A numerical method for the approximation of reachable sets of linear control systems is discussed. The method is based on the formulation of suitable optimal control problems with varying objective function, whose discretization by Runge–Kutta methods leads to finite-dimensional convex optimization problems. It turns out that the order of approximation for the reachable set depends on the particular choice of the Runge–Kutta method in combination with the selection strategy used for control approximation. For an inappropriate combination, the expected order of convergence cannot be achieved in general. The method is illustrated by two test examples using different Runge–Kutta methods and selection strategies, in which the run times are analysed, the order of convergence is estimated numerically and compared with theoretical results in similar areas.

**Keywords:** Approximation of reachable sets; Discretization of optimal control problems; Direct solution methods; Set-valued Runge–Kutta methods; Order of convergence

**2000 Mathematics Subject Classification:** 93B03; 49N05; 65L06; 49J53

1. Introduction

The subject of this article is the description of an algorithm for the approximation of reachable sets of linear control problems. The problem of determining convex reachable sets can be equivalently described by infinitely many optimal control problems (OCPs), where the objective function is adapted. By choosing only finitely many directions, approximations of reachable sets can be obtained. The occurring OCPs are not solved theoretically by use of the Pontryagin’s maximum principle as in ref. [1], but numerically by suitable discretization methods. This allows to treat also time-dependent linear problems and even special nonlinear ones with convex reachable sets. Nonpolyhedral control regions can be treated as nonlinear inequalities and equalities. Results concerning the convergence of discretized OCPs can be found in refs. [2–4] and the references stated therein.

*Corresponding author. Email: gerdts@math.uni-hamburg.de*
In this context, the particular choice of the selection strategy used for control approximation turns out to be crucial for the order of convergence and depends on the choice of the Runge–Kutta scheme used for the discretization of the underlying differential equations. To illustrate this dependency, several Runge–Kutta methods with different selection strategies (piecewise constant, piecewise linear and independent selection) are discussed in more detail for two illustrative examples.

This approach avoids cumbersome set operations (like Minkowski sums, unions of sets, etc.) and leads to known optimization methods, which in addition yield not only the endpoints of optimal trajectories, but also the entire trajectory including the corresponding optimal control. Furthermore, this approach is useful for linear control problems with control regions formulated with nonlinear restrictions [12] and for nonlinear control problems yielding convex reachable sets also. However, the close connection between set-valued analysis and optimal control is shown in section 3. A comparison with set-valued methods as in refs. [5–11] is beyond the scope of this article.

Methods for linear differential inclusions based on set-valued quadrature methods or set-valued Runge–Kutta methods are mentioned in ref. [7] as well as other methods, e.g. estimation methods for reachable sets [12] and ellipsoidal methods (cf. [11] for an overview). Newer developments of these methods achieve inner approximations [13,14] and outer approximations [15] of the reachable set [7].

The problem of the approximation of reachable sets appears in several disciplines: control theory, ordinary differential equations with uncertainties or with discontinuities in the state, necessary conditions for a minimum in nonsmooth analysis, differential games and viability theory, cf. [16–20]. The convexity of these reachable sets can be guaranteed for linear differential inclusions, but may also appear for special nonlinear problems (see e.g. [9, Example 5.2.3]).

The article is organized as follows. In section 2, basic notations and properties of reachable sets are summarized. Basic facts on the description of convex sets and arithmetic set operations are introduced and form the basis for the results of section 3. In particular, the Hausdorff and the Demyanov distances are defined, which are used to measure the speed of convergence with respect to the optimal value and the optimal trajectory, respectively. In section 3, the problem of calculating the boundary of the reachable set is reformulated, as infinitely many OCPs differ only in the objective function. These OCPs are discretized by the use of explicit Runge–Kutta methods and suitable control approximations resulting in finite-dimensional (linear/nonlinear) optimization problems. Herein, several approximation classes for the control lead to different selection strategies in the discretization. The section ends with a formulation of the proposed method for the approximation of reachable sets and its implementation. Several combinations of Runge–Kutta methods and selection strategies are discussed in section 4 with some illustrative test examples. Tables with convergence results, run times and visualizations of reachable sets are included. The numerically observed order of convergence for these combinations is compared with known result for the order in the setting of classical ODEs, OCPs and in the set-valued case. Finally, an outline for further research concludes the article.

2. Notation

In this section, some introductory definitions and results are collected.

The basic underlying problem is the following control problem

\textbf{Problem 2.1} Let $I := [t_0, t_f]$ be a real interval, $x_0 \in \mathbb{R}^n$, $A : I \to \mathbb{R}^{n \times n}$ and $B : I \to \mathbb{R}^{n \times m}$ be two $L_1$-integrable matrix functions and $U \subset \mathbb{R}^m$ be a nonempty, convex compact set.
For a given control function \( u : I \to \mathbb{R}^m \) with \( u(\cdot) \in L^\infty(I, \mathbb{R}^m) \), we are looking for an absolutely continuous solution \( x(\cdot) \in W^{1,1}(I, \mathbb{R}^n) \) of the differential equation

\[
\dot{x}(t) = A(t)x(t) + B(t)u(t) \quad (a.e. \ t \in I),
\]

(1a)

\[
x(t_0) = x_0,
\]

(1b)

\[
u(t) \in U \quad (a.e. \ t \in I).
\]

(1c)

**Definition 2.2** Let us study Problem 2.1 and let \( t \in I \). Then,

\[
\mathcal{R}(t, t_0, x_0) := \{ y \in \mathbb{R}^n \mid \exists u(\cdot) \text{ control function and } \exists x(\cdot) \text{ corresponding solution of Problem 2.1 with } x(t) = y \}
\]

is called the reachable set of the corresponding control problem for the time \( t \).

In 1965, Aumann [21] discovered the convexity of the set-valued integral, which easily leads to the convexity of the reachable set for linear control problems.

**Proposition 2.3** In Problem 2.1, the reachable set \( \mathcal{R}(t, t_0, x_0) \) is convex, compact and nonempty for every \( t \in I \).

**Proof** See e.g. [22, Theorem 1]. ■

Some notations from convex analysis are recalled, which are necessary for the explanation of the algorithm described later.

**Definition 2.4** Denote the set of all nonempty convex compact sets by \( \mathcal{C}(\mathbb{R}^n) \) in \( \mathbb{R}^n \) and let \( C \in \mathcal{C}(\mathbb{R}^n), l \in \mathbb{R}^n \).

Then,

\[
\delta^*(l, C) := \max_{c \in C} l^T c \quad \text{and} \quad Y(l, C) := \{ c \in C \mid l^T c = \delta^*(l, C) \}
\]

defines the support function of \( C \), respectively, the set of supporting points of \( C \) in direction \( l \).

Support functions, respectively, supporting points describe fully a convex compact set.

**Proposition 2.5** Let \( C \in \mathcal{C}(\mathbb{R}^n) \). Then,

\[
C = \bigcap_{\|l\|_2 = 1} \{ x \in \mathbb{R}^n \mid l^T x \leq \delta^*(l, C) \}, \quad \partial C = \bigcup_{\|l\|_2 = 1} Y(l, C),
\]

\[
C = \operatorname{co} \left( \bigcup_{\|l\|_2 = 1} \{ y(l, C) \} \right) \text{ with arbitrary } y(l, C) \in Y(l, C),
\]

where \( \partial C \) denotes the boundary of \( C \) and \( \operatorname{co}(\cdot) \) the convex hull of a set.

**Proof** See e.g. [23, section V, Theorem 2.2.2] and [23, section V, Proposition 3.1.5].

The last equation follows easily if one estimates the support function of the right-hand side in direction \( l \) by \( l^T y(l, C) = \delta^*(l, C) \) from below. ■

A common arithmetic operation on sets is the scalar multiplication and the Minkowski sum, which are recalled here.
**Definition 2.6** Let $C, D \in \mathcal{C}(\mathbb{R}^n)$, $\lambda \in \mathbb{R}$ and $A \in \mathbb{R}^{m \times n}$. Then,

\[
\lambda C := \{ \lambda c \mid c \in C \}, \quad AC := \{ Ac \mid c \in C \},
\]

\[
C + D := \{ c + d \mid c \in C, d \in D \}
\]

defines the scalar multiplication, the image of $C$ under the linear map $x \mapsto Ax$ and the Minkowski sum.

We need the following theoretical result which states convexity and compactness of the set operations defined earlier.

**Lemma 2.7** Let $C, D \in \mathcal{C}(\mathbb{R}^n)$, $\lambda \in \mathbb{R}$ and $A \in \mathbb{R}^{m \times n}$. Then, $\lambda C$ and $C + D$ are elements of $\mathcal{C}(\mathbb{R}^n)$, and $AC$ is an element of $\mathcal{C}(\mathbb{R}^m)$. Furthermore,

\[
\delta^* (l, \lambda C) = \lambda \delta^* (l, C), \quad Y(l, \lambda C) = \lambda Y(l, C) \quad \text{(if $\lambda \geq 0$)},
\]

\[
\delta^* (\eta, AC) = \delta^* (A^\top \eta, C), \quad Y(\eta, AC) = AY(A^\top \eta, C)
\]

\[
\delta^* (l, C + D) = \delta^* (l, C) + \delta^* (l, D), \quad Y(l, C + D) = Y(l, C) + Y(l, D)
\]

for all $l \in \mathbb{R}^n$, $\eta \in \mathbb{R}^m$.

