm 5.1 converges to the local minimum. At time t_{n+1} (right figure), the Newton sequence converges to a point with negative distance to the tangent plane, allowing the detection of the

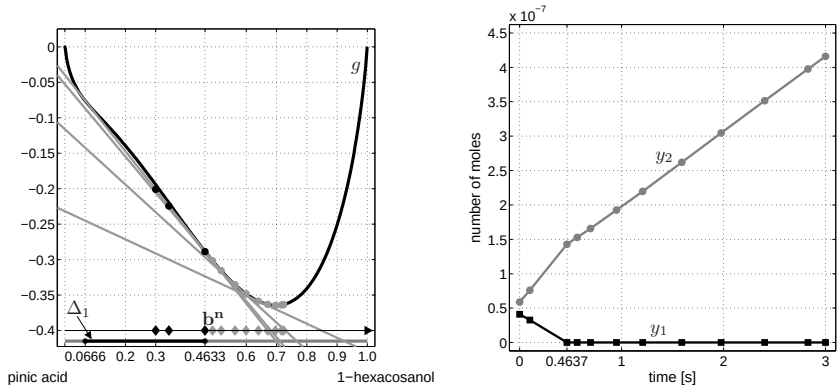

FIG. 6.2. *Activation of an inequality constraint for a two-components system. Left: evolution of* $\mathbf{b}$*; the corresponding supporting tangent plane evolves after leaving the contact with the left convex region on the graph of* g*. Right: evolution of* y_1 *and* y_2*, exhibiting discontinuities in the derivatives at the activation time.*

deactivation. This point is a good approximation of the deactivation point.

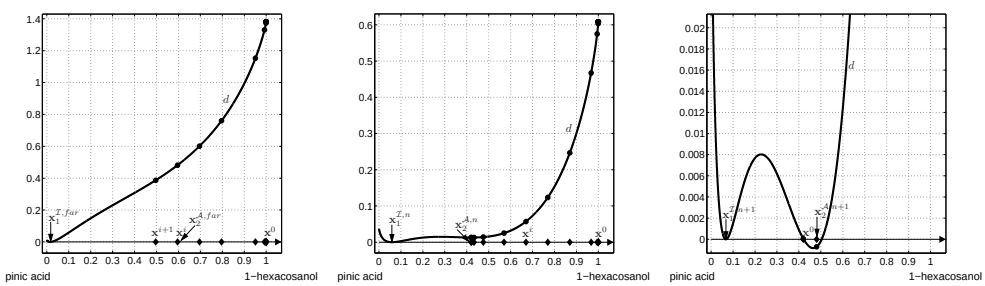

FIG. 6.3. *Computation of the minimal distance between the graph of* g *and the supporting tangent plane. Left: distance function when* $\mathbf{b}$ *is far away from the event; middle: at time* t_n*, a local minimum appears; right: at time* t_{n+1}*, convergence to a point with negative distance to the tangent plane.*

6.2. Numerical Results in Two Dimensions. The chemical system composed of pinic acid ($C_9H_{14}O_4$), 1-hexacosanol ($C_{26}H_{54}O$) and water (H_2O) at temperature 298.15 [K] and pressure 1 [atm] is considered ($r + 1 = 3$). The solution $\mathbf{b}$ and its numerical approximation are represented on a two-dimensional simplex Δ_2 [1, 17, 18]. The regions of the simplex with respectively one, two or three deactivated constraints are numbered by $1, 2, 3$ on the simplex.

Figure 6.4 illustrates the solution of one initial value problem. The initial composition $\mathbf{b}^0$ consists of 15% of pinic acid, 80% of 1-hexacosanol and 5% of water. The initial time step is 0.1[s]. Figure 6.4 (left) shows two simulated trajectories of $\mathbf{b}(t)$, one with tracking of discontinuities (grey line) and the other without tracking (black line). The grey trajectory undergoes two deactivations and one activation of constraints, whereas the black one stands for approximations $\mathbf{b}^n$ that remain single-phase points (branch of local minima) during the whole simulation.

