An algorithm for gradient-based dynamic optimization of UV ash processes

Ritschel, Tobias Kasper Skovborg; Capolei, Andrea; Gaspar, Jozsef; Jørgensen, John Bagterp

Published in:
Computers & Chemical Engineering

Link to article, DOI:
10.1016/j.compchemeng.2017.10.007

Publication date:
2018

Document Version
Peer reviewed version

Citation (APA):
Ritschel, T. K. S., Capolei, A., Gaspar, J., & Jørgensen, J. B. (2018). An algorithm for gradient-based dynamic optimization of UV ash processes. *Computers & Chemical Engineering*, *114*, 281–295. https://doi.org/10.1016/j.compchemeng.2017.10.007

Title: An algorithm for gradient-based dynamic optimization of UV flash processes

Author: Tobias K.S. Ritschel Andrea Capolei Jozsef Gaspar John Bagterp Jørgensen

PII: S0098-1354(17)30352-6
DOI: https://doi.org/doi:10.1016/j.compchemeng.2017.10.007
Reference: CACE 5913

To appear in: *Computers and Chemical Engineering*

Received date: 31-8-2017
Accepted date: 10-10-2017

- We present asingle-shooting algorithm for optimal control of UV flash processes.
- The algorithm uses an adjoint method to compute gradients.
- We formulate the optimal control problem as a bilevel optimization problem.
- A simultaneous approach for UV flash simulation is faster than a nested approach.
- We compare optimization software, linear algebra software, and compilers.

An algorithm for gradient-based dynamic optimization of UV flash processes

Tobias K. S. Ritschel, Andrea Capolei, Jozsef Gaspar, John Bagterp Jørgensen[*]

Department of Applied Mathematics and Computer Science & Center for Energy Resources Engineering (CERE),
Technical University of Denmark, DK-2800 Kgs. Lyngby, Denmark

Abstract

This paper presents a novel single-shooting algorithm for gradient-based solution of optimal control problems with vapor-liquid equilibrium constraints. Such optimal control problems are important in several engineering applications, for instance in control of distillation columns, in certain two-phase flow problems, and in operation of oil reservoirs. The single-shooting algorithm uses an adjoint method for the computation of gradients. Furthermore, the algorithm uses either a simultaneous or a nested approach for the numerical solution of the dynamic vapor-liquid equilibrium model equations. Two numerical examples illustrate that the simultaneous approach is faster than the nested approach and that the efficiency of the underlying thermodynamic computations is important for the overall performance of the single-shooting algorithm. We compare the performance of different optimization software as well as the performance of different compilers in a Linux operating system. These tests indicate that real-time nonlinear model predictive control of UV flash processes is computationally feasible.

Keywords: Dynamic optimization, Optimal control, Adjoint algorithm, Single-shooting, UV flash, Vapor-liquid equilibrium

1. Introduction

Dynamic optimization, also called optimal control, is concerned with computing an open-loop control strategy that manipulates a dynamical system in such a way that it optimizes some performance measure, e.g expected profit or deviation of a product quality from a target (Betts, 2001; Binder et al., 2001; Bryson, Jr., 1999; Diehl et al., 2009; Zavala and Biegler, 2009). Systems modeled by mass and energy balances and subject to vapor-liquid equilibrium constraints occur in a number of important process engineering applications, e.g. distillation (Biegler, 2010; Bisgaard et al., 2015, 2017; Diehl et al., 2002; Luyben, 1992; Stichlmair and Fair, 1998), cryogenic distillation (Laiglecia et al., 2012), and two-phase flow in pipelines (Hammer and Morin, 2014; Qiu et al., 2014). Vapor-liquid equilibrium constrained mass and energy conservation models also occur for a number of subsurface flow processes related to CO_2 sequestration (Stauffer et al., 2009), magmatic hydrothermal flow (Ingebritsen et al., 2010), and production of oil from an oil reservoir (Li and Johns, 2006; Lucia et al., 2012; Zaydullin et al., 2014). The UV flash is a single stage vapor-liquid equilibrium process and is thus a key component in rigorous modeling of fluid vessels and flash drums (Arendsen and Versteeg, 2009; Castier, 2010; Lima et al., 2008), distillation columns (Flatby et al., 1994), two-phase computational fluid dynamical problems (Qiu et al., 2014), and thermal and compositional oil reservoir flow (Zaydullin et al., 2014). While advanced algorithms for robust simulation of the UV flash problem exist (Castier, 2009; Saha and Carroll, 1997), no algorithm for dynamic optimization of UV flash processes seems to exist in the open literature. Dynamic optimization of UV flash processes was first explained by Ritschel et al. (2017a,b). The UV flash problem is also known as the isoenergetic-isochoric flash problem or the UVn flash problem. UVn refers to specification of the internal energy, U, the total volume, V, and the total material amount (moles), n. The second law of thermodynamics, i.e. the entropy of a closed system is maximal, is used to determine the equilibrium composition with U, V, and n specified (Michelsen, 1999). The UV flash problem is different from the more common PT flash problem that occurs in steady-state optimization problems. However, it can be demonstrated that the PT flash problem with additional constraints on the internal energy, U, and the volume, V, is equivalent to the UV flash problem. Algorithmic oriented approaches to dynamic optimization of vapor-liquid equilibrium processes use a nested method in which PT flash problems are solved in the inner loop, and outer loops converge the internal energy, U, and volume, V, to their specified values.

In this paper, we develop a single-shooting algorithm for solution of dynamic optimization problems with vapor-liquid equilibrium constraints, i.e. an algorithm for dynamic optimization of the UV flash problem. The key novelties of our algorithm are that an adjoint method is used for the computation of gradients (Jørgensen, 2007) and that the problem is formulated as a bilevel optimization problem. The numerical integration of the semi-explicit index-1 differential algebraic (DAE) system is the key computational operation in the single-shooting method. The DAE systems can be solved numerically with either a simultaneous approach or with a nested approach. The discretized differential equations and the equilibrium con-

[*] Corresponding author.

Email addresses: tobk@dtu.dk (Tobias K. S. Ritschel), acap@dtu.dk (Andrea Capolei), joca@dtu.dk (Jozsef Gaspar), jbjo@dtu.dk (John Bagterp Jørgensen)

ditions are solved simultaneously in the simultaneous approach. In the nested approach, the equilibrium conditions are solved in an inner loop for each evaluation of the discretized differential equations. The key potential advantage of a nested approch is that well established algorithms for the PT-flash may be used in the inner loop. We report numerical results as well as the computational performance for implementations in C and Matlab using different optimization software, different linear algebra software, and different compilers. The computations confirm previous results in which the simultaneous approach is faster than a nested approach (Wilhelmsen et al., 2013). Hence, the performance of the single-shooting algorithm depends on the efficiency of the thermodynamic function evaluations and the numerical linear algebra, but also on whether the equilibrium equations are solved simultaneously with the differential equations or in a loop nested to the differential equations.

Previously, Kourounis et al. (2014) developed an adjoint method for gradient-based optimization of compositional reservoir flow. Their model included isothermal and isobaric (constant temperature and pressure) vapor-liquid equilibrium processes, but not the isoenergetic-isochoric situation considered in this paper. Alternatives to the single-shooting method exist. They are the multiple-shooting method (Bock and Plitt, 1984; Capolei and Jørgensen, 2012) and the simultaneous method (Biegler, 2007). Both methods have been applied to optimal control problems with vapor-liquid equilibrium constraints (Raghunathan et al., 2004; Schäfer et al., 2007). The key insight in our formulation of the vapor-liquid equilibrium is that it may be formulated as an equality constrained optimization problem (Michelsen, 1999), which in the UV-flash case is a natural and intuitive representation of the second law of thermodynamics; i.e. the entropy is maximal of a closed system with fixed energy, volume and mass. Accordingly, optimal control problems with vapor-liquid equilibrium constraints belong to a class of bilevel optimization problems (Colson et al., 2007) and also to the closely related class of mathematical programs with equilibrium constraints (Luo et al., 1996; Outrata et al., 2013). This structure is exploited in the efficient computation of the resulting index-1 differential-algebraic system and its adjoints. However, it should be noted that we only consider the situation with both phases (vapor and liquid) present, but not the more complicated situation in which phases can appear and disappear (Biegler, 2010; Sahlodin et al., 2016; Watson et al., 2017).

This paper is organized as follows. Section 2 presents the optimal control problem in consideration. Section 3 presents the single-shooting algorithm based on both the simultaneous approach and the nested approach. Section 4 describes a dynamical UV flash model, and Section 5 demonstrates the equivalence between the UV flash problem and a PT flash problem with additional constraints on the internal energy and the volume. Section 6 discusses the implementation of the single-shooting algorithm. Section 7 presents numerical solutions to a tracking-type control problem and an economical control problem. Section 8 presents a numerical performance study. Conclusions are given in Section 9.

2. Optimal control problem

We consider the following optimal control problem (OCP)

$$\min_{[x(t);y(t);z(t)]_{t_0}^{t_f},\{u_k\}_{k\in\mathcal{N}}} \quad \phi = \phi\left([y(t);u(t);d(t)]_{t_0}^{t_f}\right) \tag{1a}$$

subject to

$$x(t_0) = \hat{x}_0, \tag{1b}$$
$$G(x(t),y(t),z(t)) = 0, \qquad t \in \mathcal{T}, \tag{1c}$$
$$\dot{x}(t) = F(y(t),u(t),d(t)), \;\; t \in \mathcal{T}, \tag{1d}$$
$$u(t) = u_k, \; t \in [t_k, \, t_{k+1}[, \quad k \in \mathcal{N}, \tag{1e}$$
$$d(t) = \hat{d}_k, \; t \in [t_k, \, t_{k+1}[, \quad k \in \mathcal{N}, \tag{1f}$$
$$\{u_k\}_{k\in\mathcal{N}} \in \mathcal{U}, \tag{1g}$$

where the objective function, ϕ, is in Lagrange form

$$\phi = \int_{t_0}^{t_f} \Phi(y(t),u(t),d(t))dt. \tag{2}$$

$x(t)$ is the state vector, $y(t)$ is a vector of algebraic variables, and $z(t)$ is a vector of adjoint algebraic variables. The estimated initial state, $\hat{x}_0$, and the predicted disturbances, $\{\hat{d}_k\}_{k\in\mathcal{N}}$, are parameters in the optimization problem. $[x(t);\, y(t);\, z(t)]_{t_0}^{t_f}$ is a vector of dependent decision variables, and $\{u_k\}_{k\in\mathcal{N}}$ are independent decision variables. The time horizon is $\mathcal{T} = [t_0, \, t_f]$, and the indices of the control intervals are $\mathcal{N} = \{0, 1, \ldots, N-1\}$.

The OCP (1) includes algebraic constraints (1c) and differential equations (1d). The equilibrium conditions for an equilibrium process can be formulated as the Karush-Kuhn-Tucker (KKT) conditions of an optimization problem. The algebraic constraints (1c) are formulated such that they can represent such KKT conditions. The differential equations (1d) are obtained from conservation principles, and the states, $x(t)$, represent the conserved quantities. The right-hand side in (1d) depends on the algebraic variables, $y(t)$, which are implicit functions of the states through the algebraic constraints (1c), i.e. $y(t) = y(x(t))$. We assume that it is possible to solve $G(x(t),y(t),z(t)) = 0$ for $y(t) = y(x(t))$ and $z(t) = z(x(t))$ when $x(t)$ is given. This is true for the vapor-liquid equilibrium processes considered in this work. We define the objective function, ψ, as

$$\psi = \psi(\{u_k\}_{k\in\mathcal{N}}; \hat{x}_0, \{\hat{d}_k\}_{k\in\mathcal{N}}) = \Big\{\phi \,:\, \text{(1b)-(1f)}\Big\}. \tag{3}$$

Given $\{u_k\}_{k\in\mathcal{N}}$, $\hat{x}_0$, and $\{\hat{d}_k\}_{k\in\mathcal{N}}$ this defines ψ as the objective function, ϕ, obtained from (2) using the solution of (1c)-(1d) with $x(t_0) = \hat{x}_0$, $u(t) = u_k$ for $t \in [t_k, t_{k+1}[$ and $k \in \mathcal{N}$, and $d(t) = \hat{d}_k$ for $t \in [t_k, t_{k+1}[$ and $k \in \mathcal{N}$, i.e. (1b) and (1e)-(1f). Fig. 1 illustrates the discretization of the inputs and the numerical computation of the continuous states. This is the principle that is used to compute ψ in the single-shooting algorithm. With ψ defined by (3), the OCP (1) with the objective function (2) can be expressed as the finite-dimensional constrained optimization problem

$$\min_{\{u_k\}_{k\in\mathcal{N}}} \quad \psi = \psi\left(\{u_k\}_{k\in\mathcal{N}}; \hat{x}_0, \{\hat{d}_k\}_{k\in\mathcal{N}}\right) \tag{4a}$$
$$\text{s.t.} \qquad \{u_k\}_{k\in\mathcal{N}} \in \mathcal{U}. \tag{4b}$$

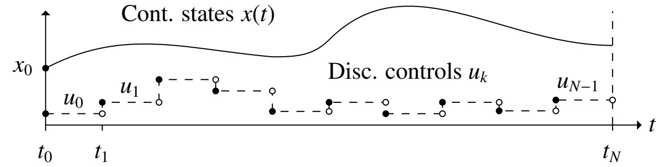


Figure 1: Sketch of the principle in the single-shooting method. The controls, $u(t)$, are discretized in time and the continuous states, $x(t)$, are considered functions of the controls. The objective function is evaluated by solving the dynamic equations for a given set of controls.

