Multi-objective optimal control of chemical processes using ACADO toolkit

F. Logist\textsuperscript{a}, M. Vallerio\textsuperscript{a}, B. Houska\textsuperscript{b}, M. Diehl\textsuperscript{b}, J. Van Impe\textsuperscript{a,}\textsuperscript{*}

\textsuperscript{a} BioTeC \& OPTEC, Chemical Engineering Department, Katholieke Universiteit Leuven, W. de Croylaan 46, 3001 Leuven, Belgium
\textsuperscript{b} SCD \& OPTEC, Electrical Engineering Department, Katholieke Universiteit Leuven, Kasteelpark Arenberg 10, 3001 Leuven, Belgium

\textbf{A R T I C L E   I N F O}

Article history:
Received 21 December 2010
Received in revised form 18 July 2011
Accepted 7 November 2011
Available online 15 November 2011

Keywords:
Multi-objective optimisation
Dynamic optimisation
Optimal control
Open source
Pareto set

\textbf{A B S T R A C T}

Many practical chemical engineering problems involve the determination of optimal trajectories given multiple and conflicting objectives. These conflicting objectives typically give rise to a set of Pareto optimal solutions. To enhance real-time decision making efficient approaches are required for determining the Pareto set in a fast and accurate way. Hereo, the current paper illustrates the use of the freely available toolkit ACADO Multi-Objective (www.acadotoolkit.org) on several chemical examples. The rationale behind ACADO Multi-Objective is the integration of direct optimal control methods with scalarisation-based multi-objective methods enabling the exploitation of fast deterministic gradient-based optimisation routines.

© 2011 Elsevier Ltd. All rights reserved.

1. Introduction

In practical chemical control problems, multiple and conflicting objectives are often present. This gives rise to a set of Pareto optimal solutions instead of a single solution (Miettinen, 1999). The most often exploited approaches to generate this Pareto set are (i) the Weighted Sum (WS) of the individual objectives or (ii) stochastic genetic algorithms (Deb, 2001). In the former case, a number of single-objective optimal control problems are solved for a grid of different weights using deterministic optimisation routines. In the latter case, a population of candidate solutions is updated based on repeated cost computations such that this population gradually evolves to the Pareto frontier. Unfortunately, both approaches exhibit certain restrictions. For the Weighted Sum it is known that (i) an equal distribution of weights does not necessarily lead to an even spread along the Pareto front, and that (ii) points in a non-convex part of the Pareto front cannot be obtained (Das \& Dennis, 1997). Stochastic approaches, although quite successful over the years (see, e.g., Bhaskar, Gupta, \& Ray, 2000; Mitra, Majumdar, \& Raha, 2004; Silva \& Biscaila, 2003 and the references therein), (i) may become time consuming due to the repeated model simulations required, (ii) are less suited to incorporate constraints exactly, and (iii) are limited to rather low dimensional search spaces. This last aspect restricts the control discretisations to coarse approximations.

To mitigate these drawbacks, several novel scalarisation-based multi-objective techniques, i.e., Normal Boundary Intersection (NBI) (Das \& Dennis, 1998), Normalised Normal Constraint (NCC) (Messac, Ismail-Yahaya, \& Mattson, 2003; Messac \& Mattson, 2004) and Enhanced Normalised Normal Constraint (ENNC) have been integrated with direct optimal control techniques (Abo-Ghander et al., 2010; Logist, Van Erdeghem, \& Van Impe, 2009). In addition, these techniques have been implemented in the freely available ACADO Multi-Objective toolkit (Logist, Houska, Diehl, \& Van Impe, 2010). The rationale is that this integration overcomes the disadvantages of the Weighted Sum, while still allowing the exploitation of fast deterministic solvers. Hence, the aim of this paper is to illustrate the usefulness of ACACO Multi-Objective as a tool to facilitate real-time decision making for dynamic chemical processes. To this end, several case-studies are presented, starting from a conceptual problem and adding gradually more complexity. Moreover, to the best of the authors’ knowledge, ACADO Multi-Objective is one of the first optimal control packages that provides systematic multi-objective optimisation features.