**Proof** To guarantee that the operations give results in $\mathcal{C}(\mathbb{R}^n)$ and the equations on the support functions, see [23, section V, Theorem 3.3.3(i) and Proposition 3.3.4]. The equations on the supporting set follow immediately from calculus rules on the subdifferential in [23, section VI, Theorem 4.1.1 and equation (3.1)] and [24, Theorem 23.9], as [23, section VI, Proposition 2.1.5 and equation (3.1)] connects the subdifferential of the support function and the supporting set.

**Definition 2.8** Let $C, D \in \mathcal{C}(\mathbb{R}^n)$. Then,

\[
d(C, D) := \max_{c \in C} \text{dist}(c, D), \quad \text{where} \quad \text{dist}(c, D) := \min_{d \in D} \| c - d \|_2,
\]

\[
d_H(C, D) := \max \{ d(C, D), d(D, C) \},
\]

\[
d_D(C, D) := \sup_{l \in T_C \cap T_D} \| y(l, C) - y(l, D) \|_2
\]
deﬁnes the one-sided Hausdorff distance, respectively, the Hausdorff distance and the Demyanov distance between two sets. Hereby, $T_C$ is deﬁned as a set of all normed directions in $\mathbb{R}^n$ for which the supporting set $Y(l, C)$ consists of only one point $y(l, C)$ ($T_D$ is deﬁned analogously for the set $D$). $T_C$ and $T_D$ are subsets of the unit sphere $S_{n-1} \subset \mathbb{R}^n$ of full measure.

Well-known properties of the support function make it easy to prove the following result for the Hausdorff distance:

**Proposition 2.9** Let $C, D \in \mathcal{C}(\mathbb{R}^n)$. Then,

\[
d_H(C, D) = \max_{\| l \|_2 = 1} | \delta^* (l, C) - \delta^* (l, D) | \leq d_D(C, D).
\]

**Proof** See e.g. [23, section V, Theorem 3.3.8] and [25, Lemma 4.1].
3. Direct solution method for the approximation of reachable sets

3.1 Connection to OCPs

As we know from Proposition 2.3 that the reachable set for Problem 2.1 is convex, it is sufficient to calculate merely the boundary of the reachable set.

Proposition 2.5 gives a motivation to calculate at least one support point (which lies automatically at the boundary) of the reachable set in direction \( l \in \mathbb{R}^n \) with \( \|l\|_2 = 1 \). Note that even in the case that the reachable set is not strictly convex and the set of supporting points is a \((n-1)\)-dimensional face, for a fixed direction \( l \), one supporting point in this direction is sufficient to reconstruct the reachable set.

Thus to calculate a supporting point \( x(t_f) \) on the boundary of the reachable set \( \mathcal{R}(t_f, t_0, x_0) \) in a fixed direction \( l \), we have to find an admissible control function \( u(\cdot) \in U \) that maximizes the functional \( y \mapsto l^\top y \) (resulting in the support function \( \delta^*(l, \mathcal{R}(t_f, t_0, x_0)) \) as optimal value). This constitutes the following special OCP of Mayer type:

\[
\begin{align*}
\text{Maximize} & \quad l^\top x(t_f) \\
\text{subject to} & \quad u(\cdot) \in L^\infty([t_0, t_f], \mathbb{R}^m), x(\cdot) \in W^{1,1}([t_0, t_f], \mathbb{R}^n) \\
& \quad x(\cdot) \text{ corresponding solution to } u(\cdot) \text{ for } (1a)–(1c).
\end{align*}
\]

We denote the optimal solution of (OCP\(_l\)) by \( x^*(\cdot; l) \) and \( u^*(\cdot; l) \), where the argument \( l \) indicates the dependency of the direction \( l \).

As already mentioned in Proposition 2.5, the convexity and compactness of the reachable set guaranteed by Proposition 2.3 lead to the equivalent representation by considering supporting points in all directions \( l \in S_{n-1} \):

\[
\mathcal{R}(t_f, t_0, x_0) = \text{co}\left( \bigcup_{l \in S_{n-1}} \{x^*(t_f; l)\} \right), \quad (2)
\]

\[
= \bigcap_{l \in S_{n-1}} \{x \in \mathbb{R}^n | l^\top x \leq l^\top x^*(t_f; l)\}. \quad (3)
\]

In section 3.3, comparable equalities to (2) and (3) are shown for the discrete reachable sets.

3.2 Approximation of reachable sets by discretized OCPs

For the moment, let \( l \in S_{n-1} \) be fixed in Problem (OCP\(_l\)).

We introduce a grid with grid points for \( N \in \mathbb{N}, N \geq 1 \) and step-size \( h \) by

\[
t_i = t_0 + ih \in [t_0, t_f], \quad (i = 0, 1, \ldots, N) \quad \text{and} \quad h = \frac{t_f - t_0}{N}. \quad (4)
\]

Each explicit Runge–Kutta scheme with \( s \) stages can be characterized by its Butcher array:

\[
\begin{array}{c|cccccccc}
\gamma_1 & 0 & \cdots & \cdots & 0 \\
\gamma_2 & \alpha_{21} & 0 & \cdots & 0 \\
\vdots & \vdots & \ddots & \ddots & \vdots \\
\gamma_s & \alpha_{s1} & \cdots & \alpha_{s,s-1} & 0 \\
\beta_1 & \cdots & \beta_{s-1} & \beta_s
\end{array}
\]
A control function \( u(t) \) is discretized on each subinterval \([t_i, t_{i+1}]\) by the approximation

\[
u_{\text{app}}^{(i)}(t; \hat{\mathbf{u}}), \quad t \in [t_i, t_{i+1}],
\]

where \( \hat{\mathbf{u}} = (u_0, u_1, \ldots, u_{P-1})^T \in \mathbb{U}^P \) is a finite-dimensional vector parametrizing the selection strategy for the control appearing in the Runge–Kutta scheme. The set of all admissible control approximations \( u_{\text{app}}(\cdot; \hat{\mathbf{u}}) \) on the whole interval \( I \) is denoted by \( \mathcal{U}_{\text{app}} \). For such a control approximation \( u_{\text{app}}^{(i)}(t; \hat{\mathbf{u}}) \) on \([t_i, t_{i+1}]\), the corresponding state approximation \( x_{\text{app}}(t; \hat{\mathbf{u}}) \) is obtained via an explicit \( s \)-step Runge–Kutta discretization scheme:

\[
\begin{align*}
x_{\text{app}}(t_{i+1}; \hat{\mathbf{u}}) &= x_{\text{app}}(t_i; \hat{\mathbf{u}}) + h \Phi(x_{\text{app}}(t_i; \hat{\mathbf{u}}), \hat{\mathbf{u}}, h), \quad i = 0, 1, \ldots, N - 1, \\
x_{\text{app}}(t_0; \hat{\mathbf{u}}) &= x_0,
\end{align*}
\]

where \( \Phi(\cdot, \cdot, h) \) is the characterizing function of the method with

\[
\Phi(x_{\text{app}}(t_j; \hat{\mathbf{u}}), \hat{\mathbf{u}}, h) := \sum_{j=1}^s \beta_j \left( A_{i,j} \eta_i^{(j)} + B_{i,j} u_{\text{app}}^{(i)}(t_j + \gamma_j h; \hat{\mathbf{u}}) \right),
\]

\[
\eta_i^{(j)} := x_{\text{app}}(t_j; \hat{\mathbf{u}}) + h \sum_{k=1}^{j-1} \alpha_{jk} \left( A_{i,k} \eta_i^{(k)} + B_{i,k} u_{\text{app}}^{(i)}(t_k + \gamma_k h; \hat{\mathbf{u}}) \right).
\]

and the notations \( A_{i,j} := A(t_i + \gamma_j h), B_{i,j} := B(t_i + \gamma_j h) \) for \( i = 0, \ldots, N - 1, j = 1, \ldots, s \). Suitable values for the coefficients in the Butcher array can be found in ref. [26].

Let us now consider examples for selection strategies used furtheron.

(i) Piecewise constant approximation:

\[
u_{\text{app}}^{(i)}(t; \hat{\mathbf{u}}) := u_i \quad \text{for} \ t \in [t_i, t_{i+1}], \quad i = 0, 1, \ldots, N - 1,
\]

with \( P = N \) independent selection variables \( u_i \in \mathbb{U} \).

(ii) Continuous and piecewise linear approximation:

\[
u_{\text{app}}^{(i)}(t; \hat{\mathbf{u}}) := u_i + \frac{t - t_i}{t_{i+1} - t_i}(u_{i+1} - u_i) \quad \text{for} \ t \in [t_i, t_{i+1}], \quad i = 0, 1, \ldots, N - 1,
\]

with \( P = N + 1 \) selection variables.