The set $\mathcal{U}$ is often a polyhedron such that the constraints (4b) can be expressed as $u_{\min} \leq u \leq u_{\max}$ and $b_l \leq Au \leq b_u$ where $u = [u_0; u_1; \ldots; u_{N-1}]$. Gradient-based optimization algorithms for solving the nonlinear program (4), and thus the optimal control problem (1), require evaluation of the objective function, ψ, and the gradients, $\{\nabla_{u_k}\psi\}_{k\in\mathcal{N}}$. This involves the numerical solution of the differential-algebraic equations (1c)-(1d) and the computation of the integral (2).

2.1. Equilibrium constraints

The equilibrium processes that we consider can be described as the solution to the following parametric optimization problem

$$\min_{y} \quad f(y) \tag{5a}$$
$$\text{s.t.} \quad g(y) = x, \tag{5b}$$
$$h(y) = 0. \tag{5c}$$

The Langrange function associated with the equilibrium optimization problem (5) is

$$\mathcal{L}(y,\eta,\mu;x) = f(y) - \eta^T(g(y)-x) - \mu^T h(y), \tag{6}$$

where η and μ are Lagrange multipliers associated with (5b) and (5c), respectively. The KKT conditions (first order optimality conditions) for a minimizer $(y = y(x), \eta = \eta(x), \mu = \mu(x))$ are

$$\nabla_y\mathcal{L}(y,\eta,\mu;x) = \nabla f(y) - \nabla g(y)\eta - \nabla h(y)\mu = 0, \tag{7a}$$
$$\nabla_\eta\mathcal{L}(y,\eta,\mu;x) = -(g(y)-x) = 0, \tag{7b}$$
$$\nabla_\mu\mathcal{L}(y,\eta,\mu;x) = -h(y) = 0. \tag{7c}$$

By introducing the vector $z = [\eta; \mu]$, we can rewrite the system (7) as

$$G(x,y,z) = 0, \tag{8}$$

which is equivalent to the algebraic constraints (1c).

3. Numerical solution

This section describes the numerical algorithms for the computation of the objective function, ψ, defined in (3) as well as the gradients with respect to the controls, $\{\nabla_{u_k}\psi\}_{k\in\mathcal{N}}$. The evaluation of ψ requires the solution of the semi-explicit differential-algebraic initial value problem

$$x(t_0) = \hat{x}_0, \tag{9a}$$
$$G(x(t),y(t),z(t)) = 0, \qquad t\in\mathcal{T}, \tag{9b}$$
$$\dot{x}(t) = F(y(t),u(t),d(t)), \quad t\in\mathcal{T}. \tag{9c}$$

When $[y(t);u(t);d(t)]_{t_0}^{t_f}$ is given, $\psi = \phi$ is computed by quadrature. An implicit method must be used for efficient numerical solution of the system (9) because it is stiff. There exists several implicit methods such as ESDIRK methods (Kristensen et al., 2004; Völcker et al., 2010) and BDF based methods (Barton and Lee, 2002; Tolsma and Barton, 2000). In this work, we use Euler's implicit method. Furthermore, we describe the computation of the gradients, $\{\nabla_{u_k}\psi\}_{k\in\mathcal{N}}$, by an adjoint method (Capolei and Jørgensen, 2012; Capolei et al., 2012; Jørgensen, 2007; Völcker et al., 2011). These gradients (or sensitivities) may also be computed by a forward method (Kristensen et al., 2004, 2005).

As described in Section 2, the time horizon, $[t_0, t_f]$, is divided into N control intervals. There can be several time steps in each control interval, but for ease of notation, we assume that there is only one time step for each control interval. We consider both a simultaneous approach and a nested approach for the numerical solution of the differential-algebraic system (9). In the simultaneous approach the discretized differential equations, $D_{k+1} = 0$, and the algebraic equations, $G(x_{k+1}, y_{k+1}, z_{k+1}) = 0$, are solved simultaneously for the state variables, x_{k+1}, the algebraic variables, y_{k+1}, and the adjoint algebraic variables, z_{k+1}. In the nested approach, the discretized differential equations, $D_{k+1} = 0$, are solved by iterating on the state variables, x_{k+1}, in an outer loop. For each iterate of x_{k+1} the algebraic variables, $y_{k+1} = y(x_{k+1})$, and the adjoint algebraic variables, $z_{k+1} = z(x_{k+1})$, are computed by solving the algebraic equations, $G(x_{k+1}, y_{k+1}, z_{k+1}) = 0$, in an inner loop. The two approaches are briefly described in the following subsections.

3.1. The simultaneous approach

Define $w = [x; y; z]$ and the residual function

$$\begin{aligned} R_{k+1} &= R_{k+1}(w_{k+1}) = R_{k+1}(w_{k+1}; x_k, u_k, \hat{d}_k) \\ &= R_{k+1}(x_{k+1}, y_{k+1}, z_{k+1}; x_k, u_k, \hat{d}_k) \\ &= \begin{bmatrix} D_{k+1}(x_{k+1}, x_k, y_{k+1}, u_k, \hat{d}_k) \\ G(x_{k+1}, y_{k+1}, z_{k+1}) \end{bmatrix}, \quad k\in\mathcal{N}, \end{aligned} \tag{10}$$

where the function $D_{k+1} = D_{k+1}(x_{k+1}, x_k, y_{k+1}, u_k, \hat{d}_k)$ is

$$D_{k+1} = x_{k+1} - x_k - \Delta t_k F(y_{k+1}, u_k, \hat{d}_k). \tag{11}$$

Euler's implicit method for the system (9) corresponds to solving the residual equations

$$R_{k+1} = R_{k+1}(w_{k+1}) = 0, \quad k\in\mathcal{N}, \tag{12}$$

for $\{w_{k+1}\}_{k\in\mathcal{N}}$ by sequentially marching forward when $x_0 = \hat{x}_0$, $\{u_k\}_{k\in\mathcal{N}}$, and $\{\hat{d}_k\}_{k\in\mathcal{N}}$ are given. We solve the residual equations

(12) with an inexact Newton method, i.e. by solving the following sequence of linear systems

$$w_{k+1}^{m+1} = w_{k+1}^{m} - \left(M_R^m\right)^{-1} R_{k+1}(w_{k+1}^m). \tag{13}$$

The initial guess for the Newton iterations, w_{k+1}^0, is the states, the algebraic variables, and the adjoint algebraic variables in the previous time step, i.e. $w_k = [x_k;\ y_k;\ z_k]$. The initial iteration matrix, M_R^0, is the Jacobian of the residual function evaluated at the initial guess

$$M_R^0 = \frac{\partial R_{k+1}}{\partial w_{k+1}}(w_{k+1}^0). \tag{14}$$

We update the iteration matrix:

$$M_R^{m+1} = \frac{\partial R_{k+1}}{\partial w_{k+1}}(w_{k+1}^{m+1}), \tag{15}$$

if the estimate, w_{k+1}^{m+1}, does not sufficiently reduce the norm of the residual function as compared to the previous estimate, w_{k+1}^m:

$$\|R_{k+1}(w_{k+1}^{m+1})\| > \tau_R \|R_{k+1}(w_{k+1}^m)\|, \tag{16}$$

where $\tau_R \in [0, 1]$. The iteration matrix is not updated (and $M_R^{m+1} = M_R^m$) if (16) is not satisfied. The Newton iterations (13) are terminated when the norm of the residual function is smaller than the specified tolerance, ϵ_R, i.e.

$$\|R_{k+1}(w_{k+1}^{m+1})\| < \epsilon_R. \tag{17}$$

The Jacobian of the residual function is

$$\frac{\partial R_{k+1}}{\partial w_{k+1}} = \begin{bmatrix} I & -\Delta t_k \frac{\partial F}{\partial y} & 0 \\ \frac{\partial G}{\partial x} & \frac{\partial G}{\partial y} & \frac{\partial G}{\partial z} \end{bmatrix}, \tag{18}$$

where the Jacobians of the algebraic function, G, are

$$\frac{\partial G}{\partial x} = \frac{\partial G}{\partial x}(w_{k+1}) = \begin{bmatrix} 0; & \mathrm{I}; & 0 \end{bmatrix}, \tag{19a}$$

$$\begin{bmatrix} \frac{\partial G}{\partial y} & \frac{\partial G}{\partial z} \end{bmatrix} = \begin{bmatrix} \frac{\partial G}{\partial y}(w_{k+1}) & \frac{\partial G}{\partial z}(w_{k+1}) \end{bmatrix} = K(w_{k+1}). \tag{19b}$$

The matrix K is the KKT matrix for the equilibrium optimization problem (5). With the adjoint algebraic variables defined as $z_{k+1} = [\eta_{k+1};\ \mu_{k+1}]$, the KKT matrix may be expressed as $K(w_{k+1}) = K(x_{k+1}, y_{k+1}, z_{k+1}) = K(y_{k+1}, \eta_{k+1}, \mu_{k+1}; x_{k+1})$, where

$$K(y, \eta, \mu; x) = \begin{bmatrix} \nabla_{yy}^2 \mathcal{L}(y, \eta, \mu; x) & -\nabla g(y) & -\nabla h(y) \\ -\nabla g(y)^T & 0 & 0 \\ -\nabla h(y)^T & 0 & 0 \end{bmatrix}. \tag{20}$$

The Hessian of the Lagrangian with respect to the algebraic variables, y, is

$$\begin{aligned} \nabla_{yy}^2 \mathcal{L} &= \nabla_{yy}^2 \mathcal{L}(y, \eta, \mu; x) \\ &= \nabla^2 f(y) - \sum_i \eta_i \nabla^2 g_i(y) - \sum_i \mu_i \nabla^2 h_i(y). \end{aligned} \tag{21}$$

The two sums in (21) span over each component of the functions g and h.

3.1.1. The adjoint method for the simultaneous approach

We substitute the residual equations (12) into the objective function, ψ, in (3):

$$\begin{aligned} \psi &= \psi(\{u_k\}_{k\in\mathcal{N}}; \hat{x}_0, \{\hat{d}_k\}_{k\in\mathcal{N}}) && (22a)\\ &= \Big\{ \phi = \sum_{k\in\mathcal{N}} \Phi_k(y_{k+1}, u_k, \hat{d}_k) \ : && (22b)\\ &\qquad x_0 = \hat{x}_0, && (22c)\\ &\qquad R_{k+1}(w_{k+1}; x_k, u_k, \hat{d}_k) = 0,\ k \in \mathcal{N}, && (22d)\\ &\qquad [x_{k+1}; y_{k+1}; z_{k+1}] = w_{k+1},\ k \in \mathcal{N} \Big\}. && (22e) \end{aligned}$$

The sum in (22b) approximates the integral in (2). Φ_k approximates the integral of Φ over $[t_k,\ t_{k+1}]$ using the rectangle rule with y_{k+1} (instead of y_k):

$$\Phi_k = \Phi_k(y_{k+1}, u_k, \hat{d}_k) = \Delta t_k \Phi(y_{k+1}, u_k, \hat{d}_k). \tag{23}$$

The integral may be approximated more accurately with other quadrature methods. However, the rectangle rule is used because of its simplicity and because it is consistent with the implicit Euler method used for integration of the differential equations. The gradients of ψ with respect to the inputs, $\{\nabla_{u_k}\psi\}_{k\in\mathcal{N}}$, are computed by solving the following equation for the adjoints, λ_N,

$$\left(\frac{\partial R_N}{\partial w_N}\right)^T \lambda_N = -\nabla_{w_N} \Phi_{N-1}, \tag{24}$$

and inserting into the following expression for the gradients

$$\nabla_{u_k}\psi = \nabla_{u_k}\Phi_k + \left(\frac{\partial R_{k+1}}{\partial u_k}\right)^T \lambda_{k+1}, \quad k \in \mathcal{N}. \tag{25}$$

Each subsequent set of adjoints, λ_k, is computed by marching backwards in the equations

$$\left(\frac{\partial R_k}{\partial w_k}\right)^T \lambda_k = -\left(\frac{\partial R_{k+1}}{\partial w_k}\right)^T \lambda_{k+1} - \nabla_{w_k}\Phi_{k-1}, \quad k = N-1, \ldots, 1. \tag{26}$$

The Jacobian of the residual function, $\frac{\partial R_k}{\partial w_k}$, is given in (18) and the Jacobian of the residual function with respect to the state variables, the algebraic variables, and the adjoint algebraic variables in the previous time step is

$$\frac{\partial R_{k+1}}{\partial w_k}(w_{k+1}; x_k, u_k, \hat{d}_k) = \begin{bmatrix} -\mathrm{I} & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}, \tag{27}$$

for $k = 1, \ldots, N-1$. The gradient of Φ_k is

$$\nabla_{w_{k+1}}\Phi_k = \begin{bmatrix} 0;\ \Delta t_k \nabla_y \Phi(y_{k+1}, u_k, \hat{d}_k);\ 0 \end{bmatrix}, \quad k \in \mathcal{N}. \tag{28}$$

3.2. The nested approach

We solve the algebraic equations (9b) for the algebraic variables, y_{k+1}^m, and the adjoint algebraic variables, z_{k+1}^m, for each estimate of the states, x^m_{k+1}. The algebraic equations are solved with an inexact Newton method:

$$\left[y^{l+1}_{k+1};\quad z^{l+1}_{k+1}\right] = \left[y^{l}_{k+1};\quad z^{l}_{k+1}\right] - \left(M^l_G\right)^{-1} G(x^m_{k+1}, y^l_{k+1}, z^l_{k+1}). \tag{29}$$