2. Problem formulation

In general, a multiple objective optimal control problem can be formulated as follows.

$$\min_{\chi(\xi), u(\xi), p, \xi_f} J_1, \ldots, J_m$$

subject to:

$$\frac{d\chi}{d\xi} = f(\chi(\xi), u(\xi), p, \xi), \quad \xi \in [0, \xi_f].$$

(1)

(2)
\[ \mathbf{0} = \mathbf{b}(x(0), x(\bar{t}), \mathbf{p}), \]  
\[ \mathbf{0} \geq \mathbf{c}_p(x(\xi), u(\xi), \mathbf{p}, \xi), \]  
\[ \mathbf{0} \geq \mathbf{c}_i(x(\xi), u(\xi), \mathbf{p}, \xi). \]

Here, \( x \) are the state variables, \( u \) are the control variables and \( \mathbf{p} \) denote the parameters to be optimised. The vector \( \mathbf{f} \) represents the dynamic system equations (on the interval \( \xi \in [0, \bar{t}] \)) with initial and terminal boundary conditions given by the vector \( \mathbf{b} \). The vectors \( \mathbf{c}_p \) and \( \mathbf{c}_i \) indicate respectively path and terminal inequality constraints on the states and controls. Each individual objective function can consist of both Mayer and Lagrange terms.

\[ J_i = h_i(x(\xi), \mathbf{p}, \xi) + \int_0^{\xi} g_i(x(\xi), u(\xi), \mathbf{p}, \xi) d\xi. \]

The admissible set \( S \) is defined to be the set of feasible points \( \mathbf{y} = (x(\cdot), u(\cdot), \mathbf{p}, \xi) \) that satisfy the dynamic equation as well as the boundary, path and terminal constraints in the above multi-objective optimal control problem.

In multi-objective optimisation, typically no single optimal solution exists, but a set of Pareto optimal solutions must be obtained. A point \( \mathbf{y}_p \in S \) is Pareto optimal if and only if there is no other point \( \mathbf{y}_q \in S \) with \( J_i(\mathbf{y}_p) \leq J_i(\mathbf{y}_q) \) for all \( i \in \{1, \ldots, m\} \) and \( J_j(\mathbf{y}_p) < J_j(\mathbf{y}_q) \) for at least one \( j \in \{1, \ldots, m\} \).

Broadly speaking, a solution is called Pareto optimal if there exists no other feasible solution that improves one objective function without worsening another.

### 3. ACADO Multi-Objective

ACADO Multi-Objective extends the ACADO toolkit for automatic control and dynamic optimisation (Houska, Ferreau, & Diehl, 2011) with several multi-objective approaches. Due to the self-contained object-oriented, C++ implementation, the toolkit (i) is easy-to-use, (ii) does not require third-party software, and (iii) allows a flexible control over algorithmic settings.

The idea behind ACADO Multi-Objective is the integration of efficient multi-objective scalarisation techniques with fast deterministic direct optimal control approaches (Logist, Houska, et al., 2010). Scalarisation methods convert the original multi-objective optimisation problem into a (series of) parametric single-objective optimisation problem whose solution each time yields one point of the Pareto set. By consistently varying the method’s parameter(s) (often referred to as weights) an approximation of the Pareto set is obtained. Despite its intrinsic drawbacks, combining the different objectives into a convex Weighted Sum (WS) is still one of the most popular scalarisation methods. NBI and NCC are alternative approaches that mitigate the WS drawbacks. Direct optimal control approaches transform the original infinite dimensional optimal control problem via discretisation into a finite dimensional Non-Linear Program (NLP). Sequential strategies (e.g., Single-Shooting (SSS)) discretise only the controls, leading to small but dense NLPs. In contrast, simultaneous approaches (e.g., Multiple-Shooting (MuS)) and Orthogonal Collocation) discretise both the controls and states, resulting in large but structured NLPs. The NLPs can be solved efficiently by deterministic optimisation routines, which exploit the sparsity.