(iii) Independent selections at intermediate grid points \( t_i + \gamma_j h \) (cf. e.g. [4]):

\[
u_{\text{app}}^{(i)}(t_i + \gamma_j h; \hat{\mathbf{u}}) := u_{i,s+j-1}, \quad i = 0, 1, \ldots, N - 1, \quad j = 1, \ldots, s,
\]

with \( P = s \cdot N \) variables.

Here, each Runge–Kutta stage creates a new selection variable for each subinterval. Modified Euler’s method (see section 4 and figure 4 in Example 4.2) yields \( \gamma_1 = 0, \gamma_2 = 1/2 \) so that two selection variables \( u_{2i} \) and \( u_{2i+1} \) are chosen from \( \mathbb{U} \) for each subinterval \([t_i, t_{i+1}]\).

Similarly, Heun’s method (see section 4 and figure 3 in Example 4.2) yields \( \gamma_1 = 0, \gamma_2 = 1 \) so that two selection variables \( u_{2i} \) and \( u_{2i+1} \) are chosen for each subinterval \([t_i, t_{i+1}]\), although \( t_i + \gamma_2 h = t_{i+1} + \gamma_1 h \) coincides for \( i = 0, \ldots, N - 1 \) and this additional freedom of choice may result in two different values for \( u_{\text{app}}^{(i)}(t_{i+1}; \hat{\mathbf{u}}) \) and \( u_{\text{app}}^{(i+1)}(t_{i+1}; \hat{\mathbf{u}}) \).

Owing to the definitions (i)–(ii) and the convexity of \( \mathbb{U} \),

\[
u_{\text{app}}^{(i)}(t; \hat{\mathbf{u}}) \in \mathbb{U} \quad \text{(for all} \ t \in [t_i, t_{i+1}]\text{)}
\]

is guaranteed for all three presented strategies. Note that further selection strategies are possible, e.g. independent selections with additional continuity constraints at the inner grid.
Approximation of reachable sets for OCP

439

points \( t_i, i = 1, \ldots, N - 1 \), or additional equality constraints at those intermediate grid points \( t_i + \gamma_j h \), where different indices \( j \) produce the same intermediate grid point (i.e. points where \( \gamma_j = \gamma_k \) with \( j \neq k \)). Additional care has to be taken for other selection strategies to fulfill (7).

Thus, by this discretization, the infinite-dimensional OCP \( \mathcal{P}_i \) is approximated by the finite-dimensional convex programming problem

\[
\begin{cases}
\text{Maximize} & l^\top x_{\text{app}}(t_N; \hat{u}) \\
\text{subject to} & \hat{u} \in U^P \\
& x_{\text{app}}(t_{i+1}; \hat{u}) = x_{\text{app}}(t_i; \hat{u}) + h \Phi(x_{\text{app}}(t_i; \hat{u}), \hat{u}, h), \\
& i = 0, 1, \ldots, N - 1 \\
& x_{\text{app}}(t_0; \hat{u}) = x_0, \\
& \hat{u} \in U^P.
\end{cases} \tag{\text{CP}_i^1}
\]

Note that \( \hat{u} \) implicitly defines a control approximation \( u_{\text{app}}^{(i)}(\cdot; \hat{u}) \) on each subinterval \([t_i, t_{i+1}]\); compare the examples (i)–(iii).

We denote the optimal solution of (\text{CP}_i^1) by \( \hat{u}^* \).

If the conditions (\text{CP}_i^1) can be written with a finite number of affine inequalities, (\text{CP}_i^1) is a linear programming problem and called (\text{LP}_i^1), otherwise a nonlinear (convex) programming problem.

In the sequel, we investigate the simplest case in more detail: Euler’s method with piecewise constant control approximation. In this case, it is possible to derive explicit solutions for the finite-dimensional problems (\text{CP}_i^1). Nevertheless, every explicit Runge–Kutta method with the selection strategies (i)–(iii) will give a similar (but more complicated) representation. The explicit formula for the solution stresses the strong connection to set-valued methods, e.g. in [5,6,8], via support functions, respectively, supporting points.

In the case of Euler’s method, the characterizing function in (5) reduces to

\[ \Phi(x_{\text{app}}(t_i; \hat{u}), \hat{u}, h) = A(t_i)x_{\text{app}}(t_i; \hat{u}) + B(t_i)u_i. \]

The recursive evaluation in (5) for Euler’s method yields

\[ x_{\text{app}}(t_N; \hat{u}) = \left( \prod_{i=0}^{N-1} Q_i \right) x_0 + h \sum_{k=0}^{N-1} \left( \prod_{j=k+1}^{N-1} Q_j \right) B_k u_k \tag{8} \]

with \( Q_i := I + hA_i, A_j := A(t_j), B_k := B(t_k) \) and the \( n \times n \)-identity matrix \( I \).

The matrix product \( \prod \) is defined as

\[
\prod_{i=k}^{j} Q_i := \begin{cases} Q_j \cdot Q_{j-1} \cdots Q_k & \text{for } k \leq j, \\ I & \text{otherwise}. \end{cases}
\]

Introducing this expression for \( x_{\text{app}}(t_f; \hat{u}) \) in (\text{LP}_i^1) yields the linear programme

\[
\begin{cases}
\text{Maximize} & l^\top \left( \sum_{k=0}^{N-1} \left( \prod_{i=k+1}^{N-1} Q_i \right) B_k u_k \right) \\
\text{subject to} & u_k \in U, \quad k = 0, 1, \ldots, N - 1.
\end{cases} \tag{\text{LP}_i^2}
\]

Note that this linear programme has the same solution \( \hat{u} \) as (\text{LP}_i^1), whereas the optimal objective function values are different because we neglected constant terms.
To compute the objective function in (LP2) very efficiently, we introduce additional artificial variables

$$\lambda_N := I,$$
$$\lambda_i := Q_i^\top \lambda_{i+1} = \lambda_{i+1} + hA_i^\top \lambda_{i+1}, \quad (i = N - 1, N - 2, \ldots, 1).$$

These artificial variables are calculated backward in time and correspond to the discretized adjoint variable of the OCP (cf. [27]).

Then, (LP2) can be replaced by

$$\begin{align*}
\text{Maximize} & \quad \sum_{k=0}^{N-1} \lambda_{k+1}^\top B_k u_k \\
\text{subject to} & \quad u_k \in U, \quad k = 0, 1, \ldots, N - 1.
\end{align*}$$

(LP3)

Lemma 2.7 gives us

$$\sum_{k=0}^{N-1} \delta^*(\lambda_{k+1}, B_k U) = \sum_{k=0}^{N-1} \delta^*(B_k^\top \lambda_{k+1}, U)$$

as optimal value of (LP3) and hence \((u_0, u_1, \ldots, u_{N-1})\) with supporting points \(u_k \in Y(B_k^\top \lambda_{k+1}, U)\) as a solution.

In the special of box constraints, i.e. \(U = \{u \in \mathbb{R}^m | \underline{u} \leq u \leq \bar{u}\}\), we define \(S_k := (S_{1}^k, \ldots, S_{m}^k)^\top := B_k^\top \lambda_{k+1} \in \mathbb{R}^m\). As the objective function

$$\sum_{k=0}^{N-1} S_k u_k = \sum_{k=0}^{N-1} \sum_{j=1}^{m} S_{j}^k \cdot u_{j}^k$$

with \(u_k = (u_{1}^k, \ldots, u_{m}^k)^\top\)

is maximized, if each term \(S_{j}^k \cdot u_{j}^k\) is maximized, the solution of (LP3) is given for \(j = 1, \ldots, m, k = 0, \ldots, N - 1\) by the choice

$$u_{j}^k = \begin{cases} 
\underline{u}_j, & \text{if } S_{j}^k < 0, \\
\bar{u}_j, & \text{if } S_{j}^k > 0, \\
\text{arbitrary}, & \text{else}.
\end{cases}$$

### 3.3 Discrete reachable sets

Discrete reachable sets are the reachable sets of the discretized equations and could be defined as endpoints of discrete solutions of the following problem.

Given the data in Problem 2.1, the discretized problem depends on the choice of the set \(U_{\text{app}}\) of all discretized control functions and on the Runge–Kutta scheme.