If $m = 0$, the initial guess for the Newton iterations, $[y^0_{k+1}; z^0_{k+1}]$, is the algebraic variables and the adjoint algebraic variables in the previous time step, $[y_k; z_k]$. Otherwise, the initial guess is the variables from the previous outer Newton iteration, $[y^{m-1}_{k+1}; z^{m-1}_{k+1}]$. The initial iteration matrix, M^0_G, is the KKT matrix (20) evaluated at the initial guess

$$M^0_G = K(x^m_{k+1}, y^0_{k+1}, z^0_{k+1}). \tag{30}$$

The iteration matrix,

$$M^{l+1}_G = K(x^m_{k+1}, y^{l+1}_{k+1}, z^{l+1}_{k+1}), \tag{31}$$

is updated if the estimates, y^{l+1}_{k+1} and z^{l+1}_{k+1}, do not sufficiently reduce the norm of the algebraic function, G, as compared to the previous estimates, y^l_{k+1} and z^l_{k+1}. This condition can be expressed as

$$||G(x^m_{k+1}, y^{l+1}_{k+1}, z^{l+1}_{k+1})|| > \tau_G ||G(x^m_{k+1}, y^l_{k+1}, z^l_{k+1})||, \tag{32}$$

where $\tau_G \in [0, 1]$. The iteration matrix is not updated (and $M^{l+1}_G = M^l_G$) if (32) is not satisfied. The Newton iterations (29) are terminated when the norm of the algebraic function, G, is smaller than a specified tolerance, ϵ_G:

$$||G(x^m_{k+1}, y^{l+1}_{k+1}, z^{l+1}_{k+1})|| < \epsilon_G. \tag{33}$$

y^m_{k+1} and z^m_{k+1} denote the estimates that satisfy the stopping criteria (33). The outer Newton iterations are terminated if x^m_{k+1} and y^m_{k+1} furthermore satisfies the discretized differential equations within a tolerance of ϵ_D:

$$||D_{k+1}(x^m_{k+1}, x_k, y^m_{k+1}, u_k, \hat{d}_k)|| < \epsilon_D. \tag{34}$$

If the stopping criteria (34) is not satisfied, the next estimate of the states, x^{m+1}_{k+1}, is computed:

$$x^{m+1}_{k+1} = x^m_{k+1} - (M^m_D)^{-1} D_{k+1}(x^m_{k+1}, x_k, y^m_{k+1}, u_k, \hat{d}_k). \tag{35}$$

The initial guess for the outer Newton iterations, x^0_{k+1}, is the states in the previous time step, x_k. The initial iteration matrix, M^0_D, is the Jacobian of D_{k+1} evaluated at the initial guess

$$M^0_D = \frac{\partial D_{k+1}}{\partial x_{k+1}}(x^0_{k+1}, x_k, y^0_{k+1}, u_k, \hat{d}_k). \tag{36}$$

y^0_{k+1} is the solution to the algebraic equations, with x^0_{k+1} as parameter, which is found with the inner Newton iterations (29). Once the outer Newton step (35) has been computed, the inner Newton iterations (29) are repeated in order to update the estimate for the algebraic variables, y^{m+1}_{k+1}, and the adjoint algebraic variables, z^{m+1}_{k+1}. The iteration matrix,

$$M^{m+1}_D = \frac{\partial D_{k+1}}{\partial x_{k+1}}(x^{m+1}_{k+1}, x_k, y^{m+1}_{k+1}, u_k, \hat{d}_k), \tag{37}$$

is updated if the estimates, x^{m+1}_{k+1} and y^{m+1}_{k+1}, do not sufficiently reduce the norm of the function D_{k+1} as compared to the previous estimates, x^m_{k+1} and y^m_{k+1}. This is expressed as the condition

$$||D_{k+1}(x^{m+1}_{k+1}, x_k, y^{m+1}_{k+1}, u_k, \hat{d}_k)|| > \tau_D ||D_{k+1}(x^m_{k+1}, x_k, y^m_{k+1}, u_k, \hat{d}_k)||, \tag{38}$$

where $\tau_D \in [0, 1]$. The iteration matrix is not updated (and $M^{m+1}_D = M^m_D$) if (38) is not satisfied. The inner and outer Newton iterations are terminated when an estimate has been reached that satisfies both the discretized differential equations and the algebraic equations, i.e. that satisfies both (33) and (34). The Jacobian of the function D_{k+1} is

$$\frac{\partial D_{k+1}}{\partial x_{k+1}} = \mathrm{I} - \Delta t_k \frac{\partial F}{\partial y}(y_{k+1}, u_k, \hat{d}_k) \frac{\partial y_{k+1}}{\partial x_{k+1}}. \tag{39}$$

The sensitivities of the algebraic variables and the adjoint algebraic variables are

$$\left[\tfrac{\partial y_{k+1}}{\partial x_{k+1}};\quad \tfrac{\partial z_{k+1}}{\partial x_{k+1}}\right] = -K(w_{k+1})^{-1} \frac{\partial G}{\partial x}(w_{k+1}). \tag{40}$$

3.2.1. The adjoint method for the nested approach

The discretized differential equations, $D_{k+1} = 0$, are substituted into the objective function, ψ, in (3):

$$\begin{aligned}
\psi &= \psi(\{u_k\}_{k\in\mathcal{N}}; \hat{x}_0, \{\hat{d}_k\}_{k\in\mathcal{N}}) && \text{(41a)}\\
&= \Big\{\phi = \sum_{k\in\mathcal{N}} \Phi_k(y_{k+1}, u_k, \hat{d}_k)\ : && \text{(41b)}\\
&\qquad x_0 = \hat{x}_0, && \text{(41c)}\\
&\qquad D_{k+1}(x_{k+1}, x_k, y_{k+1}, u_k, \hat{d}_k) = 0,\ k \in \mathcal{N}, && \text{(41d)}\\
&\qquad G(x_{k+1}, y_{k+1}, z_{k+1}) = 0,\ k \in \mathcal{N}\Big\}, && \text{(41e)}
\end{aligned}$$

where Φ_k is defined in (23). The gradients of ψ in (41) with respect to the inputs, $\{\nabla_{u_k}\psi\}_{k\in\mathcal{N}}$, are computed by solving for the adjoints, λ_N,

$$\left(\frac{\partial D_N}{\partial x_N}\right)^T \lambda_N = -\left(\frac{\partial y_N}{\partial x_N}\right)^T \nabla_{y_N}\Phi_{N-1}. \tag{42}$$

and inserting into the following expression for the gradients

$$\nabla_{u_k}\psi = \nabla_{u_k}\Phi_k + \left(\frac{\partial D_{k+1}}{\partial u_k}\right)^T \lambda_{k+1}, \quad k \in \mathcal{N}. \tag{43}$$

Each subsequent set of adjoints, λ_k, satisfies

$$\left(\frac{\partial D_k}{\partial x_k}\right)^T \lambda_k = -\left(\frac{\partial D_{k+1}}{\partial x_k}\right)^T \lambda_{k+1} - \left(\frac{\partial y_k}{\partial x_k}\right)^T \nabla_{y_k}\Phi_{k-1} \tag{44}$$

for $k = N-1, \ldots, 1$. The Jacobian of the function D_k, $\frac{\partial D_k}{\partial x_k}$, is defined in (39), and the Jacobian with respect to the states in the previous time step is

$$\frac{\partial D_{k+1}}{\partial x_k}(x_{k+1}, x_k, y_{k+1}, u_k, \hat{d}_k) = -\mathrm{I}, \quad k = 1, \ldots, N-1. \tag{45}$$

This concludes the description of the numerical methods for the computation of the objective function, ψ, defined in (3) as well as the gradients with respect to the inputs, $\{\nabla_{u_k}\psi\}_{k\in\mathcal{N}}$.

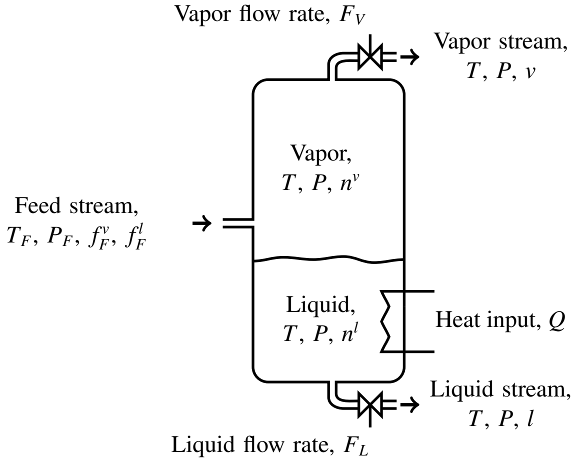


Figure 2: Sketch of a flash unit. The unit is supplied by a vapor-liquid feed stream. A vapor stream and a liquid stream extract mass from the vapor phase and the liquid phase of the mixture in the unit. The mixture is furthermore subject to external heating or cooling.

4. The dynamic UV flash model

This section describes the dynamic UV flash model, i.e. the differential-algebraic constraints (1c)-(1d) in the optimal control problem (1). Fig. 2 illustrates the flash unit. The mixture in the unit contains N_C components. We assume that two phases (vapor and liquid) always exist and that they are in chemical, thermal and mechanical equilibrium. The flash unit is continuously supplied by a feed stream at a given temperature, pressure, and composition. The mixture in the flash unit is extracted continuously through a vapor stream and a liquid stream. The total flow rate of the vapor stream, F_V, and the total flow rate of the liquid stream, F_L, are manipulated. The mixture in the unit is subject to either heating or cooling with heat flux, Q. The heat flux is also manipulated.

The following subsections describe the properties of the mixture in the unit, the feed stream, the vapor stream, and the liquid stream. We furthermore formulate the conservation equations and the equilibrium conditions. We illustrate how all needed thermodynamic properties can be computed by evaluation of the enthalpy, $H = H(T, P, n)$, the entropy, $S = S(T, P, n)$, and the volume, $V = V(T, P, n)$, as function of temperature, T, pressure, P, and composition (mole numbers), n.

4.1. The mixture in the flash unit

The flash unit contains a mixture in thermal, mechanical, and chemical vapor-liquid equilibrium. The thermal equilibrium implies that the temperature of the vapor phase, T^v, and the liquid phase, T^l, are identical, i.e. $T = T^v = T^l$. The mechanical equilibrium implies that the pressure in the vapor phase, P^v, is identical to the pressure in the liquid phase, P^l, i.e. $P = P^v = P^l$. The vapor holdup is n^v and the liquid holdup is n^l. The total composition (in moles) of the mixture is $n_i = n_i^v + n_i^l$ for $i = 1, 2, \dots, N_C$. The properties of the mixture in the flash unit are

$$N^v = \sum_{i=1}^{N_C} n_i^v, \quad N^l = \sum_{i=1}^{N_C} n_i^l, \quad N = \sum_{i=1}^{N_C} n_i,$$
$$y_i = \frac{n_i^v}{N^v}, \qquad x_i = \frac{n_i^l}{N^l}, \qquad z_i = \frac{n_i}{N}.$$

The vapor fraction is $\beta = N^v/N$. The specification of (T, P, n^v) and (T, P, n^l) also allows for the computation of the enthalpy, entropy, and volume:

$$H^v = H^v(T, P, n^v), H^l = H^l(T, P, n^l), H = H^v + H^l,$$
$$S^v = S^v(T, P, n^v), \ S^l = S^l(T, P, n^l), \ S = S^v + S^l,$$
$$V^v = V^v(T, P, n^v), \ V^l = V^l(T, P, n^l), \ V = V^v + V^l.$$

Given (H, S, V) all other thermodynamic state functions may be computed: the internal energy is $U = H - PV$, Gibbs free energy is $G = H - TS$, and Helmholtz free energy is $A = U - TS$.

4.2. The feed

The feed is a vapor-liquid mixture at temperature T_F and pressure P_F. $f_{F,i}^v$ and $f_{F,i}^l$ denote the molar vapor and liquid feed flow rates of component i. The total feed flow rate of the i'th component is $f_{F,i} = f_{F,i}^v + f_{F,i}^l$ for $i = 1, 2, \dots, N_C$, and the total flow rate is $F_F = F_F^v + F_F^l$. F_F^v and F_F^l are the total flow rates of the vapor and liquid phases of the feed. The properties of the feed stream are

$$F_F^v = \sum_{i=1}^{N_C} f_{F,i}^v, \quad F_F^l = \sum_{i=1}^{N_C} f_{F,i}^l, \quad F_F = \sum_{i=1}^{N_C} f_{F,i},$$
$$y_{F,i} = \frac{f_{F,i}^v}{F_F^v}, \qquad x_{F,i} = \frac{f_{F,i}^l}{F_F^l}, \qquad z_{F,i} = \frac{f_{F,i}}{F_F}.$$

The fraction of vapor in the feed is $\beta_F = F_F^v/F_F$. The specification of (T_F, P_F, f_F^v, f_F^l) also allows for the computation of the enthalpy, entropy, and volume of the feed stream:

$$H_F^v = H^v(T_F, P_F, f_F^v), H_F^l = H^l(T_F, P_F, f_F^l), H_F = H_F^v + H_F^l,$$
$$S_F^v = S^v(T_F, P_F, f_F^v), \ S_F^l = S^l(T_F, P_F, f_F^l), \ S_F = S_F^v + S_F^l,$$
$$V_F^v = V^v(T_F, P_F, f_F^v), \ V_F^l = V^l(T_F, P_F, f_F^l), \ V_F = V_F^v + V_F^l.$$

The feed is completely specified by (T_F, P_F, f^v, f^l) as other thermodynamic functions concerned with the feed may be computed using (H_F, S_F, V_F).

4.3. The vapor stream

The vapor outlet stream has the same temperature, T, the same pressure, P, and the same composition, y, as the vapor phase in the unit. The total flow rate of the vapor stream is F_V. This implies that the compositional flow rates are $v_i = y_i F_V$ for $i = 1, 2, \dots, N_C$. Let h^v denote the molar enthalpy, s^v denote the molar entropy, and v^v denote the molar volume of the vapor phase in the flash unit. Then the total properties of the vapor stream are $H_V = F_V h^v$, $S_V = F_V s^v$, and $V_V = F_V v^v$.