A number of optimal control packages exist, e.g., (i) commercial software as \( \text{gPROMS} \) (Process System Enterprise Limited, 2010) and \( \text{PROPT} \) (Tomlab Optimization Inc, 2010) and (ii) non-commercial codes as \( \text{DynoPC} \) (Lang & Biegler, 2007), \( \text{MUSCOD-II} \) (Leineweber, Bauer, Bock, & Schlöder, 2003; Leineweber, Schäfer, Bock, & Schlöder, 2003), \( \text{DyOS} \) (Schlegel, Stockmann, Binder, & Marquardt, 2005) and \( \text{DOTcppSB} \) (Hirnajer, Balsa-Canto, & Banga, 2009). However, it should be noted that none of these packages offer systematic and advanced multi-objective features. Fig. 1 shows the structure of ACADO Multi-Objective. Its features are the following.

- **Multi-objective optimisation methods.** Four scalarisation methods have been implemented: Weighted Sum, Normal Boundary Intersection, Normalised Normal Constraint and Enhanced Normalised Normal Constraint. The implementation is generic such that, in principle, problems with any number of objectives can be tackled.
  - **Weighted Sum (WS).** The convex Weighted Sum of the individual objectives is still most often used in practice:
    \[ \min_{\mathbf{y} \in S} \sum_{i=1}^{m} w_i J_i(\mathbf{y}). \]
    with a scalarisation parameter or weight vector \( \mathbf{w} = [w_1, w_2, \ldots, w_m]^{\top} \in \mathbb{R}^m \) (with \( \sum_{i=1}^{m} w_i = 1 \)). However, it exhibits as drawbacks that (i) the returned solutions strongly depend on the scale and scaling of the objectives, (ii) a uniform variation of the weights does not necessarily result in an even spread on the Pareto set and (iii) points in non-convex regions of the Pareto set cannot be obtained (Das & Dennis, 1997).
  - **Normal Boundary Intersection (NBI).** NBI (Das & Dennis, 1998) has been developed based on geometrically intuitive arguments in order to overcome the deficiencies of the WS. The multi-objective optimisation problem is reformulated as follows:
    \[ \max_{\mathbf{y} \in S \subseteq \mathbb{R}} \mathbf{f}^{\top} \]  
    \[ \text{s.t. : } \mathbf{J}^{\top} \mathbf{f} - \mathbf{l} \mathbf{f} = \mathbf{J}(\mathbf{y}), \]
    with \( \mathbf{J} = [J_1^{\top}, J_2^{\top}, \ldots, J_m^{\top}]^{\top} \) the utopia point which contains the minima of the individual objective functions \( J_i(\mathbf{y}^*_i) \), and \( \mathbf{f} \) the pay-off matrix. In this matrix the i-th column contains the vector \( \mathbf{J}(\mathbf{y}^*_i) - \mathbf{J}^{\top} \). Similar to the WS, the vector \( \mathbf{w} \) represents the scalarisation parameters. The rationale behind NBI is to maximise...
the distance \( l \) from a point on the convex hull of individual minima along the (quasi-)normal towards the utopia point. Hence, this formulation gives rise to additional equality constraints (Eq. (9)). As a result an even spread along the Pareto set is obtained, and possible Pareto points in non-convex regions can be detected. However, in adverse cases also non-Pareto optimal solutions can be returned.

- **(Enhanced) Normalised Normal Constraint (E/NNC).** NNC (Messac et al., 2003) is based on similar geometric ideas as NBI but combines them with an \( \epsilon \)-constraint approach (Haines, Lasdon, & Wiser, 1971). Its mathematical formulation is as follows:

\[
\min_{y \in S} \quad J_m
\]

\[
s.t. : \quad (J(x_m) - J(y))' (\Phi w - J(y)) \geq 0 \quad i = 1, \ldots, m - 1, \quad (11)
\]

where \( - \) indicates variables based on normalised objectives. Hence, a selected (normalised) objective has to be minimised, while \( m - 1 \) half planes are added as inequality constraints (Eq. (11)). These half planes are chosen orthogonal to the utopia plane, which connects all (normalised) individual minima. Although NNC can avoid returning part of the non-Pareto optimal points generated by NBI, non-Pareto optimal solutions may still be provided. Hence, a Pareto filter algorithm to remove these candidate solutions may be required. Normalisation can be achieved by first shifting the objectives such that the utopia point coincides with the origin and then pre-multiplying them with a matrix \( T \in \mathbb{R}^{m \times m} \):

\[
J(y) = T (J(y) - J^*). \quad (12)
\]