**Problem 3.1** For a time discretization (4) with step-size \(h = (t_f - t_0)/N\) and a given discretized control function \(u_{\text{app}}(\cdot, \hat{u})\), we are looking for a solution \(x_{\text{app}}(\cdot, \hat{u})\) at the grid-points \(t_i, i = 0, 1, \ldots, N\), with

$$x_{\text{app}}(t_{i+1}; \hat{u}) = x_{\text{app}}(t_i; \hat{u}) + h \Phi(x_{\text{app}}(t_i; \hat{u}), \hat{u}, h), \quad \text{for } i = 0, 1, \ldots, N - 1,$$

$$x_{\text{app}}(t_0; \hat{u}) = x_0,$$

$$u_k \in U, \quad \text{for } k = 0, 1, \ldots, P - 1,$$

$$u_{\text{app}}(\cdot, \hat{u}) \in U_{\text{app}}.$$
DEFINITION 3.2  Consider Problem 3.1 with a time discretization (4) and let \( i \in \{0, 1, \ldots, N\} \). Then,

\[
\mathcal{R}_N(t_i, t_0, x_0) := \{ y \in \mathbb{R}^n \mid \exists u_{\text{app}}(\cdot; \hat{u}) \text{ discretized control function and} \\
\exists x_{\text{app}}(\cdot; \hat{u}) \text{ corresponding solution of Problem 3.1} \\
\text{with } x_{\text{app}}(t_i; \hat{u}) = y \}
\]

is called the discrete reachable set of the corresponding discretized control problem (9a)–(9d) for the time \( t_i \).

PROPOSITION 3.3  Consider Problem 3.1 with a bounded function \( B(\cdot) \) and a time discretization (4) and let \( i \in \{0, 1, \ldots, N\} \). Then, the corresponding discrete reachable set is convex, compact and nonempty.

Proof  For a chosen discretized control function \( u_{\text{app}}(\cdot, \hat{u}) \), the discrete solution is defined by (5). The discrete reachable set coincides with the union of all such discrete solutions for all feasible discretized control functions. In (5), common appearances of \( u_k, k = 0, \ldots, P - 1 \), and \( x_{\text{app}}(t_i; \hat{u}) \) must be grouped together (first within \( \eta^{(l)}_i \) and within \( \Phi(x_{\text{app}}(t_i; \hat{u}), \hat{u}, h) \), then also within \( x_{\text{app}}(t_{i+1}; \hat{u}) \)). Repeating this step for all \( i = 0, \ldots, N - 1 \), one arrives at a representation of the form

\[
x_{\text{app}}(t_N; \hat{u}) = \hat{A}_N x_0 + h \sum_{k=0}^{P-1} \hat{B}_k u_k
\]

with appropriate matrices \( \hat{A}_N \) and \( \hat{B}_k, k = 0, \ldots, P - 1 \). Finally, the union over all independent selections \( u_k \in U \) gives with Definition 2.6

\[
\mathcal{R}_N(t_f, t_0, x_0) = \hat{A}_N x_0 + h \sum_{k=0}^{P-1} \hat{B}_k U,
\]

which shows together with Lemma 2.7 the assertion.

In the special case of Euler and piecewise constant approximation of the controls, this corresponds to the union over all vectors \( \hat{u} \in \mathbb{R}^{m(N+1)} \). The explicit form of the discrete reachable set for Euler (cf. (8)) is

\[
\mathcal{R}_N(t_f, t_0, x_0) = \left( \prod_{i=0}^{N-1} Q_i \right) x_0 + h \sum_{k=0}^{N-1} \left( \prod_{i=k+1}^{N-1} Q_i \right) B_k U.
\]

Although only demonstrated for Euler’s method in section 3.2, the calculation of (CP\(_1^l\)) (respectively, its reformulation (LP\(_3^l\))) for every explicit Runge–Kutta scheme and selection strategy (i)–(iii) automatically involves the computation of supporting points in direction \( l \) of the discrete reachable set \( \mathcal{R}_N(t_f, t_0, x_0) \) (the maximizers in (CP\(_1^l\))). The optimal value of problem (CP\(_1^l\)) coincides with the support function \( \delta^*(l, \mathcal{R}_N(t_f, t_0, x_0)) \).
Propositions 2.5 and 3.3 justify the analogue result to (2) and (3):

\[ R_N(t_f, t_0, x_0) = \bigcap_{\|l\|_2 = 1} \left\{ x \in \mathbb{R}^n \mid l^T x \leq l^T x_{\text{app}}(t_f; \hat{u}^*) \right\}, \quad (10) \]

\[ R_N(t_f, t_0, x_0) = \text{co} \left( \bigcup_{\|l\|_2 = 1} \left\{ x_{\text{app}}(t_f; \hat{u}^*) \right\} \right). \quad (11) \]

In the implementation, only a finite number of different normed directions \( l_i, i = 1, \ldots, M \), is chosen in (10) and (11). This corresponds to the fact that for complex problems, we can compute a solution of (OCP) neither analytically nor numerically for all directions \( l \). Hence, we suggest to approximate (OCP) numerically and consider only a finite number of directions \( l_i, i = 1, \ldots, M \). This yields an approximation

\[ R_M(t_f, t_0, x_0) \approx R(t_f, t_0, x_0) \]

of the reachable set, which will be studied in the next proposition. The results in ref. [7] are slightly improved by the following result.

**Proposition 3.4** Proposition 2.5 suggests to use one of the approximations

\[ C \approx C_M := \bigcap_{i=1, \ldots, M} \left\{ x \in \mathbb{R}^m \mid \langle l^i, x \rangle \leq \delta^*(l^i, C) \right\}, \]

\[ C \approx \tilde{C}_M := \text{co} \left( \bigcup_{i=1, \ldots, M} \left\{ y(l^i, C) \right\} \right) \]

with arbitrary \( y(l^i, C) \in Y(l^i, C) \) for an arbitrary set \( C \in C(\mathbb{R}^n) \), where \( G_M := \{ l^i \mid i = 1, \ldots, M \} \subset S_{n-1} \).

If \( d_H(G_M, S_{n-1}) \leq \varepsilon \) with \( \varepsilon > 0 \), then

\[ d_H(C, \tilde{C}_M) \leq 2\|C\| \cdot \varepsilon, \quad \text{where} \quad \|C\| := d_H(C; \{0\}), \]

\[ d_H(C, C_M) \leq 3\|C\| \cdot \varepsilon, \quad \text{if} \quad \varepsilon \leq \frac{1}{2} \quad \text{and} \quad C_M \text{ is bounded}. \]

**Proof** First, we prove the estimation for \( \tilde{C}_M \).

Clearly, \( \tilde{C}_M \subset C \). Hence, choose \( c \in C \) and an appropriate \( \tilde{c}_M \in \tilde{C}_M \) with

\[ \|c - \tilde{c}_M\| = d(C, \tilde{C}_M) \]

The case that \( \|c - \tilde{c}_M\| = 0 \) is trivial. Otherwise, the characterization of the best approximation yields for all \( d \in \tilde{C}_M \),

\[ \langle c - \tilde{c}_M, d - \tilde{c}_M \rangle \leq 0 \implies \text{dist}(c, \tilde{C}_M)^2 = \langle c - \tilde{c}_M, c - d \rangle. \]

Set \( v := (1/\|c - \tilde{c}_M\|)(c - \tilde{c}_M) \), choose \( l^i \in G_M \) with \( \|v - l^i\| = \text{dist}(v, G_M) \) and set \( d = y(l^i, C) \). Then,

\[ \text{dist}(c, \tilde{C}_M) \leq \frac{1}{\text{dist}(c, \tilde{C}_M)} \cdot \langle c - \tilde{c}_M, c - y(l^i, C) \rangle, \]

\[ \leq \langle l^i, y(l^i, C) \rangle - \langle l^i, c \rangle - \langle v, y(l^i, C) - c \rangle, \]

\[ = \|l^i - v\| \cdot \|y(l^i, C) - c\| \leq 2\|C\| \cdot \varepsilon. \]

For the estimation of \( C_M \), it is clear that \( C \subset C_M \). Let us estimate \( \|C_M\| \) by \( \|C\| \). For this purpose, choose an arbitrary \( l \in S_{n-1} \) and an appropriate \( l^i \in G_M \) with \( \|l - l^i\| = \text{dist}(l, G_M) \).
Then,

\[ |\delta^*(l, C_M)| \leq |\delta^*(l, C_M) - \delta^*(l^i, C_M)| + |\delta^*(l^i, C_M)|, \]

\[ \leq ||l - l^i|| \cdot ||C_M|| + ||l^i, y(l^i, C)|| \leq \varepsilon \cdot ||C_M|| + \|C\|. \]

Using \(||C_M|| = \max_{l \in S_{n-1}} |\delta^*(l, C, M)|\), we arrive at the estimation \((1 - \varepsilon)||C_M|| \leq \|C\|\).

Now, choose \(l \in S_{n-1}\) such that \(d(C_M), C) = \delta^*(l, C_M) - \delta^*(l, C)\). Again, take an appropriate \(l^i \in G_M\) with \(||l - l^i|| = \text{dist}(l, G_M)\). Hence, we have for \(\varepsilon \leq 1/2\).

\[ d(C_M), C) = \delta^*(l, C_M) - \delta^*(l^i, C_M) + (\delta^*(l^i, C) - \delta^*(l, C)) \]

\[ \leq ||l - l^i|| \cdot ||C_M|| + ||l^i - l|| \cdot ||C|| + 0 \leq \varepsilon \cdot (||C_M|| + \|C\||) \]

\[ \leq \varepsilon \cdot \left( \frac{1}{1 - \varepsilon} \cdot \|C\| + \|C\| \right) \]

\[ \leq 3\|C\| \cdot \varepsilon. \]

One method for the choice of \(G_M\) in Proposition 3.4 is to parametrize the directions by spherical coordinates and use equidistant partitions on the parameter intervals for the angles (see ref. [7, section 3.1.2]).