4.4. The liquid stream

The liquid outlet stream has the same temperature, T, the same pressure, P, and the same composition, x, as the liquid phase in the flash unit. F_L denotes the total flow rate of the liquid stream, and the compositional flow rates are $l_i = x_i F_L$ for $i = 1, 2, \ldots, N_C$. Let h^l denote the molar enthalpy, s^l denote the molar entropy, and v^l denote the molar volume of the liquid phase in the flash unit. The total properties of the liquid stream are then given by $H_L = F_L h^l$, $S_L = F_L s^l$, and $V_L = F_L v^l$.

4.5. The thermodynamic model and database

The total thermodynamic properties of the mixture, the feed, the vapor stream, and the liquid stream are computed from molar thermodynamic properties. The molar thermodynamic properties (h^v, s^v, v^v) of a vapor phase mixture are the functions $h^v = h^v(T, P, y)$, $s^v = s^v(T, P, y)$, and $v^v = v^v(T, P, y)$. The corresponding total properties of the vapor phase are $H^v = N^v h^v$, $S^v = N^v s^v$, and $V^v = N^v v^v$. Similarly, the molar thermodynamic properties (h^l, s^l, v^l) of a liquid phase mixture are the functions $h^l = h^l(T, P, x)$, $s^l = s^l(T, P, x)$, and $v^l = v^l(T, P, x)$. The corresponding total properties of the liquid phase are $H^l = N^l h^l$, $S^l = N^l s^l$, and $V^l = N^l v^l$. We compute the molar vapor-liquid properties (h^v, s^v, v^v) and (h^l, s^l, v^l) with a recently developed thermodynamic library, ThermoLib (Ritschel et al., 2016, 2017c). ThermoLib uses data and correlations from the DIPPR database (Thomson, 1996) together with either the SRK EOS or the PR EOS. It is implemented in both Matlab and C, and it is distributed as open-source software at `www.psetools.org`. It provides routines for evaluation of enthalpy, entropy, and volume. The routines furthermore provide first and second order derivatives with respect to temperature, pressure, and composition (in moles).

4.6. The equilibrium constraint - vapor-liquid equilibrium

Vapor-liquid equilibrium processes may be formulated as optimization problems (Michelsen, 1999). When the internal energy, U, and the volume, V, are specified, the optimization problem determining the equilibrium temperature, pressure, and vapor-liquid compositions is called the UV flash problem. The UV flash optimization problem is

$$\max_{T,P,n^v,n^l} \quad S = S^v(T,P,n^v) + S^l(T,P,n^l) \tag{48a}$$

$$\text{s.t.} \quad U^v(T,P,n^v) + U^l(T,P,n^l) = U, \tag{48b}$$

$$V^v(T,P,n^v) + V^l(T,P,n^l) = V, \tag{48c}$$

$$n_i^v + n_i^l = n_i, \quad i = 1, \ldots, N_C. \tag{48d}$$

U is the specified internal energy, and V is the total fixed volume of the flash unit. n_i is the specified total amount (in moles) of component i in the flash unit. The optimization problem (48) is a mathematical statement of the second law of thermodynamics which states that the entropy of a closed system at equilibrium is maximal. The UV flash optimization problem (48) is in the form of the optimization problem (5) where

$$f(y) = f(T,P,n^v,n^l) = -\left(S^v(T,P,n^v) + S^l(T,P,n^l)\right), \tag{49a}$$

$$g(y) = g(T,P,n^v,n^l) = \begin{bmatrix} U^v(T,P,n^v) + U^l(T,P,n^l) \\ n^v + n^l \end{bmatrix}, \tag{49b}$$

$$h(y) = h(T,P,n^v,n^l) = V^v(T,P,n^v) + V^l(T,P,n^l) - V. \tag{49c}$$

The state variables, x, the algebraic variables, y, and the adjoint algebraic variables, z, are

$$x = [U; n] \in \mathbb{R}^{1+N_C}, \tag{50a}$$

$$y = \left[T; P; n^v; n^l\right] \in \mathbb{R}^{2+2N_C}, \tag{50b}$$

$$z = [\mu; \eta] \in \mathbb{R}^{2+N_C}. \tag{50c}$$

The UV flash problem is sometimes referred to as the UVn flash problem to indicate that U, V, and n are specified. It is also known as the isoenergetic-isochoric flash (constant energy – constant volume).

4.7. The differential equations - conservation equations

The mass and energy conservation equations that describe the temporal evolution of the internal energy and the total holdup are

$$\dot{U}(t) = H_F^v(t) + H_F^l(t) - H_V(t) - H_L(t) + Q(t), \tag{51a}$$

$$\dot{n}_i(t) = f_{F,i}^v(t) + f_{F,i}^l(t) - v_i(t) - l_i(t), \quad i = 1, .., N_C. \tag{51b}$$

U is the internal energy, and n_i is the total holdup of component i. H_V and H_L are the enthalpies of the vapor and liquid streams, respectively. H_F^v and H_F^l are the vapor and liquid enthalpies of the feed. $f_{F,i}^v$ and $f_{F,i}^l$ are the molar vapor and liquid feed flow rates. v_i and l_i are the molar vapor and liquid stream flow rates. The conservation equations (51) are in the form of the differential equations (1d) where

$$F(y(t), u(t), d(t)) = \begin{bmatrix} H_F^v(t) + H_F^l(t) - H_V(t) - H_L(t) + Q(t) \\ f_F^v(t) + f_F^l(t) - v(t) - l(t) \end{bmatrix}. \tag{52}$$

The algebraic variables, y, are defined in (50), and the controls (manipulated variables), u, and the disturbance variables, d, are

$$u = [Q; F_V; F_L] \in \mathbb{R}^3, \tag{53a}$$

$$d = \left[T_F; P_F; f_F^v; f_F^l\right] \in \mathbb{R}^{2+2N_C}. \tag{53b}$$

5. The relation between the UV flash and the PT flash

In this section, we demonstrate that the solution of the UV flash is identical to the solution of a PT flash with constraints on the internal energy, U, and the volume, V. The solution to the PT flash problem is the vapor-liquid composition that minimizes Gibbs energy at a given temperature and pressure while satisfying a mass balance constraint. The PT flash is the most common type of flash because it can be formulated as an unconstrained minimization problem and therefore solved efficiently.

5.1. The UV flash

The UV flash maximization problem (48) is equivalent to the following minimization problem

$$
\begin{aligned}
\min_{T,P,n^v,n^l} \quad & -S = -\left(S^v(T,P,n^v) + S^l(T,P,n^l)\right), && (54a)\\
s.t. \quad & U^v(T,P,n^v) + U^l(T,P,n^l) = U, && (54b)\\
& V^v(T,P,n^v) + V^l(T,P,n^l) = V, && (54c)\\
& n_i^v + n_i^l = n_i, \quad i = 1,\dots,N_C. && (54d)
\end{aligned}
$$

The solution to the UV flash problem is the temperature, pressure, and vapor-liquid composition of a closed system at equilibrium, (T,P,n^v,n^l), with a given internal energy, volume, and total composition. The Lagrange function of the UV flash problem (54) is

$$
\begin{aligned}
\mathcal{L} = & -\left(S^v(T,P,n^v) + S^l(T,P,n^l)\right)\\
& -\kappa\left(U^v(T,P,n^v) + U^l(T,P,n^l) - U\right)\\
& -\lambda\left(V^v(T,P,n^v) + V^l(T,P,n^l) - V\right) \qquad (55)\\
& -\sum_{i=1}^{N_C} \bar{\mu}_i \left(n_i^v + n_i^l - n_i\right).
\end{aligned}
$$

κ is the Lagrange multiplier associated with the internal energy constraint (54b), λ is the Lagrange multiplier associated with the volume constraint (54c), and $\{\bar{\mu}_i\}_{i=1}^{N_C}$ are the Lagrange multipliers associated with the mole balances (54d). In this section, we use the notation $S^{vl} = S^v + S^l$, $U^{vl} = U^v + U^l$, and $V^{vl} = V^v + V^l$ to distinguish these properties from the specified values U and V. The first order optimality conditions for the UV flash minimization problem (54) are

$$
\begin{aligned}
&\frac{\partial \mathcal{L}}{\partial T} = -\frac{\partial S^{vl}}{\partial T} - \kappa\frac{\partial U^{vl}}{\partial T} - \lambda\frac{\partial V^{vl}}{\partial T} = 0, && (56a)\\
&\frac{\partial \mathcal{L}}{\partial P} = -\frac{\partial S^{vl}}{\partial P} - \kappa\frac{\partial U^{vl}}{\partial P} - \lambda\frac{\partial V^{vl}}{\partial P} = 0, && (56b)\\
&\frac{\partial \mathcal{L}}{\partial n_i^v} = -\frac{\partial S^v}{\partial n_i^v} - \kappa\frac{\partial U^v}{\partial n_i^v} - \lambda\frac{\partial V^v}{\partial n_i^v} - \bar{\mu}_i = 0,\ i = 1,\dots,N_C, && (56c)\\
&\frac{\partial \mathcal{L}}{\partial n_i^l} = -\frac{\partial S^l}{\partial n_i^l} - \kappa\frac{\partial U^l}{\partial n_i^l} - \lambda\frac{\partial V^l}{\partial n_i^l} - \bar{\mu}_i = 0,\ i = 1,\dots,N_C, && (56d)\\
&U^{vl}(T,P,n^v,n^l) = U^v(T,P,n^v) + U^l(T,P,n^l) = U, && (56e)\\
&V^{vl}(T,P,n^v,n^l) = V^v(T,P,n^v) + V^l(T,P,n^l) = V, && (56f)\\
&n_i^v + n_i^l = n_i, \quad i = 1,\dots,N_C. && (56g)
\end{aligned}
$$

(56a)-(56b) are the stationarity conditions associated with the temperature and pressure gradients. (56c)-(56d) are the stationarity conditions associated with the component derivatives, and (56e)-(56g) are the feasibility conditions. The first order optimality conditions (56) of the equality constrained optimization problem (54) are necessary conditions for a minimizer of (54).

5.2. The UV constrained PT flash

The solution to the PT flash problem, (n^v, n^l), minimizes Gibbs free energy, G, subject to a mass balance constraint

$$
\begin{aligned}
\min_{n^v,n^l} \quad & G = G^v(T,P,n^v) + G^l(T,P,n^l) && (57a)\\
s.t. \quad & n_i^v + n_i^l = n_i, \quad i = 1,\dots,N_C. && (57b)
\end{aligned}
$$

The Lagrange function of the PT flash problem (57) is

$$
\mathcal{L} = \left(G^v(T,P,n^v) + G^l(T,P,n^l)\right) - \sum_{i=1}^{N_C} \mu_i(n_i^v + n_i^l - n_i), \qquad (58)
$$

where $\{\mu_i\}_{i=1}^{N_C}$ denote the Lagrange multipliers associated with the mole balances (57b). The first order optimality conditions for the PT flash are

$$
\begin{aligned}
&\frac{\partial \mathcal{L}}{\partial n_i^v} = \frac{\partial G^v}{\partial n_i^v} - \mu_i = 0, \quad i = 1,\dots,N_C, && (59a)\\
&\frac{\partial \mathcal{L}}{\partial n_i^l} = \frac{\partial G^l}{\partial n_i^l} - \mu_i = 0, \quad i = 1,\dots,N_C, && (59b)\\
&n_i^v + n_i^l = n_i, \qquad i = 1,\dots,N_C. && (59c)
\end{aligned}
$$

(59a)-(59b) are the stationarity conditions, and (59c) are the feasibility conditions. The first order optimality conditions (59) are necessary conditions for a minimizer of the PT flash (57). The PT flash provides the equilibrium composition of the vapor phase, n^v, and the liquid phase, n^l, at given temperature, T, and pressure, P. We combine the PT flash problem (57) with the following constraints on the internal energy, U, and the volume, V,

$$
\begin{aligned}
&U^{vl}(T,P,n^v,n^l) = U^v(T,P,n^v) + U^l(T,P,n^l) = U, && (60a)\\
&V^{vl}(T,P,n^v,n^l) = V^v(T,P,n^v) + V^l(T,P,n^l) = V. && (60b)
\end{aligned}
$$

The constraints (60) are identical to the constraints (54b)-(54c) in the UV optimization problem. A point (T,P,n^v,n^l) satisfying the PT flash (57) and the UV constraints (60) must satisfy the conditions (59) and (60).