Because Messac and Mattson (2004) considered only the shifting and scaling of the individual objectives in the classic Normalised Normal Constraint method (NNC), the matrix \( T \) is diagonal. The diagonal elements are the following:

\[
[T]_{i,i} = \frac{1}{J^*_i - J^*}, \quad (13)
\]

where \( J^*_i = \max_j [J^*_j], j = 1, \ldots, m \) is the maximum for objective function \( i \) for the set of individual minimisers \( y^*_j \). ENNC (Sanchis, Martínez, Blasco, & Salcedo, 2008) is an extension of NNC and uses adapted normalisation strategies. These authors introduce a different matrix \( T \) which can be generalised as follows:

\[
T = E \Phi^{-1}, \quad (14)
\]

with \( E \) an \( m \times m \) matrix containing zeros on the diagonal and ones on the off-diagonal. As a result cases with more than two objectives now also lead to a unique and uniform coverage of the Pareto set.

- **Weight generation.** The scalarisation parameters or weights \( w_i \) are generated automatically based on a given step size. Currently, only convex combinations can be generated, i.e., satisfying \( w_1 + w_2 + \cdots + w_m = 1 \) and \( w_i \geq 0 \). However, more advanced generation schemes that leave out the positivity constraints (Messac & Mattson, 2004), can be incorporated as well.

- **Single-objective optimisation problem initialisation.** Different initialisation strategies for the series of single objective optimisation problems can be selected. First, all single objective optimisation problems can be initialised using the same fixed values provided by the user. Second, a hot-start strategy that exploits the result of the previous single objective optimisation problem to initialise the next one, can introduce a significant decrease in computation time.

- **Direct optimal control methods.** In the current version of ACADO, only Single Shooting (Sargent & Sullivan, 1978) and Multiple Shooting (Bock & Plitt, 1984) are available. Orthogonal Collocation (Biegler, 1984) approaches are currently being implemented.

- **Integration and collocation methods.** Different integration routines are available for ordinary differential equation (ODE) systems, e.g., (i) explicit Runge-Kutta type integrators (RK12 (adaptive Euler), RK23 (second order), RK45 (Dormand–Prince), and (ii) implicit integrators such as Backward Differentiation Formulae, which can also be exploited to solve systems of differential and algebraic equations (DAEs). Settings involve, e.g., the absolute and relative integration tolerances. In the future several collocation schemes will be made available.

- **Automatic differentiation.** All integrators are equipped with automatic differentiation options. In particular, automatic differentiation is used to compute first and second order backward sensitivities of the objective, constraints, and differential equation with respect to control inputs, parameters, and initial values. Note that the optimal control problem formulation in ACADO is set up by using a special syntax which allows storing the functions in form of evaluation trees (Houska et al., 2011). This allows employing state of the art automatic differentiation algorithms (Griewank, 1989; Griewank & Walther, 2008).

- **Optimisation routines.** As optimisation routines Sequential Quadratic Programming (SQP) and Interior Point methods are available. The SQP employs QP subproblems (Ferreau, Bock, & Diehl, 2008) when solving the QPs for generating the direction, and uses a line search for obtaining the step size. Settings that can be specified, relate to, e.g., (i) the choice of the Hessian approximation (e.g., exact, constant, BFGS, block BFGS), (ii) the Karush–Kuhn–Tucker optimisation tolerance, (iii) the value for the Levenberg–Marquardt regularisation parameter and (iv) the minimum stepsize of the line-search globalisation parameter.

- **Pareto filter.** As NBI and (E)NNC may produce non-Pareto optimal points, the candidate solution set and corresponding weight values can be filtered using a Pareto filter algorithm. Two filters are available. The first filter simply compares a generated point with every other generated point, and if a point is not Pareto optimal, it is eliminated (Messac & Mattson, 2004). The second filter is able to partially remove non-Pareto optimal points without the need for generating a set first (Logist & Van Impe, in press).