### 3.4 Implementation

In the sequel, we briefly discuss some numerical methods, which are suitable for solving the discretized OCP \((CP_1)\). Of course, the choice of an appropriate method depends on the explicit representation of the control region \(U\). Hence, we restrict the discussion to convex sets \(U\) defined by

\[ U = \{u \in X \mid g_i(u) \leq 0, \quad i = 1, \ldots, r\}, \]

where \(X:=\{u \in \mathbb{R}^m \mid \tilde{A}_u = b, \quad u \geq 0\}\) with a matrix \(\tilde{A} \in \mathbb{R}^{q \times m}\) and the functions \(g_i(\cdot), \quad i = 1, \ldots, r, \) could be either linear or nonlinear.

If the functions \(g_i\) in (12) are affine, then problem \((CP_1)\) is a linear optimization problem and can be solved by the well-known simplex method or some interior point method [28], suitable for linear programmes. In the special case of an Euler approximation and \(U\) defined by box constraints only, a very efficient method is described in section 3.2.

If the functions \(g_i\) are convex and smooth, i.e. at least continuously differentiable, then the resulting problem \((CP_1)\) is a convex but nonlinear programming problem and the sequential quadratic programming method is appropriate provided the functions \(g_i\) are defined for infeasible points [29]. Alternatively, the method of feasible directions is applicable, especially if the functions \(g_i\) are only defined for admissible points [30].

If the functions \(g_i\) are convex but nonsmooth, the bundle method, respectively, the bundle trust region method (BT method) is suitable [31,32]. In addition, Kelley’s cutting plane method is also applicable [33].

In the case that the support function or the supporting points of the convex control set \(U\) are known, general control regions \(U\) can be approximated according to Proposition 3.4.
ALGORITHM 1  Let us formulate the overall method.

1. Choose a (pointwise explicit) Runge–Kutta method.
2. Choose a suitable selection strategy (e.g. one of (i)–(iii) in section 3.2) for the control approximation (in the optimal case, the order of the set-valued method is known as $O(h^p)$).
3. Choose a finite number of normed directions, i.e. $\mathcal{G}_M = \{ l^i | i = 1, \ldots, M \}$ with $d_{H}(\mathcal{G}_M, S_{n-1}) \leq \varepsilon$. If the order is known in (2), one could choose $\varepsilon = 1/N^p$ to guarantee an overall order of $p$.
4. For all directions $l^i \in \mathcal{G}_M, i = 1, \ldots, M$,
   
   Solve the optimal control problem $(OCP_{l^i})$ numerically with a direct solution method by discretizing $(OCP_{l^i})$ with the Runge–Kutta method in (2) and the selection strategy in (3).
   
   Solve the resulting finite-dimensional optimization problem $(CP_{l^i})$. The optimal value $\langle l^i, x_{\text{app}}(t_N; \hat{u}) \rangle$ of this problem gives the support function $\delta^* (l^i, \mathcal{R}_N(t_f, t_0, x_0))$ in direction $l^i$, the maximizer $x_{\text{app}}(t_N; \hat{u})$ yields a support point $y(l^i, \mathcal{R}_N(t_f, t_0, x_0))$ for this direction.
5. Plot the convex polygon with one of the representations

   $$
   \tilde{R}_{N,M}(t_f, t_0, x_0) = \text{co} \left( \bigcup_{i=1}^{M} \{ y(l^i, \mathcal{R}_N(t_f, t_0, x_0)) \} \right),
   $$

   $$
   R_{N,M}(t_f, t_0, x_0) = \bigcap_{i=1}^{M} \{ x \in \mathbb{R}^m | \langle l^i, x \rangle \leq \delta^* (l^i, \mathcal{R}_N(t_f, t_0, x_0)) \},
   $$

   which approximate the reachable set $\mathcal{R}(t_f, t_0, x_0)$ with the overall accuracy $O(\max\{h^p, \varepsilon\})$.

Often, a speed-up of the earlier algorithm can be achieved by exploiting the observation that the numerical solutions of $(OCP_{l^i})$ for neighbouring directions $l$, say $l^i$ and $l^{i+1}$, are usually close. Hence, in a so-called warm-start, the solution for direction $l^i$ is used as an initial guess for the neighbouring direction $l^{i+1}$.

4. Examples

In the sequel, we refer to the OCP, the differential equation (1a) and (1b), the control constraint (1c) and the control approximations discussed in (i)–(iii) in section 3.2.

The following Runge–Kutta methods are studied for the numerical computation of reachable sets:

|       | 0   | 0   | 0   | 0   | 0   | 0   |
|-------|-----|-----|-----|-----|-----|-----|
| Euler’s method | 0   | 1   | 1   | 0   | 1/2 | 1/2 |
| Heun’s method  | 1   | 1/2 | 1/2 | 0   | 0   | 1   |
| Modified Euler’s method | 0   | 1   | 0   | 0   | 0   | 0   |

For all numerical experiments, the number of directions $M$ in Proposition 3.4 is chosen via 1256 subintervals of $[0, 2\pi]$ for the angle parameter. The computations, which involve the solution of 1256 optimization problems for each step-size $h$, are done with a modified method in ref. [34] at a Linux PC with an Intel Pentium III 750 MHz processor and 1 GB RAM. For simplicity, the methods with different selection strategies are tested for time-independent two-dimensional problems (in which one could even calculate a theoretical solution for reference purposes). Nevertheless, the framework presented before is still valid and the methods
could be used also in more complicated problems met in practice (time-dependent and higher dimensional ones).

From Definition 2.8 and Proposition 2.9, it is clear that the approximation of the reachable set corresponds to a uniform convergence of the optimal value functions measured with the Hausdorff distance, whereas the approximation of trajectories corresponds to the uniform convergence of the maximizers and the choice of the Demyanov distance.

In all examples below, a combination method in [6,7] consisting of a (point-wise) Runge-Kutta method and a set-valued quadrature method is used with a huge number of subintervals to calculate a reference set \( \mathcal{R}_{\text{ref}}(t_f, t_0, x_0) \). For these reference methods, the order of convergence is already known so that a reference set yields approximations for the Hausdorff and the Demyanov distance

\[
d_H(\mathcal{R}(t_f, t_0, x_0), \mathcal{R}_N(t_f, t_0, x_0)) \approx d_H(\hat{\mathcal{R}}_{\text{ref}}(t_f, t_0, x_0), \mathcal{R}_N(t_f, t_0, x_0))
\]

\[
\approx \delta_H^M := \max_{i=1,\ldots,M} |\delta^*(l_i, \hat{\mathcal{R}}_{\text{ref}}(t_f, t_0, x_0)) - \delta^*(l_i, \mathcal{R}_N(t_f, t_0, x_0))|,
\]

\[
d_D(\mathcal{R}(t_f, t_0, x_0), \mathcal{R}_N(t_f, t_0, x_0)) \approx d_D(\hat{\mathcal{R}}_{\text{ref}}(t_f, t_0, x_0), \mathcal{R}_N(t_f, t_0, x_0))
\]

\[
\approx \delta_D^M := \max_{i=1,\ldots,M} \|Y(l_i, \hat{\mathcal{R}}_{\text{ref}}(t_f, t_0, x_0)) - Y(l_i, \mathcal{R}_N(t_f, t_0, x_0))\|_2
\]

by chosen directions \( l_i \in \mathcal{G}_M, i = 1, \ldots, M \) (the latter requires to take directions of unique support). By comparing the different values on the basis of the optimal value function in the calculation of the discrete reachable set, the order of convergence is estimated by assuming

\[
\max_{i=1,\ldots,M} |\delta^*(l_i, \hat{\mathcal{R}}_{\text{ref}}(t_f, t_0, x_0)) - \delta^*(l_i, \mathcal{R}_N(t_f, t_0, x_0))| = C \cdot \left(\frac{t_f - t_0}{N}\right)^p.
\]

For two subsequent values of \( N \) in the computed approximations, the left-hand side in (15) is calculated by the programme and as a consequence also \( p \) from the two equations (15) involving \( N_1 \) and \( N_2 \).

The analogue estimation is done for the Demyanov distance and the corresponding maximizers in the calculation of the discrete reachable set. The angle \( \varphi_j \) for the direction \( l^j = \left(\frac{\cos(\varphi_j)}{\sin(\varphi_j)}\right) \in \mathbb{R}^2, \varphi_j = 2j\pi/M, j = 0, \ldots, M - 1 \), for which the maximum in (13) and (14) is attained as shown in the following tables.