5.3. Equivalence of the UV flash and the UV-constrained PT flash

We show that the first order optimality conditions (56) of the UV flash (54) are equivalent to the first order optimality conditions (59) of the PT flash (57) in combination with the UV constraints (60). This shows that the solution of the UV flash problem (54) is equivalent to the solution of the PT flash problem (57) combined with the specification of internal energy, U, and volume, V, in (60). The feasibility conditions (56e)-(56g) for the UV flash (54) are equivalent to the feasibility conditions (59c) for the PT flash (57) and the UV constraints (60). The thermodynamic relations $G = H - TS$ and $H = U + PV$ imply that $G = U + PV - TS$ such that

$$
\begin{aligned}
&\frac{\partial G^v}{\partial n_i^v} = \frac{\partial U^v}{\partial n_i^v} + P\frac{\partial V^v}{\partial n_i^v} - T\frac{\partial S^v}{\partial n_i^v}, \quad i = 1,\dots,N_C, && (61a)\\
&\frac{\partial G^l}{\partial n_i^l} = \frac{\partial U^l}{\partial n_i^l} + P\frac{\partial V^l}{\partial n_i^l} - T\frac{\partial S^l}{\partial n_i^l}, \quad i = 1,\dots,N_C. && (61b)
\end{aligned}
$$

Therefore, the stationarity conditions (59a)-(59b) of the PT flash (57) may be reformulated using (61):

$$\frac{\mu_i}{T} = -\frac{\partial S^v}{\partial n_i^v} - \left(-\frac{1}{T}\right)\frac{\partial U^v}{\partial n_i^v} - \left(-\frac{P}{T}\right)\frac{\partial V^v}{\partial n_i^v}, \quad i = 1, \dots, N_C, \tag{62a}$$

$$\frac{\mu_i}{T} = -\frac{\partial S^l}{\partial n_i^l} - \left(-\frac{1}{T}\right)\frac{\partial U^l}{\partial n_i^l} - \left(-\frac{P}{T}\right)\frac{\partial V^l}{\partial n_i^l}, \quad i = 1, \dots, N_C. \tag{62b}$$

The stationarity conditions (62) for the PT flash (57) are equivalent to the stationarity conditions (56c)-(56d) for the UV flash (54) provided that the Lagrange multipliers $(\kappa, \lambda, \{\bar{\mu}_i\}_{i=1}^{N_C})$ are

$$\kappa = -1/T, \tag{63a}$$

$$\lambda = -P/T, \tag{63b}$$

$$\bar{\mu}_i = \mu_i/T, \quad i = 1, \dots, N_C. \tag{63c}$$

The fundamental thermodynamic relations (Smith et al., 2005, Chap. 6)

$$\frac{\partial S}{\partial T} = \frac{C_P}{T}, \tag{64a}$$

$$\frac{\partial U}{\partial T} = C_P - P\frac{\partial V}{\partial T}, \tag{64b}$$

may be used to demonstrate that (56a) is satisfied when $\kappa = -1/T$ and $\lambda = -P/T$. Similarly, the fundamental thermodynamic relations (Smith et al., 2005, Chap. 6)

$$\frac{\partial S}{\partial P} = -\frac{\partial V}{\partial T}, \tag{65a}$$

$$\frac{\partial U}{\partial P} = -T\frac{\partial V}{\partial T} - P\frac{\partial V}{\partial P}, \tag{65b}$$

may be used to demonstrate that (56b) is satisfied when $\kappa = -1/T$ and $\lambda = -P/T$. Alternatively, the relations (64)-(65) may be substituted into (56a)-(56b) and the resulting 2×2 linear system of equations may be solved for κ and λ to obtain $\kappa = -1/T$ and $\lambda = -P/T$.

5.4. *Significance*

We have demonstrated that the solution of the PT flash (57) in combination with the UV constraints (60) is identical to the solution of the UV flash (54). This implies that existing efficient algorithms for the PT flash may be combined with the UV constraints to obtain a solution that is identical to the solution of the UV flash problem (54). Similarly, the Lagrange multipliers of the UV flash problem may be computed from the UV constraints and the PT flash using (63). The KKT matrix (20) of the UV flash problem (54) can thus be obtained without directly solving the UV flash problem.

6. Implementation

This section briefly describes the transformations that are made to the vapor-liquid equilibrium constraints in the implementation of the single-shooting algorithm. We furthermore discuss the details of the different software libraries and compilers that are used in the implementation.

6.1. *Transformation of the UV flash*

In the implementation, the optimality conditions (56) for the UV flash are replaced by the optimality conditions (59) for the PT flash combined with the UV constraints (60). We furthermore eliminate variables and scale equations. The resulting algebraic equations are similar to the equations that are solved in the Newton approach by Michelsen and Mollerup (2007, Chap. 14). We subtract (59b) from (59a) in order to eliminate the Lagrange multipliers, $\{\mu_i\}_{i=1}^{N_C}$, from the first order optimality conditions (59) for the PT flash. Furthermore, we eliminate the vapor mole numbers, n^v, from the linear mass balance constraints (57b):

$$n_i^v = n_i^v(n_i, n_i^l) = n_i - n_i^l, \quad i = 1, \dots, N_C. \tag{66}$$

The first order optimality conditions (59) are thus transformed to

$$\frac{\partial G^l}{\partial n_i^l}(T, P, n^l) - \frac{\partial G^v}{\partial n_i^v}(T, P, n^v) = 0, \quad i = 1, \dots, N_C, \tag{67}$$

which can be solved for the liquid mole numbers n^l. We rewrite the internal energy constraint (60a) using the thermodynamic relations $U^v = H^v - PV^v$ and $U^l = H^l - PV^l$:

$$\begin{aligned} &U - H^v(T, P, n^v) + PV^v(T, P, n^v) \\ &\quad - H^l(T, P, n^l) + PV^l(T, P, n^l) = 0. \end{aligned} \tag{68}$$

We furthermore substitute the volume constraint (60b):

$$U + PV - H^v(T, P, n^v) - H^l(T, P, n^l) = 0. \tag{69}$$

The optimality conditions (67) and the internal energy constraint (69) are scaled with $1/(RT)$. The volume constraint (60b) is scaled with $P/(RT)$:

$$\frac{1}{RT}\left(\frac{\partial G^l}{\partial n_i^l}(T, P, n^l) - \frac{\partial G^v}{\partial n_i^v}(T, P, n^v)\right) = 0, \tag{70a}$$

$$\frac{1}{RT}\left(U + PV - H^l(T, P, n^l) - H^v(T, P, n^v)\right) = 0, \tag{70b}$$

$$\frac{P}{RT}\left(V^l(T, P, n^l) + V^v(T, P, n^v) - V\right) = 0. \tag{70c}$$

The internal energy, U, and the volume, V, are specified, and the algebraic equations (70) can be solved for the temperature, T, the pressure, P, and the liquid mole numbers, n^l. The temperature and pressure are, however, large in magnitude as compared to the mole numbers. The implementation therefore uses logarithmic values of the temperature and pressure as algebraic variables:

$$y = \left[\ln T; \ln P; n^l\right] \in \mathbb{R}^{2+N_C}. \tag{71}$$

The algebraic equations (70) are written compactly

$$G(x, y) = 0, \tag{72}$$

where x is defined in (50a).

6.2. Units of variables and scaling

Large differences in the elements of Jacobian matrices can lead to ill-conditioned linear systems of equations in the numerical solution schemes. Such ill-conditioning can furthermore lead to imprecise results. Several quantities are therefore not in SI-units in the implementation. The unit of time, t, is hr, and the flow rates, F_F, F_V, and F_L, are given in kmol/hr. The compositional flow rates, $f^v_{F,i}$, $f^l_{F,i}$, v_i, and l_i for $i = 1, \dots, N_C$, are also in kmol/hr. Mole numbers, i.e. n_i, n^v_i, and n^l_i for $i = 1, \dots, N_C$, are given in kmol. The internal energy, U, and the enthalpies are in MJ, and the heat input, Q, is in MJ/hr.

6.3. Software libraries, compilers, and hardware

The single-shooting algorithm is implemented in Matlab. The algorithm is implemented with both a simultaneous approach and a nested approach for the solution of the dynamic UV flash model equations. The Matlab implementation uses Matlab routines from ThermoLib as well as Matlab's function for constrained nonlinear optimization, fmincon. The single-shooting algorithm, with a simultaneous approach, is also implemented in C. The C implementation uses C routines from ThermoLib as well as either of four solvers for nonlinear constrained optimization: Matlab's fmincon, the open-source IPOPT 3.12.4 (Wächter and Biegler, 2006), the commercial NPSOL 5.0, or the commercial KNITRO 10.2. We use fmincon's and KNITRO's interior point (IP) barrier methods. IPOPT also uses an IP barrier method, and NPSOL uses an active set sequential quadratic programming (SQP) method. We compare the performance obtained with GCC compilers and with Intel compilers. We use Netlib's linear algebra libraries (BLAS and LAPACK) together with the GCC compilers, and we use Intel's linear algebra library (Intel MKL) with the Intel compilers. ThermoLib and IPOPT are compiled with a C compiler, and NPSOL is compiled with a Fortran compiler. KNITRO is compiled prior to purchase. The performance study in Section 8 is carried out on a 64-bit workstation with 15.6 GB memory and four Intel Core i7 3.60 GHz cores. Each core has 64 kB of level 1 cache and 256 kB of level 2 cache. The workstation has a shared level 3 cache of 8,192 kB.

7. Optimal control examples

This section presents two optimal control problems. The first is a tracking-type control problem where an ideal thermodynamic model is used. This problem has 144 decision variables. The second is an economical optimal control problem where a nonideal thermodynamic model is used. This problem has 864 decision variables.

7.1. Optimal tracking example

In this example, we consider a flash unit with a volume of 10 m^3. The unit processes a mixture of benzene, toluene, and diphenyl. We use an ideal thermodynamic model for the computation of thermodynamic properties. This is reasonable because these three components are very similar and because we consider high ranges of temperature and low ranges of pressure. The time horizon is $[t_0, t_f] = [0 \text{ hr}, 4 \text{ hr}]$. The initial condition is a steady state which is described in Appendix A. The feed rate, F_F, increases by 50% at time $t = 2$ hr. The length of the control intervals is $\Delta t_k = 5$ minutes for $k \in \mathcal{N}$. That is a total of 48 control intervals during the 4 hr time horizon. The controls are: the heat input, Q, the total vapor flow rate, F_V, and the total liquid flow rate, F_L. The objective is to keep the temperature, pressure, and liquid volume at desired setpoints.

7.1.1. Objective function

The objective function, ϕ, is the integral of the squared differences between the controlled variables and the setpoints:

$$\phi = \int_{t_0}^{t_f} \Big[\alpha_T \left(\ln T - \ln T^{set} \right)^2 + \alpha_P \left(\ln P - \ln P^{set} \right)^2 + \alpha_{V^l} \left(V^l - V^{l,set} \right)^2 \Big] dt. \tag{73}$$

The controlled variables include logarithmic temperature and pressure because of the choice of algebraic variables described in Section 6. The weights $(\alpha_T, \alpha_P, \alpha_{V^l})$ determine the relative importance of following each setpoint. The objective function (73) is supplemented with a regularization term that penalizes: (a) the temporal change in the controls and (b) the difference between the controls in the first control interval and a set of initial reference controls: $Q_{-1} = -1$ MJ/hr, $F_{L,-1} = 0.6$ kmol/hr and $F_{V,-1} = 0.4$ kmol/hr. The discrete objective is augmented with this regularization term:

$$\hat{\Phi}_k = \Phi_k + \Big[\alpha_Q \left(Q_k - Q_{k-1} \right)^2 + \alpha_{F_V} \left(F_{V,k} - F_{V,k-1} \right)^2 + \alpha_{F_L} \left(F_{L,k} - F_{L,k-1} \right)^2 \Big] \Delta t_k, \quad k \in \mathcal{N}. \tag{74}$$

The weights $(\alpha_Q, \alpha_{F_V}, \alpha_{F_L})$ determine the size of the penalty of temporal changes in the controls. The weights are $(\alpha_T, \alpha_P, \alpha_{V^l}) = (2000, 20, 2000)$ and $(\alpha_Q, \alpha_{F_V}, \alpha_{F_L}) = (0.05, 10, 10)$. The derivatives of the additional regularization terms are derived analytically and added to the derivatives that are computed with the adjoint algorithm.

7.1.2. Constraints

The sum of the vapor and liquid stream flow rates is bounded from above by a factor times the feed flow rate:

$$F_{V,k} + F_{L,k} \le \alpha_F F_{F,k}, \quad k \in \mathcal{N}. \tag{75}$$

The factor is $\alpha_F = 1.2$. Furthermore, the controls are subject to the following bound constraints

$$Q_k \in [-60 \text{ MJ/hr}, 10 \text{ MJ/hr}], \tag{76a}$$
$$F_{L,k} \in [0.1 \text{ kmol/hr}, 1.5 \text{ kmol/hr}], \tag{76b}$$
$$F_{V,k} \in [0.1 \text{ kmol/hr}, 1.5 \text{ kmol/hr}], \tag{76c}$$

for $k \in \mathcal{N}$. The derivatives of the linear constraints (75) and the bound constraints (76) are derived analytically.

7.1.3. Optimal control strategy

Fig. 3(a) shows the controlled variables and the vapor fraction obtained with the optimal strategy (blue solid) and with a reference strategy (green dashed). A step occurs in the setpoints (black dashed) at time $t = 2$ hr. The controlled variables successfully follow the setpoints both before and after the step. The large variations in the temperature and pressure significantly affect the vapor-liquid equilibrium. This is evident from the large variations in the vapor fraction and in the vapor-liquid composition shown in Fig. 4. Fig. 3(b) shows the optimal strategy, the reference strategy, and the bound constraints (red dashed). The reference strategy is constructed such that the controlled variables eventually reach the setpoints after the step. However, the transition is significantly faster with the optimal strategy. This is because the optimal strategy uses the information about the step in the setpoints to initiate the transition already around $t = 1$ hr.

7.2. Minimal cooling example

In this example, we consider a flash unit with a volume of 1 m^3. The unit processes a mixture of methane, ethane, propane, n-heptane, and hydrogen sulfide (H_2S). We use a non-ideal thermodynamic model based on the PR EOS to compute thermodynamic properties. The time horizon is $[t_0,\ t_f] =$ [0 hr, 24 hr]. The initial condition is a steady state which is described in Appendix A. The feed composition changes at time $t = 12$ hr such that the total feed mole fraction of H_2S increases from 2.0% to 6.0%. The length of the control intervals is $\Delta t_k = 5$ minutes for $k \in \mathcal{N}$, which gives a total of 288 control intervals during the 24 hr time horizon. The objective is to minimize the energy consumption (in terms of cooling). Meanwhile, an upper bound on the H_2S vapor mole fraction should be satisfied.