Figure 6.4 (left) demonstrates that the tracking of such events strongly influences the solution of the initial value problem. Figure 6.4 (middle) is a zoomed-in view on the phase diagram that illustrates how the trajectories move away from each other

after the first deactivation. At the end both trajectories converge to the unique stationary solution of the closed system. Figure 6.4 (middle) emphasizes the importance to detect and compute the discontinuity points accurately.

Figure 6.4 (right) illustrates the (piecewise continuously differentiable) evolution of y_α, $\alpha = 1, \ldots, p$, the number of moles relative to each liquid phases $\mathbf{x}_\alpha$ present in the aerosol. At $t = 0$, $y_1 = y_2 = 0$ and $y_3 > 0$, and two constraints are activated (*i.e.* the particle only contains the third liquid phase). Then constraints are activated/deactivated and the variables y_α, $\alpha = 1, \ldots, p$ present jumps of the derivatives at each event.

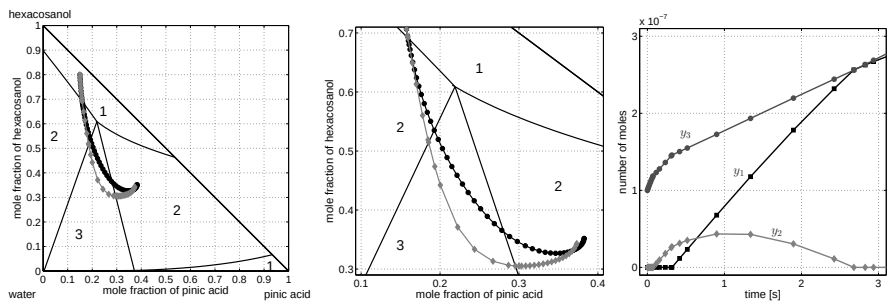

FIG. 6.4. *Left: Evolution of* $\mathbf{b}$ *on the phase diagram of the particle without the tracking of the discontinuity points (black line) and with the tracking (grey line); middle: zoomed-in view; right: time evolution of the number of moles relative to each liquid phase present in the particle.*

In the particular case when all $y_\alpha(t)$ remain strictly positive, $\boldsymbol{\lambda}(t)$ is constant and the tangent plane remains unchanged [2]. The solution computed by solving a pure optimization problem is considered as the "exact solution". Similarly, solving the pure optimization problem allows to accurately compute the boundaries of Δ_2, where all variables $y_\alpha(t)$ are strictly positive. Due to the particular expression of the fluxes $\mathbf{f}$, the differential equations are decoupled from the optimization problem, and the system of differential-algebraic equations is reduced to a system of linear ODEs. The "exact solution" is therefore the intersection of the trajectory $\mathbf{b}(t)$ with the linear given interface.

Four different examples are considered starting all from the area on Δ_2 and going to one of the areas where only 2 constraints are inactive. Figure 6.5 illustrates the error on the computation of activation points between the approximated and exact solutions for each example. It shows that the error on both the time and location of the activation is negligible, up to machine precision and algorithm tolerance, and validate the accuracy of our algorithm.

6.3. Numerical Results in Higher Dimensions. When r is greater than 3, the phase diagrams cannot be easily visualized. In this section we compare the CPU times for different values of r. Table 6.1 summarizes the computational time for 6 examples (for $r = 2, 3$ and 17 resp.) that run on an Intel processor of 2.4 GHz. The CPU times illustrate respectively the total time of execution, the time for the detection of events, the time for the computation of the activation, the time spent in going backwards in the trajectory, the time for the computation of the deactivation and the total time for the detection and computation of the discontinuities. Table 6.1 shows that the larger r, the more expensive the tracking of discontinuity points (as expected). However, the percentage of computational cost of the resulting cost for