**Example 4.1** (see [35, Example in section 4])

Let us consider the following example with \( n = 2, m = 1, x_0 = (0, 0)^\top, I = [0, 1], U = [0, 1] \), and

\[
A(t) = \begin{pmatrix} 0 & 1 \\ 0 & 0 \end{pmatrix}, \quad B(t) = \begin{pmatrix} 0 \\ 1 \end{pmatrix}.
\]

In figure 1, approximations to the reachable set \( \mathcal{R}(1, 0, x_0) \) are shown: in the left picture, approximations for Euler’s method with piecewise constant selections are shown (first order of convergence) and in the right one, the corresponding ones for Heun’s method with continuous and piecewise linear control approximation (second order of convergence) are depicted. Here, the reference set is calculated by the combination method of set-valued iterated trapezoidal rule together with Heun’s method introduced in refs. [6,7] with \( \hat{N} = 1,000,000 \) and differs from the reachable set \( \mathcal{R}(1, 0, x_0) \) by \( \mathcal{O}(1/\hat{N}^2) \). In both cases, the reference set is drawn with solid lines. The dashed lines show the approximations for \( N = 10, 20, 40 \) for Euler’s method on the left picture (note the halfening of the distance of the upper right corner of the
sets when the number of subintervals is doubled). At the right one, the dashed lines show the approximations for $N = 1, 2, 4$ for Heun’s method (a smaller number of subintervals is chosen so that one could still see a difference of the corresponding approximations in figure 1). Note the more rapid convergence even for these small numbers of subintervals in comparison with Euler’s method.

Also in view of CPU times, Heun’s method is much faster than Euler’s method, if CPU times for (approximately) the same accuracies $\delta_H^M$ are compared. For instance, the CPU time for Euler’s method with $N = 320$ and $\delta_H^M = 0.0015625$ amounts to 401.62, cf. table 1, whereas the CPU time for Heun’s method with $N = 10$ and $\delta_H^M = 0.001247$ is 1.89 s, cf. table 2.

Table 1. Order of convergence for Euler’s method (left table: approximation of the reachable set, right table: approximation of the trajectories).

| N     | Hausdorff distance $\delta_H^M$ | Estimated order $p$ | Angle $\varphi_i$ | CPU (s) | N     | Demyanov distance $\delta_D^M$ | Estimated order $p$ | Angle $\varphi_i$ |
|-------|----------------------------------|---------------------|-------------------|--------|-------|---------------------------------|---------------------|------------------|
| 10    | 0.050000000                      | NaN                 | 0.005             | 1.27   | 10    | 0.13702925                      | NaN                 | 5.555            |
| 20    | 0.025000000                      | 1.00000             | 0.005             | 2.77   | 20    | 0.06806368                      | 1.00953             | 5.555            |
| 40    | 0.012500000                      | 1.00000             | 0.005             | 7.99   | 40    | 0.03392323                      | 1.00461             | 5.515            |
| 80    | 0.006250000                      | 1.00000             | 0.005             | 26.70  | 80    | 0.01731662                      | 0.97012             | 5.535            |
| 160   | 0.003125000                      | 1.00000             | 0.005             | 99.91  | 160   | 0.00861479                      | 1.00727             | 5.535            |
| 320   | 0.001562500                      | 1.00000             | 0.005             | 401.62 | 320   | 0.00426388                      | 1.01465             | 5.535            |
| 640   | 0.000781250                      | 1.00000             | 0.005             | 1673.58| 640   | 0.00209303                      | 1.02657             | 5.625            |

Table 2. Order of convergence for Heun’s method (left table: approximation of the reachable set, right table: approximation of the trajectories).

| N     | Hausdorff distance $\delta_H^M$ | Estimated order $p$ | Angle $\varphi_i$ | CPU (s) | N     | Demyanov distance $\delta_D^M$ | Estimated order $p$ | Angle $\varphi_i$ |
|-------|----------------------------------|---------------------|-------------------|--------|-------|---------------------------------|---------------------|------------------|
| 10    | 0.00124700                       | NaN                 | 3.095             | 1.89   | 10    | 0.06636590                      | NaN                 | 5.555            |
| 20    | 0.00031111                       | 2.00295             | 3.120             | 4.50   | 20    | 0.03273184                      | 1.01975             | 5.555            |
| 40    | 0.00007788                       | 1.99805             | 6.275             | 15.06  | 40    | 0.01668369                      | 0.97226             | 2.400            |
| 80    | 0.00001947                       | 1.99990             | 3.140             | 55.94  | 80    | 0.00848003                      | 0.97630             | 5.535            |
| 160   | 0.00000488                       | 1.99688             | 6.260             | 210.66 | 160   | 0.00419649                      | 1.01488             | 5.535            |
| 320   | 0.00000122                       | 1.99929             | 3.145             | 840.86 | 320   | 0.00205473                      | 1.03024             | 5.535            |
| 640   | 0.00000030                       | 2.00266             | 6.225             | 3482.91| 640   | 0.00099208                      | 1.05042             | 5.625            |
In table 1, the approximated distances (13) and (14) to the reference set and the estimated value of the convergence order \( p \) from (15) are shown.

Table 1 shows the expected convergence order 1 for the reachable set and the trajectories for Euler’s method. The approximated Hausdorff distance \( \delta_H^M \) is attained at the upper right corner. For Heun’s method with continuous and piecewise linear control approximation, table 2 shows order of convergence 2 for the reachable set and only order 1 for the trajectories.

**Remark 1** Let us consider again the problem stated in Example 4.1. As Veliov explains in ref. [35], the convergence of the trajectory could not be better than \( \mathcal{O}(h) \) in this example. During the calculation of the discrete reachable set, the direct solution method calculates first-order approximations to the control and to the state components (coordinates \( x_1 \) and \( x_2 \)) for all directions \( l^i \in \mathcal{G}_M \), \( i = 1, \ldots, M \). In figure 2, these approximations to the control and to the state components (coordinates \( x_1 \) and \( x_2 \)) are shown for Heun’s method with continuous and piecewise linear selections for \( l^i = \left( \cos(\varphi_i), \sin(\varphi_i) \right) \), and \( \varphi_i = \arctan(-0.5) \). The reference solution and control are computed by the method itself with \( N = 1280 \) (solid line), and in dashed lines, the approximations for \( N = 10, 20, 40 \). As it is clearly seen in this figure, the order of convergence is only 1.

**Example 4.2** (see [6, Example 4.4]) Let us consider the following example with \( n = 2 \), \( m = 2 \), \( x_0 = (0, 0)\top \), \( I = [0, 2] \), \( U = \{x \in \mathbb{R}^2 | \|x\|_2 \leq 1 \} \), and

\[
A(t) = \begin{pmatrix} 0 & 1 \\ -2 & -3 \end{pmatrix}, \quad B(t) = \begin{pmatrix} 1 \\ 0 \end{pmatrix}.
\]

This example introduces the nonlinear constraint ‘\( u_1^2 + u_2^2 \leq 1 \)’ for the control variable \( u = (u_1, u_2)\top \in U \).

The second-order approximations to the reachable set \( \mathcal{R}(2, 0, x_0) \) calculated by Heun’s method with piecewise constant controls, respectively, with piecewise linear and continuous controls are shown in figure 3. The set with the solid line shows the reference set (calculated with the combination method ‘RK(4)/set-valued Simpon’s rule’ with \( \hat{N} = 200,000 \) and accuracy \( \mathcal{O}(1/\hat{N}^4) \)) and the dashed lines represent the approximations for \( N = 5, 10, 20 \). At the left picture, the convergence order \( \mathcal{O}(h^2) \) can be seen by studying the upper part of the boundaries of the sets.

Both selection strategies seem to converge with order 2, which is assured by tables 3 and 4. Nevertheless, figure 4 shows that the choice of the selection strategies for the control should depend on the Runge–Kutta method. In figure 4, the piecewise constant selection strategy is

![Figure 2](image_url)  
**Figure 2.** First-order approximations to the control (left) and the state components \((x_1(\cdot)\) in the middle, \(x_2(\cdot)\) at the right) by Heun’s method for Example 4.1.
Figure 3. Second-order approximations to the reachable set for Heun’s method with piecewise constant control approximation (left), respectively, with piecewise linear and continuous control approximation (right) for Example 4.2.

compared with the independent control selections in $t_i$ and $t_i + (h/2)$ for modified Euler’s method (see (6)). The latter selection strategy destroys order of convergence 2 of the Runge–Kutta method. This is verified in tables 5 (order $O(h^2)$) and 6 (only order $O(h)$) for the convergence to the reachable set and the trajectories.

An overall comparison of the CPU times for comparable accuracies shows that methods of higher order are superior. The extremely high CPU times for the modified Euler’s method with independent selection strategy (iii) in table 6 are due to a highly oscillatory structure of the numerical solutions of the respective optimal control problems.