7.2.1. Objective function

Since the heat input is negative (non-positive), $Q \leq 0$, in the case of cooling, the objective function is

$$\phi = -\int_{t_0}^{t_f} Q\, dt. \tag{77}$$

We incorporate bound constraints on the H_2S vapor mole fraction, $y_{\mathrm{H_2S}}$, and on the overall vapor fraction, β, into the objective function using logarithmic barrier functions. The H_2S vapor mole fraction is bounded from above: $y_{\mathrm{H_2S},k} \leq y^{\max}_{\mathrm{H_2S},k}$ for $k = 1, \ldots, N$. The overall vapor fraction is bounded from above and below: $\epsilon_\beta \leq \beta_k \leq 1 - \epsilon_\beta$ for $k = 1, \ldots, N$. The discrete objective function is augmented with the logarithmic barrier functions in the adjoint algorithm:

$$\begin{aligned}\Phi_k = &-Q_k \Delta t_k - \alpha_{\mathrm{H_2S}} \ln\left(y^{\max}_{\mathrm{H_2S},k+1} - y_{\mathrm{H_2S},k+1}\right)\\ &- \alpha_\beta \left[\ln\left(\beta_{k+1} - \epsilon_\beta\right) + \ln\left((1-\epsilon_\beta) - \beta_{k+1}\right)\right],\end{aligned} \tag{78}$$

for $k \in \mathcal{N}$. The first term is a term in the discrete approximation of the objective function (77). The second term represents the upper bound on the H_2S vapor mole fraction, and the third term represents the bounds on the vapor fraction. The weights are $(\alpha_{\mathrm{H_2S}}, \alpha_\beta) = (1.0, 0.6)$, and the vapor fraction threshold is $\epsilon_\beta = 0.05$. The upper bound on the H_2S vapor mole fraction is

$$y^{\max}_{\mathrm{H_2S},k} = \begin{cases} 0.02, & k = 1, \ldots, N/2, \\ 0.04, & k = N/2 + 1, \ldots, N, \end{cases} \tag{79}$$

i.e. the upper bound is 2% during the first 12 hrs and 4% during the last 12 hrs. The objective function (78) is furthermore augmented with a regularization term that penalizes the temporal change in the controls:

$$\begin{aligned}\hat{\Phi}_k = \Phi_k + \Big[&\alpha_Q \left(Q_k - Q_{k-1}\right)^2 + \alpha_{F_V}\left(F_{V,k} - F_{V,k-1}\right)^2\\ &+ \alpha_{F_L}\left(F_{L,k} - F_{L,k-1}\right)^2\Big]\Delta t_k, \quad k = 1, \ldots, N-1.\end{aligned} \tag{80}$$

The weights $(\alpha_Q, \alpha_{F_V}, \alpha_{F_L})$ determine the size of the penalty of the temporal change in each control variable. The weights are $(\alpha_Q, \alpha_{F_V}, \alpha_{F_L}) = (0.1, 10, 10)$. The derivatives of this additional regularization term are derived analytically. In this problem, there is no set of initial reference controls as was the case in the optimal tracking problem in Section 7.1.

7.2.2. Constraints

The total mass supplied by the feed should be equal to the total mass extracted by the vapor and liquid streams:

$$F_{V,k} + F_{L,k} = F_{F,k}, \qquad k \in \mathcal{N}. \tag{81}$$

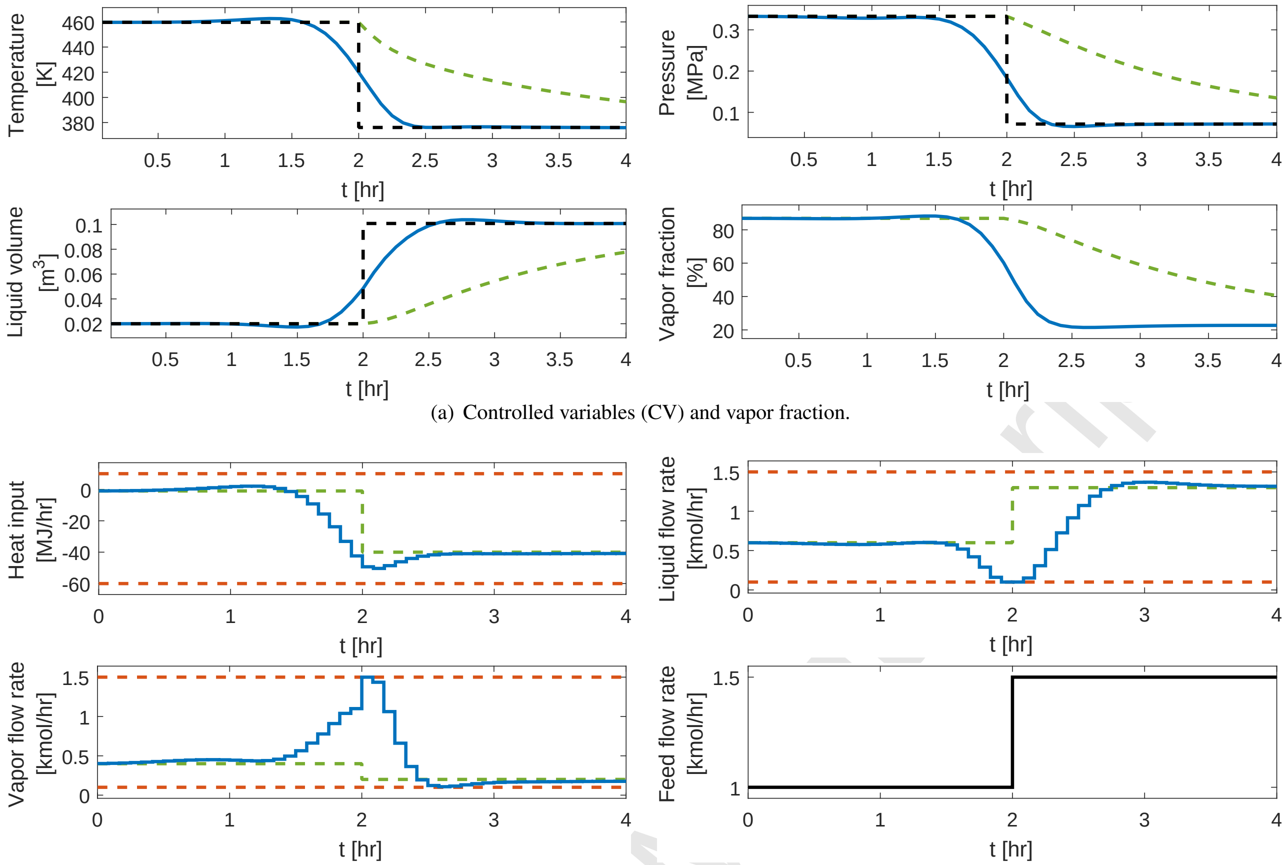
Furthermore, the controls are subject to the following bound constraints

$$Q_k \in [-150\ \text{MJ/hr}, 0\ \text{MJ/hr}], \tag{82a}$$
$$F_{L,k} \in [4\ \text{kmol/hr}, 6\ \text{kmol/hr}], \tag{82b}$$
$$F_{V,k} \in [6\ \text{kmol/hr}, 8\ \text{kmol/hr}], \tag{82c}$$

for $k \in \mathcal{N}$. The derivatives of the linear constraints (81) and the bound constraints (82) are derived analytically.

7.2.3. The optimal strategy

Fig. 5(a) shows the energy consumption and the H_2S vapor mole fraction obtained with the optimal strategy (blue solid) and with a reference strategy (green dashed). The optimal strategy reduces the total energy consumption from 2,400 MJ to around 1,700 MJ. This is a reduction of 26%. The upper bound on the H_2S vapor mole fraction is satisfied at all times. Fig. 5(b) shows the temperature, the pressure, the liquid volume, and the vapor fraction. The large variations in the temperature and pressure cause large variations in the vapor fraction. However, the vapor-liquid composition is not affected significantly as can be seen from Fig. 6. The vapor consists primarily of methane, and the liquid phase is primarily heptane and methane. Fig. 5(c) shows the optimal strategy and the reference strategy. The optimal strategy uses the vapor flow rate and the liquid flow rate to compensate for the increase in the heat input during the first 12 hrs. However, the optimal strategy does not increase the



(a) Controlled variables (CV) and vapor fraction.

(b) Inputs and feed flow rate.

Figure 3: The optimal tracking of temperature, pressure, and liquid volume setpoints for a ternary mixture containing benzene, toluene, and diphenyl. Blue solid: Optimal strategy. Green dashed: Reference strategy. Red dash-dotted: Bounds. Black solid: Feed. Black dashed: Setpoints.

heat input during the last 12 hrs. This is because the upper constraint on the H_2S vapor mole fraction is incorporated into the objective function. This undesired behavior can be avoided by adjusting the parameter α_{H_2S} in the barrier function during the iterations of the optimization algorithm.

8. Performance study

This section presents a performance study of several implementations of the single-shooting algorithm. The algorithm is used to solve the optimal tracking problem described in Section 7.1 and the minimal cooling problem described in Section 7.2. We compare the efficiency of: (a) the simultaneous and the nested approach, (b) the exact and the inexact Newton method, (c) Matlab and C implementations, (d) different NLP solvers, and (e) different compilers and linear algebra libraries.

8.1. The simultaneous and the nested approaches

Table 1 shows several performance measures for a Matlab implementation of the single-shooting algorithm. The simultaneous approach is faster than the nested approach for both problems and regardless of whether an exact or an inexact Newton method is used. This is due to fewer factorizations, back substitutions, and evaluations of thermodynamic properties. Table 2 shows the computation times of thermodynamic function evaluations, factorizations, and back substitutions. The evaluation of thermodynamic functions and their derivatives is more time consuming than solving linear systems for both the Matlab and the C implementations. However, the difference is significantly larger for the Matlab implementations. There is little or no computational overhead in evaluating thermodynamic functions for the Jacobian instead of for the residual equations when an ideal thermodynamic model is used. This is the case for the optimal tracking problem. The overhead is significantly larger when equations of state are used in the thermodynamic computations, as is the case for the minimal cooling problem.

8.2. Exact and inexact Newton methods

For both problems, it is more efficient to use an exact Newton method in the nested approach. This is likely due to frequent updating of the Jacobian in the inner Newton iterations. For the simultaneous approach, there is no advantage in using an inexact Newton method when solving the optimal tracking problem. This is because the overhead associated with evaluating thermodynamic properties for the Jacobians is very small. This is not true when equations of state are used. For the minimal cooling problem, where equations of state are used, the performance of the simultaneous approach thus improves by roughly 35% when an inexact Newton method is used. This improvement is

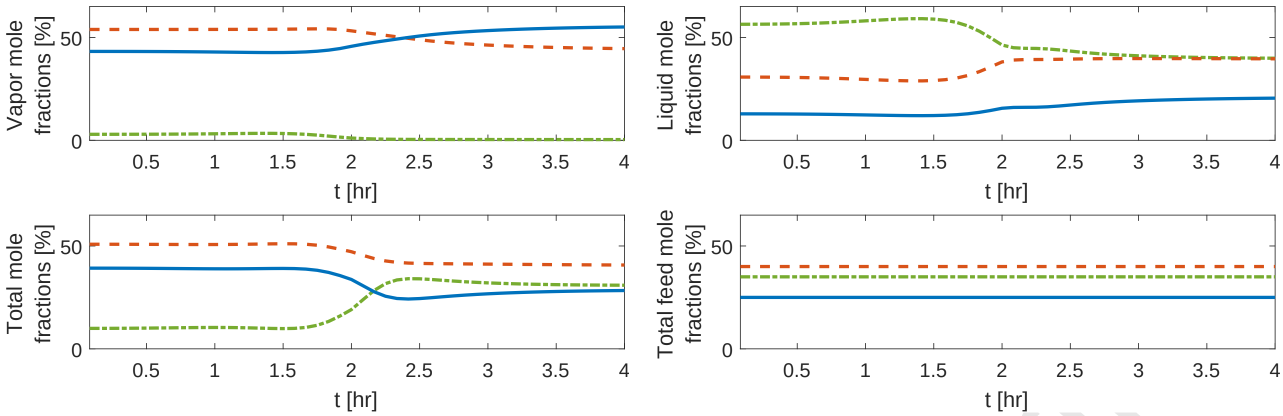


Figure 4: The mixture mole fractions and the total feed mole fractions obtained with the optimal tracking strategy. Blue solid: Benzene. Red dashed: Toluene. Green dash-dotted: Diphenyl.

also, in part, due to the fewer function evaluations in the NLP algorithm.

8.3. *Programming language, NLP solvers, and compilers*

Table 3 shows performance measures for several implementations of the single-shooting algorithm using the simultaneous approach with an inexact Newton method. The implementations use different optimization libraries (NLP solvers), compilers, and linear algebra libraries. The least efficient is a pure Matlab implementation using fmincon. This implementation solves the optimal tracking problem in 8.5 s and the minimal cooling problem in 137.8 s. A mixed implementation, which uses fmincon and C routines for the simulation and the gradient computations, improves over the pure Matlab implementation. It attains a speedup of 10 for the optimal tracking problem and 3.7 for the minimal cooling problem, as compared to the pure Matlab implementation.

The C implementation using IPOPT is not efficient for the optimal tracking problem. This is because IPOPT is used with a limited-memory BFGS approximation. This implementation is more efficient for the larger minimal cooling problem where it performs as well as the implementations using KNITRO and NPSOL when GCC compilers are used. The implementation that uses NPSOL is in general the fastest. Its performance improves significantly when Intel compilers are used. It thus attains a speedup of 83 for the optimal tracking problem and 55 for the minimal cooling problem. The Intel compilers do not improve the performance of the other implementations significantly.