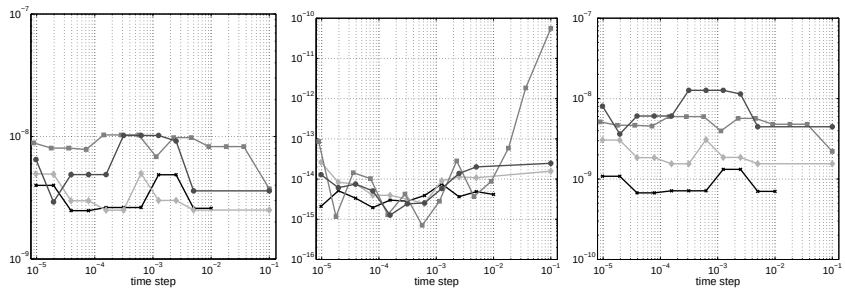

FIG. 6.5. *Error on the computation of the activation/deactivation of inequality constraints: the case of the activation of a constraint. Error on the location of activation* $\|\mathbf{b}(t_\star) - \mathbf{b}^n\|_2$ *(left);* $\|\mathbf{b}(t_n) - \mathbf{b}^n\|_2$ *(middle) and error on the activation time* $|t_\star - \tau|$ *(right).*

the tracking remains stable as r becomes larger. The time for the computation of activations is negligible compared to the time for deactivations. For both cases the number of iterations in the splitting algorithm 5.2 is equal to 3 in average and the number of iterates for the bisection in the deactivation case is equal to 30 in average.

TABLE 6.1

Computational cost percentages of the algorithm for system with $r + 1 = 3, 4, 18$. *Legend is as follows: code: total time; detect: time for the detection of events; act.: computation of activation time; backwards: time spend in going backwards in the trajectory for checking purposes; deact.: computation of deactivation time; total disc.: total time for detection and computation of events.*

	detect	act.	backwards	deact.	total disc.
Ex. 1: $r = 2$ [%]	5.3	15.2	16.0	25.3	61.8
Ex. 2: $r = 2$ [%]	2.6		8.0	27.3	37.9
Ex. 3: $r = 3$ [%]	25.7	38.6			64.3
Ex. 4: $r = 3$ [%]	10.1		27.1	23.5	60.7
Ex. 5: $r = 17$ [%]	23.1	18.6	19.7	14.6	76
Ex. 6: $r = 17$ [%]	41.8		4.1	24.7	70.6

Ex. 1: 2 deactivations and 1 activation; Ex. 2: 1 deactivation; Ex. 3: 1 activation; Ex. 4: 1 deactivation; Ex. 5: 1 deactivation and 3 activations; Ex. 6: 1 deactivation.

7. Conclusion. A numerical method for the simulation of differential equations coupled with a global optimization problem has been presented. It allows to take into consideration the activation/deactivation of inequality constraints that occurs at unknown times. It couples an implicit Runge-Kutta method (RADAU5), with tracking techniques that rely on dense output formulas, nonlinear programming techniques for non-convex constrained optimization and geometric considerations. Numerical results in the framework of atmospheric chemistry for the simulation of the dynamics of organic aerosol particles have illustrated the accuracy and efficiency of the method.

Acknowledgments. The authors would like to thank Professors J. W. He (University of Houston) and J. Rappaz (EPFL) for fruitful discussions and helpful comments. The contribution of E. Hairer is supported by the Fonds National Suisse, project No. 200020-121561.

REFERENCES

[1] N. R. Amundson, A. Caboussat, J. W. He, C. Landry, and J. H. Seinfeld. A dynamic optimization problem related to organic aerosols. *C. R. Acad. Sci.*, 344(8):519–522, 2007.

[2] N. R. Amundson, A. Caboussat, J. W. He, and J. H. Seinfeld. Primal-dual interior-point algorithm for chemical equilibrium problems related to modeling of atmospheric organic aerosols. *J. Optim. Theory Appl.*, 130(3):375–407, 2006.

[3] H. Antil, R. H. W. Hoppe, and C. Linsenmann. Path-following primal-dual interior-point methods for shape optimization. *Journal of Numerical Mathematics*, 15(2):81–100, 2007.

[4] U. M. Ascher and L. R. Petzold. *Computer Methods for Ordinary Differential Equations and Differential-Algebraic Equations*. Society for Industrial and Applied Mathematics, 1998.

[5] J. C. Butcher. *The numerical analysis of ordinary differential equations: Runge-Kutta and general linear methods*. Wiley, Chichester, 1987.