Modified Euler’s method with piecewise constant control approximation is faster than Heun’s method with piecewise constant control approximation at the same accuracy and order, cf. tables 3 and 5. Note that it is possible to prove that the discrete reachable sets are identical for both methods because of time invariance. Numerically, small differences appear

| $N$ | Hausdorff distance $\delta_H^M$ | Estimated order $p$ | Angle $\varphi_i$ | CPU (s) |
|-----|-------------------------------|---------------------|-------------------|--------|
| 5   | 0.10328935                    | NaN                 | 1.370             | 11.14  |
| 10  | 0.02307167                    | 2.16250             | 1.530             | 47.84  |
| 20  | 0.00521186                    | 2.14625             | 1.575             | 94.83  |
| 40  | 0.000123195                    | 2.08086             | 4.735             | 194.83 |
| 80  | 0.000029922                    | 2.04164             | 1.600             | 12660.20 |
| 160 | 0.000007372                    | 2.02105             | 4.745             | –      |

| $N$ | Demyanov distance $\delta_D^M$ | Estimated order $p$ | Angle $\varphi_i$ |
|-----|-------------------------------|---------------------|-------------------|
| 5   | 0.37223126                    | NaN                 | 0.905             |
| 10  | 0.07159599                    | 2.37825             | 0.885             |
| 20  | 0.01535558                    | 2.22112             | 4.025             |
| 40  | 0.00355544                    | 2.11066             | 4.025             |
| 80  | 0.00085565                    | 2.05493             | 4.025             |
| 160 | 0.00020992                    | 2.02719             | 4.025             |

Table 3. Order of convergence for Heun’s method with piecewise constant control approximation.

| $N$ | Hausdorff distance $\delta_H^M$ | Estimated order $p$ | Angle $\varphi_i$ | CPU (s) |
|-----|-------------------------------|---------------------|-------------------|--------|
| 5   | 0.04517018                    | NaN                 | 1.720             | 14.29  |
| 10  | 0.00772443                    | 2.54787             | 4.235             | 53.48  |
| 20  | 0.00203009                    | 1.92789             | 4.300             | 206.44 |
| 40  | 0.00051385                    | 1.98211             | 4.335             | 1568.60 |
| 80  | 0.00012897                    | 1.99429             | 1.210             | 17136.38 |
| 160 | 0.00003229                    | 1.99784             | 1.220             | –      |

| $N$ | Demyanov distance $\delta_D^M$ | Estimated order $p$ | Angle $\varphi_i$ |
|-----|-------------------------------|---------------------|-------------------|
| 5   | 0.16781545                    | NaN                 | 1.185             |
| 10  | 0.04611035                    | 1.86371             | 0.875             |
| 20  | 0.01077147                    | 2.09788             | 4.015             |
| 40  | 0.00257382                    | 2.06523             | 0.875             |
| 80  | 0.00062804                    | 2.03497             | 4.015             |
| 160 | 0.00015506                    | 2.01802             | 4.015             |

Table 4. Order of convergence for Heun’s method with continuous and piecewise linear control approximation.
in tables 3 and 5 due to different formulations of the corresponding optimization problems (CP). At a first glance, Heun’s method with continuous and piecewise linear control approximation seems to be slower than the previous two methods in view of CPU times, cf. table 4. But taking into account the Hausdorff distance $\delta_H$ and comparing CPU times for (approximately) the same accuracy, it turns out that Heun’s method with continuous and piecewise linear control approximation is actually not worse than modified Euler’s method (compare $N = 20$ in table 5 and $N = 10$ in table 4).

Table 7 shows a comparison on results concerning the order of convergence and classical ODEs in ref. [26], for optimal control problems in refs. [3,4] and for the set-valued case in the approximation of reachable sets refs. [5,7,8,10].

In the last column, citations for proofs and corresponding assumptions on smoothness and on the problem are additionally mentioned. The indication ‘<2’ based on the work [3] means that the conditions for order 2 (the main topic of this article) are not all fulfilled. In ref. [4], only

### Table 5. Order of convergence for the modified Euler’s method with piecewise constant control approximation.

| $N$ | Hausdorff distance $\delta_H$ | Estimated order $p$ | Angle $\varphi_i$ | CPU (s) | | Hausdorff distance $\delta_H$ | Estimated order $p$ | Angle $\varphi_i$ | CPU (s) |
|-----|-----------------------------|---------------------|------------------|--------| | | | | |
| 5   | 0.10328935                  | NaN                 | 1.370            | 7.19   | 5    | 0.37223121                  | NaN                 | 0.905 |
| 10  | 0.02307167                  | 2.16250             | 1.530            | 28.07  | 10   | 0.07159599                  | 2.37825             | 0.885 |
| 20  | 0.00521186                  | 2.14625             | 1.575            | 104.29 | 20   | 0.01535559                  | 2.22112             | 4.025 |
| 40  | 0.00123195                  | 2.08086             | 4.735            | 639.13 | 40   | 0.00355571                  | 2.11056             | 4.025 |
| 80  | 0.00029922                  | 2.04164             | 1.600            | 7670.68| 80   | 0.00085566                  | 2.05503             | 0.885 |

### Table 6. Order of convergence for the modified Euler’s method with independent selection strategy (iii).

| $N$ | Hausdorff distance $\delta_H$ | Estimated order $p$ | Angle $\varphi_i$ | CPU (s) | | Hausdorff distance $\delta_H$ | Estimated order $p$ | Angle $\varphi_i$ | CPU (s) |
|-----|-----------------------------|---------------------|------------------|--------| | | | | |
| 5   | 0.83583108                  | NaN                 | 4.030            | 46.08  | 5    | 1.03202096                  | NaN                 | 0.730 |
| 10  | 0.33319435                  | 1.32685             | 0.855            | 210.87 | 10   | 0.36562913                  | 1.49702             | 3.850 |
| 20  | 0.15333206                  | 1.11970             | 5.340            | 1581.42| 20   | 0.16060144                  | 1.18690             | 3.760 |
| 40  | 0.07575471                  | 1.01725             | 5.360            | 28884.83| 40   | 0.07933801                  | 1.01740             | 4.720 |
| 80  | 0.03762644                  | 1.00959             | 2.225            | –      | 80   | 0.03952243                  | 1.00534             | 4.720 |
Table 7. Comparison of order results of first- and second-order methods.

| RK-method   | Order for ODE | Selection strategy | Order for OCP in [4] | Order for OCP in [3] | Order for DI |
|-------------|---------------|--------------------|----------------------|----------------------|-------------|
| Euler       | 1             | Piecewise constant | 1                    | <2                   | 1 in [5]    |
| Modified Euler | 2            | Piecewise constant | –                    | 2                    | 2 in [10]   |
| Heun        | 2             | Piecewise constant | –                    | 2                    | 2 in [8]    |
|             | Two independent ones | –                  | <2                   | 2                    | 2 in [8]    |

results for the independent selection strategy are mentioned and all weights $\beta_i$ are assumed to be positive.

This is not the case for modified Euler, and even an example in refs. [3,4] shows that for a special linear control problem without restrictions on the control and quadratic cost functional, this method with the strategy (iii) of two independent selections in $[t_j, t_{j+1}]$ gives no convergence at all. The minimal order 1 is proven and an order breakdown to 1 for a special example is noticed for a set-valued realization of the algorithm in ref. [10]. Note that additional smoothness assumptions on the optimal solution as well as the coercivity are formulated in refs. [3,4] contrary to the results cited in the last column.

Let us remark that the last column in table 7 coincides with the numerical experiments of the direct solution method in Examples 4.1 and 4.2.

5. Outline of further research

It is known that set-valued quadrature methods in ref. [6] could lead to an order of convergence greater than 2 if the problem satisfies additional smoothness conditions, cf. [7]. In this case, selection strategies with piecewise constant controls are no longer appropriate. Preliminary computer experiments with the classical Runge–Kutta method show that order of convergence greater than 2 is attainable. However for these Runge–Kutta methods, suitable selection strategies for the approximation of reachable sets have to be studied in more detail. In this context, additional difficulties arise if state constraints are present, because these constraints should be fulfilled also at the intermediate stages of the Runge-Kutta scheme (as in ref. [9]).

Further research can be conducted towards the study of Runge–Kutta schemes as in refs. [36–38], where the selection strategy is motivated by Volterra series involving multiple control integrals. In the special case of two selections per Runge–Kutta step, this leads to alternative selection sets of the type

$$\left( u^{(i)}_{\text{app}}(t_i + \gamma_1 h; \hat{u}), u^{(i)}_{\text{app}}(t_i + \gamma_2 h; \hat{u}) \right) \in \hat{U} \subset U \times U,$$

where $U \times U$ corresponds to case (iii) of independent selections in section 3.

This set $\hat{U}$ can be described by finitely many nonlinear inequalities and equalities, which can be easily imposed as additional constraints in the discretized optimal control problems of Algorithm 1.