9. Conclusions

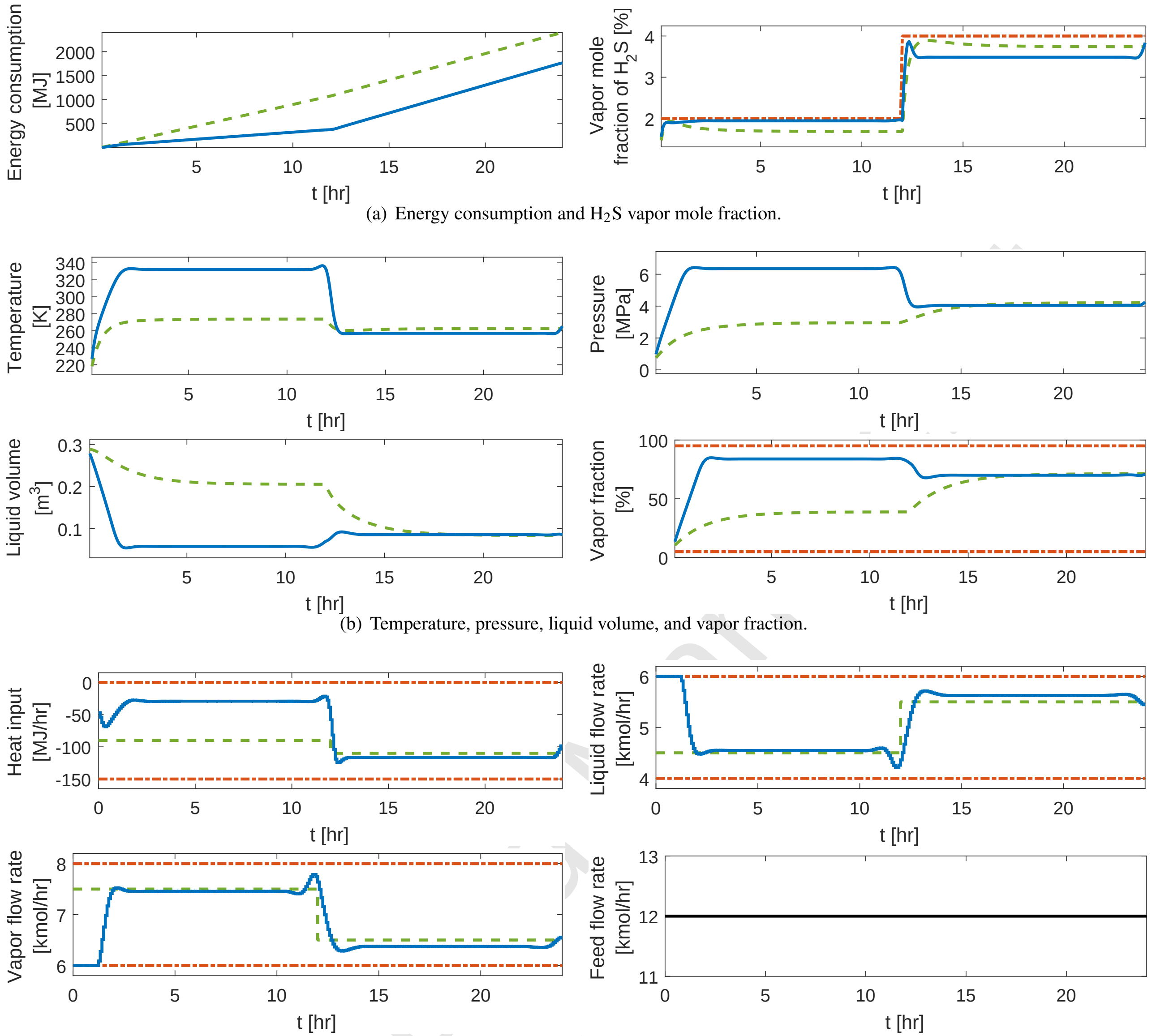
We have presented a single-shooting algorithm for gradient-based dynamic optimization of vapor-liquid equilibrium processes. It uses the adjoint method for the computation of gradients. We have described a dynamic UV flash model and demonstrated that the UV flash is equivalent to a UV-constrained PT flash, i.e. a PT flash combined with specifications of the internal energy and volume. We have presented a numerical performance study with two optimal control problems. The first problem is a small tracking-type control problem. The second is a medium-size economical control problem. We have compared the efficiency of a simultaneous approach and a nested approach for the numerical solution of the differential-algebraic constraints in the optimal control problem. The simultaneous approach is faster because it uses fewer factorizations, back substitutions, and thermodynamic function evaluations. Not surprisingly, the performance study also shows that the C implementations are significantly faster than a pure Matlab implementation. Furthermore, we have compared the performance of four nonlinear constrained optimization solvers: Matlab's fmincon, IPOPT, KNITRO, and NPSOL. NPSOL is generally the fastest solver for this problem formulation, and the performance of NPSOL is significantly improved when Intel compilers are used. The Intel compilers do not improve the efficiency of the implementations using fmincon, IPOPT, and KNITRO significantly. The implementation using NPSOL is able to solve the tracking problem in 0.1 s which is 83.4 times faster than the pure Matlab implementation. Furthermore, it solves the economical control problem in 2.5 seconds which is 55.5 times faster than the Matlab implementation.

10. Acknowledgements

We would like to acknowledge the financial support from: 1) Innovation Fund Denmark in the CITIES project (1305-00027B) and in the OPTION project (63-2013-3), 2) the interreg project Smart Cities Accelerator (10606 SCA), and 3) EUDP 64013-0558 in the IEA Annex for energy efficient process control.

Appendix A. Parameter values for the optimal control examples

Table A.4 contains values of quantities related to the optimal tracking problem described in Section 7.1. It contains the control variables and the disturbance variables necessary for computing the initial and final steady states that are used to generate the setpoints in the objective function (73). This initial steady



(a) Energy consumption and H_2S vapor mole fraction.

(b) Temperature, pressure, liquid volume, and vapor fraction.

(c) Inputs and feed flow rate.

Figure 5: Optimal flash separation of a mixture of four hydrocarbons and hydrogen-sulfide with minimal energy consumption (in terms of cooling). Blue solid: Optimal strategy. Green dashed: Reference strategy. Red dash-dotted: Bounds. Black solid: Feed.

state is also used as the initial condition. Furthermore, it contains the feed variables as well as the reference strategy. Table A.5 contains values of quantities related to the minimal cooling problem described in Section 7.2. It contains the control variables and the disturbance variables used to compute the initial steady state that is used as initial condition in the minimal cooling problem. Furthermore, the table contains the feed variables and the reference strategy.

References

Arendsen, A.R.J., Versteeg, G.F., 2009. Dynamic thermodynamics with internal energy, volume, and amount of moles as states: Application to liquefied gas tank. Industrial & Engineering Chemistry Research 48, 3167–3176.

Barton, P.I., Lee, C.K., 2002. Modeling, simulation, sensitivity analysis, and optimization of hybrid systems. ACM Transactions on Modeling and Computer Simulation , 256–289.

Betts, J.T., 2001. Practical Methods for Optimal Control Using Nonlinear Programming. Advances in Design and Control, SIAM.

Biegler, L.T., 2007. An overview of simultaneous strategies for dynamic optimization. Chemical Engineering and Processing: Process Intensification 46, 1043–1053.

Biegler, L.T., 2010. Nonlinear programming: concepts, algorithms, and applications to chemical processes. SIAM.

Binder, T., Blank, L., Bock, H.G., Bulirsch, R., Dahmen, W., Diehl, M., Kronseder, T., Marquardt, W., Schlöder, J.P., von Stryk, O., 2001. Introduction to model based optimization of chemical processes on moving horizons, in: Online optimization of large scale systems. Springer, pp. 295–339.

Bisgaard, T., Huusom, J.K., Abildskov, J., 2015. Modeling and analysis of conventional and heat-integrated distillation columns. AIChE Journal 61, 4251–4263.

Bisgaard, T., Skogestad, S., Abildskov, J., Huusom, J.K., 2017. Optimal operation and stabilising control of the concentric heat-integrated distillation column (hidic). Computers & Chemical Engineering 96, 196–211.

Bock, H.G., Plitt, K.J., 1984. A multiple shooting algorithm for direct solution of optimal control problems, in: Proceedings 9th IFAC World Congress,

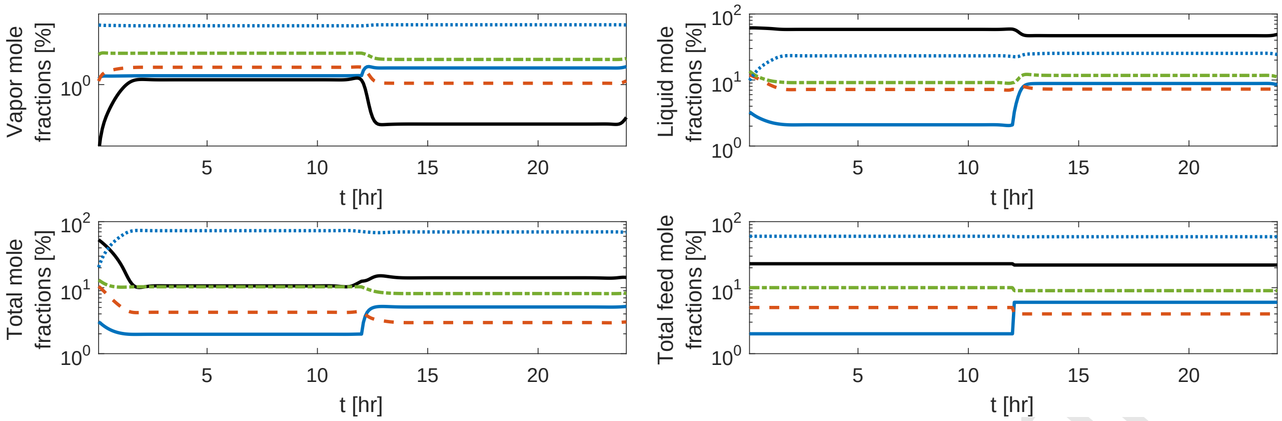


Figure 6: The mixture mole fractions and the total feed mole fractions for the minimal cooling strategy. Blue dotted: Methane. Green dash-dotted: Ethane. Red dashed: Propane. Black solid: Heptane. Blue solid: Hydrogen sulfide.

Table 1: Performance indicators for the solution of the two optimal control problems using the single-shooting algorithm with the simultaneous approach (Sim.) and the nested approach (Nest.). The implementations use either an exact or an inexact Newton method. Sim. time and Adj. time are the relative computation times of the simulation and the adjoint gradient computations. There is no relation between the relative computation times in the two examples. NLP Iter. and NLP Func. are the number of iterations and function evaluations performed by Matlab's fmincon, respectively. Fact. is the number of factorizations and Back. sub. is the number of back substitutions. Furthermore, the number of evaluations of thermodynamic properties for evaluating residual equations and for evaluating Jacobians are shown.

Optimal tracking problem

	Sim. time	Adj. time	NLP Iter.	NLP Func.	Fact.	Back. sub.	Thermo. for residual eq.	Thermo. for Jacobian
Sim. (inexact)	0.271	0.118	202	203	20,881	42,946	33,202	20,881
Sim. (exact)	0.264	0.118	202	203	32,632	32,632	0	42,376
Nest. (inexact)	1.000	0.145	202	203	138,358	179,585	105,328	118,853
Nest. (exact)	0.680	0.140	202	203	142,507	142,507	0	113,792

Minimal cooling problem

	Sim. time	Adj. time	NLP Iter.	NLP Func.	Fact.	Back. sub.	Thermo. for residual eq.	Thermo. for Jacobian
Sim. (inexact)	0.184	0.094	156	173	103,763	159,501	109,677	103,763
Sim. (exact)	0.300	0.116	147	213	166,462	166,462	0	227,806
Nest. (inexact)	1.000	0.129	150	230	738,140	889,473	475,960	605,349
Nest. (exact)	0.792	0.123	146	205	654,674	654,674	0	507,831

Pergamon Press. pp. 242–247.

Bryson, Jr., A.E., 1999. Dynamic Optimization. Addison-Wesley.

Capolei, A., Jørgensen, J., 2012. Solution of constrained optimal control problems using multiple shooting and ESDIRK methods, in: Proceedings of the 2012 American Control Conference, pp. 295–300.

Capolei, A., Völcker, C., Frydendall, J., Jørgensen, J.B., 2012. Oil reservoir production optimization using single shooting and ESDIRK methods, in: Automatic Control in Offshore Oil and Gas Production, International Federation of Automatic Control. pp. 286–291.

Castier, M., 2009. Solution of the isochoric-isoenergetic flash problem by direct entropy maximization. Fluid Phase Equilibria 276, 7–17.

Castier, M., 2010. Dynamic simulation of fluids in vessels via entropy maximization. Journal of Industrial and Engineering Chemistry 16, 122–129.

Colson, B., Marcotte, P., Savard, G., 2007. An overview of bilevel optimization. Annals of operations research 153, 235–256.

Diehl, M., Bock, H.G., Schlöder, J.P., Findeisen, R., Nagy, Z., Allgöwer, F., 2002. Real-time optimization and nonlinear model predictive control of processes governed by differential-algebraic equations. Journal of Process Control 12, 577–585.

Diehl, M., Ferreau, H.J., Haverbeke, N., 2009. Efficient numerical methods for nonlinear mpc and moving horizon estimation, in: Magni, L., Raimondo, D.M., Allgöwer, F. (Eds.), Nonlinear Model Predictive Control. Towards New Challenging Application. Springer. volume 384 of *Lecture Notes in Control and Information Sciences*.