- **Visualisation.** The resulting Pareto sets can be directly plotted for cases with up to three objectives. The optimal states and controls and their corresponding Pareto optimal cost values can be exported in different formats, which allows an easy post-processing in different environments.

- **Default settings.** For all algorithmic settings, default values have been specified which frees the user from explicitly selecting values for all options.

4. Case studies

Three dynamic chemical processes are used as case studies to illustrate ACADO Multi-Objective’s features. As the focus is on a clear illustration, the cases start with a conceptual example and gradually add complexity. Nevertheless, it must be mentioned that the three cases cover a wide range of typical control problems, e.g., singular vs. non-singular optimal solutions, fixed vs. free end times, control and state path constraints, and terminal constraints.

The current approach aims at approximating and visualising the Pareto set, such that the decision maker can select one solution afterwards according to his/her preferences. Hence, it is assumed that there is only an a posteriori articulation of preferences by the decision maker, because the results are first generated and the
selection is performed afterwards (Marler & Arora, 2004). It is noted that whenever preference information from the decision maker is available (e.g., prices for the different costs) before the computations have started, an a priori multi-objective approach can and must be followed. For instance, a single optimisation problem can be formulated using the prices as weights in Weighted Sum approach. This Weighted Sum then represents an economic profit function. However, this approach is not always possible or desirable. First, price information is not always available or can be highly uncertain. Second, the different objectives can be incommensurable and, hence, hard to combine. Or third, the decision maker does not want to specify a priori preferences and is interested in evaluating different alternative optimal solutions. In these cases, an a posteriori multi-objective approach is useful. In addition, interactive multi-objective optimisation methods exist. With these methods, the decision maker gradually provides more information about his/her preferences during the solution process (see, e.g., Hakanen, Miettinen, Mäkelä, & Manninen, 2005).

4.1. Case I: catalyst mixing problem in a tubular reactor

This problem considers a steady-state plug flow reactor of fixed length $z$. The reactor is packed with catalyst pellets. These catalysts are required to stimulate a series of reactions (one reversible and one irreversible $S_1 \leftrightarrow S_2 \rightarrow S_3$). These assumptions give rise to the following model:

\[
\frac{dx_1}{dz} = -u(x_1 - 10x_2),
\]

\[
\frac{dx_2}{dz} = u(x_1 - 10x_2) - (1 - u)x_2,
\]

with $x_1$ and $x_2$ the concentrations of $S_1$ and $S_2$, $u$ the fraction of catalyst A and $z$ the spatial coordinate. The original problem (Vassiliadis, Balsa-Canto, & Banga, 1999) considered the optimal mixing policy of the two catalysts in order to maximise the production of species $S_3$ at the reactor outlet:

\[
J_1 = -(1 - x_1(z_f) - x_2(z_f)),
\]

where the reactor has a length $z_f$ equal to 1. To this end the optimal catalyst mixing profiles $u(z)$ and $(1 - u(z))$ along the reactor must be determined. The fraction of catalyst A is bounded:

\[
0 \leq u(z) \leq 1,
\]

which directly also bounds the fraction of catalyst B:

\[
0 \leq 1 - u(z) \leq 1.
\]

At the inlet only $S_1$ is fed:

\[
\mathbf{x}(0) = [1, 0, 0, 0, 0, 0, 65, 2]^T.
\]

To introduce a multi-objective nature, the minimisation of the amount of the most expensive catalyst, i.e., catalyst A, is added as an objective:

\[
J_2 = \int_0^{z_f} u(z)dz.
\]