The proposed method itself can be easily adapted to the calculation of convex reachable sets for special nonlinear differential inclusions. For the numerical solution of discretized optimal control problems, efficient algorithms are available, cf. [27,34,39]. In the more general case of nonconvex reachable sets, suitable modifications of our approach have to be studied. Theoretical results in this direction can be found in refs. [5,40] for Runge–Kutta methods of orders 1 and 2. A survey of other methods is given in refs. [9,20].
However, those Runge–Kutta methods with appropriate selection strategies, which show higher order of convergence in the linear case, are worth being investigated also in the nonlinear case. In addition, these methods have to be compared with set-valued Runge–Kutta methods on the basis of the set arithmetics, cf. [9], which also work in the general nonlinear case. First step in this direction can be found in ref. [9, Example 5.3.1].

Acknowledgements

The authors would like to express their thanks to the anonymous referees and the co-editor for their valuable comments and suggestions.

References

[1] Varaiya, P., 2000. Reach set computation using optimal control. In: M.K. Inan and R.P. Kurshan (Eds), Verification of digital and hybrid systems. Proceedings of the NATO ASI, Antalya, Turkey, May 26–June 6, 1997, NATO Science Series F: Computer and Systems Sciences, Vol. 170 (Berlin: Springer), pp. 323–331.

[2] Malanowski, K., Bűskens, C. and Maurer, H., 1997, Convergence of approximations to nonlinear optimal control problems. In: A. Fiacco (Ed.) Mathematical Programming with Data Perturbations, Lecture Notes in Pure and Applied Mathematics, Vol. 195 (New York: Dekker), pp. 253–284.

[3] Dontchev, A.L., Hager, W.W. and Veliov, V.M., 2000, Second-order Runge–Kutta approximations in control constrained optimal control. SIAM Journal on Numerical Analysis, 38(1), 202–226.

[4] Hager, W.W., 2000, Runge–Kutta methods in optimal control and the transformed adjoint system. Numerische Mathematik, 87(2), 247–282.

[5] Dontchev, A.L. and Farkhi, E., 1989, Error estimates for discretized differential inclusions. Computing, 41(4), 349–358.

[6] Baier, R. and Lempio, F., 1994, Computing aumann’s integral. In: A.B. Kurzhanski and V.M. Veliov (Eds), Modeling Techniques for Uncertain Systems, Proceedings of a Conference held in Sopron, Hungary, June 6–10, 1992, Progress in Systems and Control Theory, Vol. 18 (Basel: Birkhäuser), pp. 71–92.

[7] Baier, R., 1995, Mengenwertige integration und die diskrete approximation erreichbarer Mengen. Bayreuther Mathematische Schriften, 50, 1–248.

[8] Veliov, V.M., 1992, Second order discrete approximation to linear differential inclusions. SIAM Journal on Numerical Analysis, 29(2), 439–451.

[9] Chahma, I.A., 2003, Set-valued discrete approximation of state-constrained differential inclusions. Bayreuther Mathematische Schriften, 67, 3–162.

[10] Baier, R., 2005, Selection strategies for set-valued Runge–Kutta methods. In: Z. Li, L.G. Vulkov and J. Wasniewski (Eds), Numerical Analysis and Its Applications, Third International Conference, NAA 2004, Rousse, Bulgaria, June 29–July 3, 2004, Lecture Notes in Control and Information Sciences, Vol. 3401 (Berlin–Heidelberg: Springer), pp. 49–57

[11] Kurzhanski, A.B. and Vályi, I., 1997, Systems & Control: Foundations & Applications. Ellipsoidal Calculus for Estimation and Control (Boston–Basel–Berlin: Birkhäuser).

[12] Gayek, J.E., 1986, Approximating reachable sets for a class of linear control systems. International Journal of Control, 43(2), 441–453.

[13] Kurzhanski, A.B. and Varaiya, P., 2000, Ellipsoidal techniques for reachability analysis: internal approximation. Systems and Control Letters, 41(3), 201–211.

[14] Kurzhanski, A.B. and Varaiya, P., 2002, On ellipsoidal techniques for reachability analysis, part II: internal approximations, box-valued constraints. Optimization Methods and Software, 17(2), 207–237.

[15] Kurzhanski, A.B. and Varaiya, P., 2002, On ellipsoidal techniques for reachability analysis, part I: external approximations. Optimization Methods and Software, 17(2), 177–206.

[16] Blagodatskikh, V.I. and Filippov, A.F., 1986, Differential inclusions and optimal control. In Proceedings of the Steklov Institute of Mathematics, Vol. 4 (North-Holland:Amsterdam), pp. 199–259.

[17] Aubin, J.-P. and Cellina, A., 1984, Differential Inclusions. Grundlehen der mathematischen Wissenschaften, Vol. 264 (Berlin–Heidelberg–New York–Tokyo: Springer Verlag).

[18] Saint-Pierre, P., 1994, Approximation of the viability kernel. Applied Mathematics and Optimization, 29(2), 187–209.

[19] Filippov, A.F., 1988, Mathematics and Its Applications (Soviet Series). Differential Equations with Discontinuous Righthand Sides (Dordrecht–Boston–London: Kluwer Academic Publishers).

[20] Dontchev, A.L. and Lempio, F., 1992, Difference methods for differential inclusions: a survey. SIAM Review, 34(2), 263–294.

[21] Aumann, R.J., 1965, Integrals of set-valued functions. Journal of Mathematical Analysis and Applications, 12(1), 1–12.

[22] Sonneborn, L.M. and van Vleck, F.S., 1965, The bang–bang principle for linear control problems. Journal of the Society for Industrial and Applied Mathematics, Series A: Control, 2(2), 151–159.
[23] Hiriart-Urruty, J.-B. and Lemaréchal, C., 1993, *Convex Analysis and Minimization Algorithms I. Fundamentals*, Grundlehren der mathematischen Wissenschaften, Vol. 305 (Berlin–Heidelberg–New York–London–Paris–Tokyo–Hong Kong–Barcelona–Budapest: Springer).

[24] Rockafellar, R.T., 1972, *Convex Analysis*, Princeton Mathematical Series, (2nd edn) Vol. 28 (Princeton, NJ: Princeton University Press).

[25] Diamond, P., Kloeden, P., Rubinov, A. and Vladimirov, A., 1997, Comparative properties of three metrics in the space of compact convex sets. *Set-Valued Analysis*, 5(3), 267–289.

[26] Butcher, J.C., 1987, *The Numerical Analysis of Ordinary Differential Equations. Runge–Kutta and General Linear Methods* (Chichester–New York–Brisbane–Toronto–Singapore: John Wiley and Sons).

[27] Büskens, C., 1998, Optimierungsmethoden und Sensitivitätsanalyse für optimale Steuerprozesse mit Steuer- und Zustandsbeschränkungen. PhD thesis, Westfälische Wilhems-Universität Münster (Germany), Fachbereich Mathematik.

[28] Wright, S.E., 1997, *Primal–Dual Interior-Point Methods* (Philadelphia, PA: SIAM).

[29] Schittkowsk, K., 1983, On the convergence of a sequential quadratic programming method with an augmented Lagrangian line search function. *Mathematiche Operationsforschung und Statistik, Series Optimization*, 14(2), 197–216.

[30] Zoutendijk, G., 1960, *Methods of Feasible Directions* (Amsterdam: Elsevier Publishing Company).

[31] Lemaréchal, C., Strodiot, J.J. and Bihain, A., 1981, On a bundle algorithm for nonsmooth optimization. In O.L. Mangasarian, R.R. Meyer and S.M. Robinson (Eds) *Nonlinear Programming*, Vol. 4 (New York: Academic Press), pp. 245–282.

[32] Schramm, H., 1989, Eine Kombination von bundle- und trust-region-Verfahren zur Lösung nichtdifferenzierbarer Optimierungsprobleme. *Bayreuther Mathematische Schriften*, 30, 1–205.

[33] Kelley, J.E., 1960, The cutting-plane method for solving convex programs. *Journal of the Society for Industrial and Applied Mathematics*, 8, 703–712.

[34] Gerdts, M., 2001, Numerische Methoden optimaler Steuerprozesse mit differential-algebraischen Gleichungssystemen höheren Indexes und ihre Anwendungen in der Kraftfahrzeugsimulation und Mechanik. *Bayreuther Mathematische Schriften*, 61, 1–139.

[35] Veliov, V.M., 1989, Approximations to differential inclusions by discrete inclusions. Working Paper WP-89-017, IIASA, Laxenburg, Austria.

[36] Lempio, F. and Veliov, V.M., 1998, Discrete approximations of differential inclusions. *Bayreuther Mathematische Schriften*, 54, 149–232.

[37] Ferretti, R., 1997, High-order approximations of linear control systems via Runge–Kutta schemes. *Computing*, 58(4), 351–364.

[38] Grüne, L. and Kloeckl, P., 2001, Higher order numerical schemes for affinely controlled nonlinear systems. *Numerische Mathematik*, 89(4), 669–690.

[39] Gerdts, M., 2003, Direct shooting method for the numerical solution of higher index DAE optimal control problems. *Journal of Optimization Theory and Applications*, 117(2), 267–294.

[40] Veliov, V.M., 1989, Second order discrete approximations to strongly convex differential inclusions. *Systems and Control Letters*, 13, 263–269.