Flatby, P., Skogestad, S., Lundström, P., 1994. Rigorous dynamic simulation of distillation columns based on UV-flash, in: IFAC Symposium on Advanced Control of Chemical Processes (ADCHEM '94), pp. 261–266.

Hammer, M., Morin, A., 2014. A method for simulating two-phase pipe flow with real equations of state. Computers & Fluids 100, 45–58.

Ingebritsen, S.E., Geiger, S., Hurwitz, S., Driesner, T., 2010. Numerical simulation of magmatic hydrotermal systems. Reviews of Geophysics 48, 1–33.

Jørgensen, J.B., 2007. Adjoint sensitivity results for predictive control, state- and parameter-estimation with nonlinear models, in: European Control Conference (ECC), 2007, IEEE. pp. 3649–3656.

Kourounis, D., Durlofsky, L.J., Jansen, J.D., Aziz, K., 2014. Adjoint formulation and constraint handling for gradient-based optimization of compositional reservoir flow. Computational Geosciences 18, 117–137.

Kristensen, M.R., Jørgensen, J.B., Thomsen, P.G., Jørgensen, S.B., 2004. An ESDIRK method with sensitivity analysis capabilities. Computers & Chemical Engineering 28, 2695–2707.

Kristensen, M.R., Jørgensen, J.B., Thomsen, P.G., Michelsen, M.L., Jørgensen, S.B., 2005. Sensitivity analysis in index-1 differential algebraic equations by ESDIRK methods, in: Proceedings of the 16th IFAC World Congress, Prague, Czech Republic. pp. 212–217.

Laiglecia, J.I., Lopez-Negrete, R., Diaz, M.S., Biegler, L.T., 2012. A simultaneous dynamic optimization approach for natural gas processing plants, in: Proceedings of Foundations of Computer Aided Process Operations (FOCAPO).

Li, Y., Johns, R.T., 2006. Rapid flash calculations for compositional simulation. SPE Reservoir Evaluation & Engineering SPE 95732, 521–529.

Table 2: Absolute computation time (in microseconds) of evaluating thermodynamical properties and solving a linear system $Ax = b$ for x where A is a dense matrix. The sizes of A are chosen according to the sizes of the matrices that occur in the Newton iterations in the simultaneous approach (Sim.) and in the inner and outer Newton iterations in the nested approach (Nest. inner and Nest. outer, respectively). We distinguish between evaluating thermodynamic properties to be used in residual equations and to be used in Jacobians. The linear systems are solved with an LU factorization (Fact.) and a back substitution (Back. sub.). The C factorizations and back substitutions are obtained with Netlib's LAPACK routines `dgetrf` and `dgetrs`.

Optimal tracking problem

	Thermo. eval.		Sim. ($A \in \mathbb{R}^{9\times9}$)		Nest. inner ($A \in \mathbb{R}^{5\times5}$)		Nest. outer ($A \in \mathbb{R}^{4\times4}$)	
	for residual eq.	for Jacobian	Fact.	Back. sub.	Fact.	Back. sub.	Fact.	Back. sub.
Matlab	34.588	39.829	3.514	2.271	2.844	2.117	2.634	1.887
C	1.435	1.428	0.575	0.188	0.205	0.112	0.145	0.092

Minimal cooling problem

	Thermo. eval.		Sim. ($A \in \mathbb{R}^{31\times31}$)		Nest. inner ($A \in \mathbb{R}^{16\times16}$)		Nest. outer ($A \in \mathbb{R}^{15\times15}$)	
	for residual eq.	for Jacobian	Fact.	Back. sub.	Fact.	Back. sub.	Fact.	Back. sub.
Matlab	211.319	464.254	5.121	2.516	3.169	2.213	3.032	2.151
C	4.305	6.431	1.165	0.290	0.359	0.150	0.288	0.129

Lima, E.R.A., Castier, M., Biscaia Jr., E.C., 2008. Differential-algebraic approach to dynamic simulations of flash drums with rigorous evaluation of physical properties. Oil & Gas Science and Technology-Revue de l'IFP 63, 677–686.

Lucia, A., Bonk, B.M., Waterman, R.A., Roy, A., 2012. A multi-scale framework for multi-phase equilibrium flash. Computers and Chemical Engineering 36, 79–98.

Luo, Z.Q., Pang, J.S., Ralph, D., 1996. Mathematical programs with equilibrium constraints. Cambridge University Press.

Luyben, W.L. (Ed.), 1992. Practical Distillation Control. Van Nostrand Reinhold, New York.

Michelsen, M., Mollerup, J., 2007. Thermodynamic Models: Fundamentals and Computational Aspects. Tie-Line Publications.

Michelsen, M.L., 1999. State function based flash specifications. Fluid Phase Equilibria 158, 617–626.

Outrata, J., Kocvara, M., Zowe, J., 2013. Nonsmooth approach to optimization problems with equilibrium constraints: theory, applications and numerical results. volume 28. Springer Science & Business Media.

Qiu, L., Wang, Y., Reitz, R.D., 2014. Multiphase dynamic flash simulations using entropy maximization and application to compressible flow with phase change. AIChE Journal 60, 3013–3024.

Raghunathan, A.U., Diaz, M.S., Biegler, L.T., 2004. An MPEC formulation for dynamic optimization of distillation operations. Computers & chemical engineering 28, 2037–2052.

Ritschel, T.K.S., Capolei, A., Jørgensen, J.B., 2017a. The adjoint method for gradient-based optimization of UV flash processes, in: Espuna, A., Graells, M., Puigjaner, L. (Eds.), 27th European Symposium on Computer Aided Process Engineering - ESCAPE 27, Barcelona, Spain.

Ritschel, T.K.S., Capolei, A., Jørgensen, J.B., 2017b. Dynamic optimization of UV flash processes, in: FOCAPO / CPC 2017, Tucson, Arizona.

Ritschel, T.K.S., Gaspar, J., Capolei, A., Jørgensen, J.B., 2016. An Open-source Thermodynamic Software Library. Technical Report DTU Compute Technical Report-2016-12. Department of Applied Mathematics and Computer Science, Technical University of Denmark.

Ritschel, T.K.S., Gaspar, J., Jørgensen, J.B., 2017c. A thermodynamic library for simulation and optimization of dynamic processes, in: Proceedings of the 20th World Congress of the International Federation of Automatic Control.

Saha, S., Carroll, J.J., 1997. The isoenergetic-isochoric flash. Fluid phase equilibria 138, 23–41.

Sahlodin, A.M., Watson, H.A.J., Barton, P.I., 2016. Nonsmooth model for dynamic simulation of phase changes. AIChE Journal 62, 3334–3351.

Schäfer, A., Kühl, P., Diehl, M., Schlöder, J., Bock, H.G., 2007. Fast reduced multiple shooting methods for nonlinear model predictive control. Chemical Engineering and Processing: Process Intensification 46, 1200–1214.

Smith, J.M., Van Ness, H.C., Abbott, M.M., 2005. Introduction to Chemical Engineering Thermodynamics. 7th ed., McGraw-Hill, New York, NY.

Stauffer, P.H., Viswanthan, H.S., Pawar, R.J., Guthrie, G.D., 2009. A system model for geologic sequestration of carbon dioxide. Environmental Science & Technology 43, 565–570.

Stichlmair, J.G., Fair, J.R., 1998. Distillation. Principles and Practice. Wiley-VCH, New York.

Thomson, G., 1996. The DIPPR® databases. International Journal of Thermophysics 17, 223–232.

Tolsma, J.E., Barton, P.I., 2000. DAEPACK an open modeling environment for legacy models. Industrial & Engineering Chemistry Research 39, 1826–1839.

Völcker, C., Jørgensen, J.B., Stenby, E.H., 2011. Oil reservoir production optimization using optimal control, in: IEEE Conference on Decision and Control 2011, IEEE Control Systems Society. pp. 7937–7943.

Völcker, C., Jørgensen, J.B., Thomsen, P.G., Stenby, E.H., 2010. Adaptive stepsize control in implicit Runge-Kutta methods for reservoir simulation, in: Proceedings of the 9th International Symposium on Dynamics and Control of Process Systems (DYCOPS 2010), pp. 509–514.

Wächter, A., Biegler, L.T., 2006. On the implementation of an interior-point filter line-search algorithm for large-scale nonlinear programming. Mathematical programming 106, 25–57.

Watson, H.A.J., Vikse, M., Gundersen, T., Barton, P.I., 2017. Reliable flash calculations: Part 1. nonsmooth inside-out algorithms. Industrial & Engineering Chemistry Research 56, 960–973.

Wilhelmsen, Ø., Skaugen, G., Hammer, M., Wahl, P.E., Morud, J.C., 2013. Time efficient solution of phase equilibria in dynamic and distributed systems with differential algebraic equation solvers. Industrial & Engineering Chemistry Research 52, 2130–2140.

Zavala, V.M., Biegler, L.T., 2009. Nonlinear programming strategies for state estimation and model predictive control, in: Magni, L., Raimondo, D.M., Allgöwer, F. (Eds.), Nonlinear Model Predictive Control. Towards New Challenging Applications. Springer. volume 384 of *Lecture Notes in Control and Information Sciences*, pp. 419–432.

Zaydullin, R., Voskov, D.V., James, S.C., Henley, H., Lucia, A., 2014. Fully compositional and thermal reservoir simulation. Computers & Chemical Engineering 63, 51–65.

Table 3: Absolute computation time (in seconds) and relative computation time of solving the two optimal control problems using the single-shooting algorithm with the simultaneous approach. Average over ten solutions to the optimal control problem.

Optimal tracking problem

		GCC compilers and Netlib's BLAS/LAPACK				Intel compilers and Intel MKL			
NLP solver	fmincon	fmincon	IPOPT	KNITRO	NPSOL	fmincon	IPOPT	KNITRO	NPSOL
Prog. Lang.	Matlab	C	C	C	C	C	C	C	C
Iterations	202	202	486	186	152	202	486	186	152
Func. Eval.	203	203	1517	190	153	203	1517	190	153
Absolute	8.503	0.851	1.319	0.291	0.223	0.843	0.911	0.245	0.102
Relative	1.000	0.100	0.155	0.034	0.026	0.099	0.107	0.029	0.012
Speedup	1.0	10.0	6.4	29.2	38.1	10.1	9.3	34.7	83.4

Minimal cooling problem

		GCC compilers and Netlib's BLAS/LAPACK				Intel compilers and Intel MKL			
NLP solver	fmincon	fmincon	IPOPT	KNITRO	NPSOL	fmincon	IPOPT	KNITRO	NPSOL
Prog. Lang.	Matlab	C	C	C	C	C	C	C	C
Iterations	156	156	328	117	150	156	328	117	150
Func. Eval.	173	173	1364	121	152	173	1364	121	152
Absolute	137.803	37.525	6.868	6.654	7.002	37.929	6.437	8.149	2.481
Relative	1.000	0.272	0.050	0.048	0.051	0.275	0.047	0.059	0.018
Speedup	1.0	3.7	20.1	20.7	19.7	3.6	21.4	16.9	55.5

Table A.4: Parameter values used in the optimal tracking problem in Section 7.1. Certain parameters have different values on the first and the second half of the time interval $[t_0,\ t_f] = [0\text{ hr},\ 4\text{ hr}]$.

	$t \in [0\text{ hr},\ 4\text{ hr}]$	Unit
	Initial Steady State	
T_F	505.0	K
P_F	1.0	MPa
z_F	[0.25; 0.40; 0.35]	
F_F	1.0	kmol/hr
Q	-1.0	MJ/hr
F_L	0.6	kmol/hr
F_V	0.4	kmol/hr
	Final Steady State	
T_F	505.0	K
P_F	1.0	MPa
z_F	[0.25; 0.40; 0.35]	
F_F	1.5	kmol/hr
Q	-40.0	MJ/hr
F_L	1.3	kmol/hr
F_V	0.2	kmol/hr

	$t \in [0\text{ hr},\ 2\text{ hr}]$	$t \in [2\text{ hr},\ 4\text{ hr}]$	Unit
	Feed		
T_F	505.0	505.0	K
P_F	1.0	1.0	MPa
z_F	[0.25; 0.40; 0.35]	[0.25; 0.40; 0.35]	
F_F	1.0	1.5	kmol/hr
	Reference Control Strategy		
Q	-1.0	-40.0	MJ/hr
F_L	0.6	1.3	kmol/hr
F_V	0.4	0.2	kmol/hr

Table A.5: Parameter values used in the minimal cooling problem in Section 7.2. Certain parameters have different values on the first and the second half of the time interval $[t_0,\ t_f] = [0\text{ hr},\ 24\text{ hr}]$.

	$t \in [0\text{ hr},\ 24\text{ hr}]$	Unit
	Initial Steady State	
T_F	335.15	K
P_F	1.0	MPa
z_F	[0.60; 0.10; 0.05; 0.23; 0.02]	
F_F	12.0	kmol/hr
Q	-150.0	MJ/hr
F_L	4.5	kmol/hr
F_V	7.5	kmol/hr

	$t \in [0\text{ hr},\ 12\text{ hr}]$	$t \in [12\text{ hr},\ 24\text{ hr}]$	Unit
	Feed		
T_F	335.15	335.15	K
P_F	1.0	1.0	MPa
z_F	[0.60; 0.10; 0.05; 0.23; 0.02]	[0.59; 0.09; 0.04; 0.22; 0.06]	
F_F	12.0	12.0	kmol/hr
	Reference Control Strategy		
Q	-90.0	-110.0	MJ/hr
F_L	4.5	5.5	kmol/hr
F_V	7.5	6.5	kmol/hr