4.2. Case II: temperature control of a batch reactor

The second case considers the Williams–Otto fed-batch reactor (see, e.g., Hannemann & Marquardt, 2010). In this reactor the following reactions take place: $A + B \rightarrow C$, $C + B \rightarrow P + E$, and $P + C \rightarrow G$. The reactant $A$ is initially present, whereas the reactant $B$ is continuously added. During the exothermic reactions, the products $P$ and $E$ as well as the side product $G$ are formed. The heat generated by the exothermic reactions is removed by a cooling jacket. This leads to the following dynamic model:

\[
\frac{dx_A}{dt} = \frac{-x_A u_1}{1000} - k_1 \eta_1 x_A x_B,
\]

\[
\frac{dx_B}{dt} = \frac{(1 - x_B) u_1}{1000} + k_1 \eta_1 x_A x_B - k_2 \eta_2 x_B x_C,
\]

\[
\frac{dx_C}{dt} = \frac{-x_C u_1}{1000} + k_2 \eta_2 x_B x_C - k_3 \eta_3 x_C x_P - k_4 \eta_3 x_C x_P,
\]

\[
\frac{dx_P}{dt} = \frac{x_P u_1}{1000} + k_2 \eta_2 x_B x_C - k_4 \eta_3 x_C x_P,
\]

\[
\frac{dx_E}{dt} = \frac{-x_E u_1}{1000} + k_3 \eta_3 x_C x_P,
\]

\[
\frac{dx_G}{dt} = \frac{-x_G u_1}{1000} + k_5 \eta_3 x_C x_P,
\]

\[
\frac{dP}{dt} = \frac{(T_f - T) u_1}{1000} + k_8 \eta_1 x_A x_B + k_9 \eta_2 x_B x_C + k_{10} \eta_3 x_C x_P - l_1 (T - 1000 u_2),
\]

\[
\frac{dV}{dt} = \frac{u_1}{1000},
\]

with as state variables the dimensionless concentrations $x_i$, the reactor temperature $T$ and the liquid volume $V$. The Arrhenius dependencies are $\eta_1 = e^{-66666.7/(T + 723.15)}$, $\eta_2 = e^{-5333.3/(T + 723.15)}$ and $\eta_3 = e^{-11111.0/(T + 723.15)}$. For the concrete values of the constants, see Hannemann and Marquardt (2010). The manipulated variables are $u_1$, i.e., the feeding rate of $B$ and $u_2$, i.e., the scaled jacket fluid temperature. For given initial conditions:

\[
\mathbf{x}(0) = [1, 0, 0, 0, 0, 0, 65, 2]^T,
\]

the aim is to maximise the economic yields of the products $P$ and $E$ at end of the operation $t_f$:

\[
J_1 = -x_P (t_f) V(t_f),
\]

\[
J_2 = -x_E (t_f) V(t_f),
\]

while meeting the following safety and constructive limits:

\[
60 \leq T(t) \leq 90,
\]

\[
0 \leq u_1(t) \leq 5.784,
\]

\[
0.02 \leq u_2(t) \leq 0.1,
\]

and without overfilling the tank:

\[
V(t_f) \leq 5.
\]

Note that minus signs are present in objectives $J_1$ and $J_2$ in order to cast everything in a minimisation frame.

4.3. Case III: temperature control of a jacketed tubular reactor

Case III concerns a jacketed tubular reactor in which an exothermic, irreversible first order reaction ($A \rightarrow B$) takes place. The reaction is assumed to operate under steady-state conditions:

\[
\frac{dx_1}{dz} = \frac{\alpha}{\beta} (1 - x_1) e^{j^2/(1 + x_2)},
\]

\[
\frac{dx_2}{dz} = \frac{\alpha}{\beta} (1 - x_1) e^{j^2/(1 + x_2)} + \beta (u - x_2),
\]

with $x_1$, $x_2$ and $u$ the dimensionless reactant concentration, reactor temperature and jacket temperature. (For parameters values, see Logist, Smets, & Van Impe, 2008 or Logist, Van Erdeghem, Smets, &
Van Impe, 2009). The original aim was to derive an optimal jacket fluid temperature that maximises on the one hand conversion:

\[ J_1 = C_0(1 - x_1(z_f)) \]  

and on the other hand the heat recovery via the jacket:

\[ J_2 = \frac{\beta}{\sigma} \int_0^{z_f} (u(z) - x_2)dz \]  

For practical reasons bounds are imposed on all variables:

\[ x_{1, \min} \leq x_1(z) \leq x_{1, \max} \]  

\[ x_{2, \min} \leq x_2(z) \leq x_{2, \max} \]  

\[ u_{\min} \leq u(z) \leq u_{\max} \]  

The initial conditions are given as:

\[ x(0) = [0, 0]^T \]  

To make this case more challenging the reactor length \( z_f \) is allowed to vary in between 0.5 and 1, and this variable is added as a third objective to be minimised:

\[ J_3 = z_f \]  

However, a minimum conversion has to be ensured:

\[ x_1(z_f) \geq 0.90 \]  

Note again that all objective functions have been specified such that they fit in a minimisation framework.