[6] A. Caboussat and C. Landry. Dynamic optimization and event location in atmospheric chemistry. *Proc. Appl. Math. Mech. (PAMM)*, 7(1):2020035–2020036, 2007.

[7] A. Caboussat and C. Landry. A second order scheme for solving optimization-constrained differential equations with discontinuities. In *Numerical Mathematics and Advanced Applications*, pages 761–768. Springer Verlag, Berlin, 2008. Proceedings of Enumath 2007.

[8] P. Deuflhard. *Newton methods for nonlinear problems: affine invariance and adaptive alorithms*. Springer Verlag, Berlin, 2004.

[9] J. M. Esposito and V. Kumar. A state event detection algorithm for numerically simulating hybrid systems with model singularities. *ACM Transactions on Modeling and Computer Simulation (TOMACS)*, 17:1–22, 2007.

[10] B. Faugeras, J. Pousin, and F. Fontvieille. An efficient numerical scheme for precise time integration of a diffusion-dissolution/precipitation chemical system. *Math. Comp.*, 75(253):209–222, 2005.

[11] C. W. Gear and O. Østerby. Solving ordinary differential equations with discontinuities. *ACM Trans. Math. Software*, 10(1):23–44, 1984.

[12] N. I. M. Gould. On practical conditions for the existence and uniqueness of solutions to the general equality quadratic programming problem. *Math. Prog.*, 32:90–99, 1985.

[13] N. Guglielmi and E. Hairer. Computing breaking points in implicit delay differential equations. *Advances in Computational Mathematics*, 29(3):229–247, 2008.

[14] E. Hairer, S. P. Nørsett, and G. Wanner. *Solving Ordinary Differential Equations I: Nonstiff Problems. 2nd Edition*, volume 8 of *Springer Series in Computational Mathematics*. Springer-Verlag, 1993.

[15] E. Hairer and G. Wanner. *Solving Ordinary Differential Equations II. Stiff and Differential-Algebraic Problems. 2nd Edition*, volume 14 of *Springer Series in Computational Mathematics*. Springer-Verlag, 1996.

[16] E. Hairer and G. Wanner. Stiff differential equations solved by Radau methods. *J. Comput. Appl. Math.*, 111:93–111, 1999.

[17] G. A. Iglesias-Silva, A. Bonilla-Petriciolet, P. T. Eubank, J. C. Holste, and K. R. Hall. An algebraic method that includes Gibbs minimization for performing phase equilibrium calculations for any number of components or phases. *Fluid Phase Equilibria*, 210:229–245, 2003.

[18] Y. Jiang, G. R. Chapman, and W. R. Smith. On the geometry of chemical reaction and phase equilibria. *Fluid Phase Equilibria*, 118(1):77–102, 1996.

[19] J.-L. Lions. *Optimal control of systems governed by partial differential equations*. Springer-Verlag, berlin, New York, 1971.

[20] C. M. McDonald and C. A. Floudas. GLOPEQ: A new computational tool for the phase and chemical equilibrium problem. *Computers and Chemical Engineering*, 21(1):1–23, 1996.

[21] J. Nocedal and S. J. Wright. *Numerical Optimization*. Springer-Verlag, 1999.

[22] P. J. Rabier and A. Griewank. Generic aspects of convexification with applications to thermodynamic equilibrium. *Arch. Rat. Mech. Anal.*, 118(4):349–397, 1992.

[23] P. J. Rabier and W. C. Rheinboldt. *Theoretical and Numerical Analysis of Differential-Algebraic Equations*, volume VIII of *Handbook of Numerical Analysis (P.G. Ciarlet, J.L. Lions eds)*, pages 183–542. Elsevier, Amsterdam, 2002.

[24] J. H. Seinfeld and S. N. Pandis. *Atmospheric Chemistry and Physics: From Air Pollution to Climate Change*. Wiley, New York, 1998.

[25] L. F. Shampine, I. Gladwell, and R. W. Brankin. Reliable solution of special event location problems for ODEs. *ACM Trans. Math. Software*, 17:11–25, 1991.

[26] Vladimir Veliov. On the time-discretization of control systems. *SIAM J. Control Optim.*, 35(5):1470–1486, 1997.