5. Results

In the current study, both Single and Multiple Shooting approaches with an SQP optimiser have been combined with the four multi-objective methods: WS, NBI NNC and ENNC. Table 1 provides an overview of the algorithmic settings and the computational results for the different cases. Each time a uniform distribution of the scalarisation parameters is used.

### 5.1. Case I: catalyst mixing problem in a tubular reactor

![Fig. 2. Catalyst mixing: Pareto set.](image)

Fig. 2 displays the obtained Pareto set. The trade-off between the use of catalyst A and the conversion to species S2 is clearly visible. Note that \( J_1 \) has been depicted, which represents minus the conversion and which has to be minimised. When looking at the different multi-objective methods, it is clear that the WS does not yield an equal distribution along the Pareto set as points tend to...
cluster near the optimal conversion extreme. The other methods all three yield the same results, but the spread is now more uniform.

Fig. 3 depicts a selection of the optimised state and control variables along the Pareto front. When the complete focus is put on limiting the use of catalyst A, it is seen that the control consists of one minimum arc, meaning that only catalyst B will be used. As a result, no reaction occurs because species $S_1$, which is the only species initially present, is not converted and no $S_2$ or $S_3$ is formed. However, when the production of species $S_3$ does play a role, the catalyst profiles exhibit a maximum–singular–minimum type arc structure. The more the focus shifts towards the production of $S_3$, the larger the maximum and singular arc become and the higher the values are during the singular interval.

It has to be noted that in literature results have been reported in which analytical expressions for the different optimal arcs are used (e.g., Gunn, 1967). Hence, this leads to a more accurate control profile. However, the analytical derivations required to obtain these relations become rapidly tedious when the number of states increases.

5.2. Case II: temperature control of a batch reactor

The second case extends the number of controls and states involved to two and eight, respectively. The approximation of the Pareto set, which is depicted in Fig. 4, is highly perturbed when using the WS. However, a much more accurate approximation is found with the other three approaches. Also, the trade-off is clearly visible, i.e., focussing on the production of $P$ decreases the production of $E$ and vice versa. It has to be noted that $J_1$ and $J_2$ have been employed, which are the negative values of the production and, hence, these have to be minimised.
5.3. Case III: temperature control of a jacketed tubular reactor

The third case tests the proposed multi-objective optimal control procedures for more than two, i.e., three objectives. The resulting 3D Pareto front is depicted in Fig. 6 and is clearly continuous and convex. As a highly uneven distribution is obtained with the WS, the WS results have been omitted. When comparing the results for NBI, NNC and ENNC, it is observed that in this case only ENNC and NBI yield identical results. The results obtained with NNC slightly differ. These differences are related to the different normalisation schemes used in NNC and ENNC and only occur for cases with more than two objectives (Logist & Van Impe, in press). Nevertheless, an accurate approximation with a nice spread is returned by all three methods.

Fig. 7 depicts a selection of the optimal controls and corresponding states. When looking at the optimal controls, the arc structure is typically of the maximum-minimum-constrained-minimum type. Hence also here, the solutions are solely determined by active constraints. This arc sequence leads to a so-called trapezoidal reactor temperature profile. In the first reactor part the reactor temperature has to be as high as possible (without violating the upper bound) in order to stimulate conversion and heat production. In the last part, the reactor temperature has to be decreased in order to recover heat. When solely conversion or heat recovery is aimed at, almost the entire reactor is used (i.e., $z_f = 1$ or $z_f = 0.87$, respectively). However, in the former case the control is used to maintain the upper reactor temperature until the outlet (i.e., no second minimum arc), whereas in the latter case the upper temperature is only
maintained at the upper bound in a small part, in favour of a large heat recovery section at the end (i.e., a large second minimum arc). Alternatively, the reactor length is reduced to its minimum value, when this objective is solely concentrated on. As expected, the intermediate Pareto optimal points exhibit also intermediate behaviour, i.e., reactor lengths shorter than 1 but still with trapezoidal reactor temperature profiles.

5.4. Discussion

Despite the fine control discretisations (i.e., going from 25 over 40 to 50 piecewise constant parts), the toolkit is able to solve the multi-objective optimal control problems accurately and sufficiently fast for both the bi- and tri-objective cases. For integration and optimisation tolerances of at most 1E–5 and without specific tuning of several options, CPU times on a 2.40 GHz machine with 4 GB RAM are between 0.1 and 1.4 s per Pareto point. Hence, the time to compute the entire Pareto set is below 1 minute. This is not only the case for examples II and III, which are completely determined by active constraints, but also for example I. For this last case, still consistent and meaningful results are obtained despite the insensitivity that singular optimal control problems often exhibit. Also both fixed and free end time problems can easily be dealt with. When comparing the Single and Multiple Shooting results a slight advantage in favour of Single Shooting is observed as the investigated dynamic systems do not exhibit instabilities and the results are not determined by too restrictive active state constraints. The algorithmic speed-up due to hot-starting is clear and concerns factors between 1.5 and 4.0 for the cases investigated.

To analyse the influence of the control discretisation, additional optimisation experiments with finer control grids have been performed (see also Table 1). In general the improvement of the cost values was rather low with respect to the increase in computation times. For instance, when looking at the individual objectives, no more than 0.5% improvement has been observed. Hence, there exists a trade-off between numerical accuracy and computational burden. However, the actual choice for a certain trade-off depends on the preferences of the user. In the current case, 20, 40 and 50 piecewise constant parts have been found to be acceptable.

To speed up computations, additional approaches can be implemented such as adaptive grid refinement methods (Schlegel et al., 2005) or structure detection techniques (Schlegel & Marquardt, 2006a,b). Alternatively, a user can exploit a coarse grid to get a fast and rough approximation of the different alternatives along the Pareto set. Based on this approximation he/she can select one or more points according this his/her preferences. For these selected points, these profiles can be re-optimised using a finer grid in order to increase the accuracy. This idea is similar to the one reported in Logist, Sager, Kirches, and Van Impe (2010) for the computation of Pareto sets for multi-objective dynamic optimisation problems with integer controls. Here, an approximation of the Pareto set is quickly computed based on a convex relaxation of the integer control. Afterwards, an integer feasible solution can be computed up to any desired accuracy for a selected point via grid refinement and sum up rounding (Sager, 2006).

6. Conclusions

In summary, the freely available toolkit ACADO Multi-Objective, which can be downloaded from www.acadotoolkit.org, has been shown to be able to solve different types of multi-objective optimal control problems appearing in chemical engineering (e.g., with/without singular arcs, with fixed/free end point, with control/state constraints, with terminal constraints) for different numbers of objectives. The use of this toolkit has been illustrated for several chemical processes of increasing complexity. To the best of the authors’ knowledge, ACADO is one of the first optimal control software tools that systematically provides multi-objective features. Future developments may include the inclusion of additional weight generation schemes and additional multi-objective approaches.

Acknowledgements

Work supported in part by Katholieke Universiteit Leuven: OT/10/035, OPTEC (Center-of-Excellence Optimization in Engineering) (FPV/10/002), SCORES4CHEM (KP/09/005); by the Belgian Federal Science Policy Office: Belgian Program on Interuniversity Poles of Attraction; by EU FP7-HD-MPC (ICT-223854), FP7-EMBOCON (ICT-248940) and industry project ACCM. M. Valle- rio has a Ph.D. grant of the Institute for the Promotion of Innovation through Science and Technology in Flanders (IWV-Vlaanderen). J.F. Van Impe holds the chair Safety Engineering sponsored by the Belgian chemistry and life sciences federation essencia.

References


Process System Enterprise Limited. (2010). gPROMS.


Tomlab Optimization Inc. (2010). *PROPT – Matlab optimal control software*.