

Optimalsteuerung kooperierender Mehrfahrzeugsysteme

Optimal Control of Cooperative Multi-Vehicle Systems

Christian Reinl, Markus Glocker, Oskar von Stryk, Technische Universität Darmstadt

Preprint of an article which appeared in
at-Automatisierungstechnik, Vol. 57, No. 6, pp. 296-305, 2009

Nichtlineare hybride dynamische Systemmodelle kooperativer Optimalsteuerungsprobleme ermöglichen eine enge und formale Kopplung von diskreter und kontinuierlicher Zustandsdynamik, d.h. von dynamischer Rollen-, Aktionszuweisung mit wechselnder physikalischer Fahrdynamik. In den resultierenden gemischt-ganzzahligen Mehrphasen-Optimalsteuerungsproblemen können Beschränkungen an diskrete und kontinuierliche Zustands- und Steuervariablen berücksichtigt werden, z.B. Formations- oder Kommunikationsanforderungen. Zwei numerische Verfahren werden untersucht: ein Dekompositionsansatz mit Branch-and-Bound und direktem Kollokationsverfahren sowie die Approximation durch große, gemischt-ganzzahlige lineare Optimierungsaufgaben. Die Verfahren werden auf exemplarische Problemstellungen angewendet: Die simultane Wegpunktfolgen- und Trajektorienoptimierung von Luftfahrzeugen sowie die Optimierung von Rollenverteilung und Trajektorien im Roboterfußball.

Nonlinear hybrid dynamical systems for modeling optimal cooperative control enable a tight and formal coupling of discrete and continuous state dynamics, i.e. of dynamic role and task assignment with switching dynamics of motions. In the resulting mixed-integer multi-phase optimal control problems constraints on the discrete and continuous state and control variables can be considered, e.g., formation or communication requirements. Two numerical methods are investigated: a decomposition approach using branch-and-bound and direct collocation methods as well as an approximation by large-scale, mixed-integer linear problems. Both methods are applied to example problems: the optimal simultaneous waypoint sequencing and trajectory planning of a team of aerial vehicles and the optimization of role distribution and trajectories in robot soccer.

Schlagwörter: nichtlineare gemischt-ganzzahlige Optimalsteuerung, hybride dynamische Systeme, wechselnde Dynamik und Beschränkungen, kooperative Mehrfahrzeugsysteme

Keywords: nonlinear mixed-integer optimal control, hybrid dynamical systems, switched dynamics and constraints, cooperative multi-vehicle systems

1 Einleitung und Stand der Forschung

Kooperative Optimalsteuerungsprobleme für Mehrfahrzeugsysteme treten in einer Fülle neuer Anwendungen auf. Diese reichen von kooperativen Überwachungsaufgaben unbemannter Flugsysteme über Aufklärungsaufgaben mobiler Sensornetzwerke zur kooperativen Trajektorien- und Aufgabenplanung für mobile autonome Roboterteams. Ansätze und Methoden zur Steuerung von kooperativen Mehrfahrzeugsystemen, oder auch Mehrroboter-Systemen, sind äußerst vielfältig und hängen unter anderem von den Fahrzeugklassen, den Aufgaben des Systems und den

zur Verfügung stehenden Sensordaten ab. Einen breiten Überblick liefert [17] und die darin zitierte Literatur.

Die hier vorgestellte Methodik befasst sich mit der optimalen Steuerung kooperativer Gesamtsysteme, in denen die Bewegungsdynamik der Fahrzeuge entscheidenden Einfluss auf die Qualität der Aufgabenausführung hat. Typische Fragestellungen sind z.B. „Wieviel Zeit benötigt eine Flotte von Luftfahrzeugen zur kooperativen Erkundung eines Gebietes?“ oder „Was ist die beste Verteilung von Teilaufgaben an Roboter eines Teams bei gleichzeitiger Optimierung der Bewegungstrajektorien?“.

Die zahlreichen Ansätze der dynamischen Koordination und Rollenverteilung für Roboterteams reichen von komplett verhaltensbasierten Methoden über Publish/Subscribe-Architekturen und Markt-basierte Ansätze bis hin zu mehrschichtigen Architekturen [19]. Ein Framework für hybride Systeme zur Rollenzuweisung für kooperative Mehrrobotersysteme wurde in [6] vorgestellt.

Arbeiten von *Engell* (z.B. [18, 23]) oder *Bemporad* [2] befassten sich mit gemischt-ganzzahligen linearen Problemen zur approximativen Optimalsteuerung hybrider Systeme. Zuverlässige Lösungsalgorithmen dafür wurden kontinuierlich weiterentwickelt [1]. Zur Anwendung beim Roboterwettkampf RoboFlag wurde in [8] bereits eine effiziente Methode zur Modellierung komplexer Mehrrobotersysteme entwickelt. Unter Verwendung eines vereinfachten linearen Bewegungsdynamikmodells werden die dynamischen Mehrroboter-Planungsprobleme als gemischt-logisches lineares dynamisches System formuliert.

Zur Beschreibung von Rollen- und Aktionswechsell während der kooperativen Aufgabenerfüllung werden in der vorliegenden Arbeit hybride Zustandsautomaten vorgeschlagen, die aus diskreten (Rollen, Aktionen) und kontinuierlichen Zuständen, welche durch Bewegungsdifferentialgleichungen und algebraische Beschränkungen charakterisiert werden, bestehen. Auf Grundlage nichtlinearer hybrider dynamischer Systeme erlaubt diese Modellierung eine sehr enge und formelle Kopplung diskreter und kontinuierlicher Zustandsdynamik. Im Gegensatz zu bisherigen Modellen der Mehrroboterkooperation mit allgemeinen hybriden dynamischen Zustandsautomaten (z.B. [6]) wird hier die noch wenig untersuchte Frage der Optimalsteuerung dieser Modelle betrachtet [11].

In der vorliegenden Arbeit werden nun Modelle der auftretenden Problemstellungen sowohl in gemischt-ganzzahlige nichtlineare Optimalsteuerungsprobleme als auch in stückweise lineare, zeitdiskrete Systeme überführt. Deren Vereinbarkeit mit Kollokationsmethoden sowie die prinzipielle Eignung der Ansätze für Probleme der Mehrfahrzeugkooperation wird abschließend diskutiert.

2 Modellbildung, hybride Automaten und Optimalsteuerung

Charakteristisch für die Optimierung hybrider Systeme sind dynamische Prozesse in wechselnden Betriebsmodi, die so geplant und gesteuert werden sollen, dass eine systemübergreifende Kostenfunktion minimiert wird. Die komplexen Strukturen, die sich durch Kopplung diskreter und kontinuierlicher Zustände ergeben, lassen sich direkt durch hybride Automaten abbilden.

2.1 Hybride Zustandsautomaten

Ein hybrider Automat (Abk. HA) [13] $H = (Q, E, \mathbb{X}, \mathbb{U}, \text{init}, \text{inv}, \text{flow}, \text{jump}, \text{event})$ besteht aus einem endlichen, gerichteten Multigraphen (Q, E) , mit

Knoten $q \in Q$, die als Systemzustand bezeichnet werden, und Kanten $e \in E \subset Q \times Q$, den Systemschaltungen. Die Menge $\mathbb{X} = \{x_1, \dots, x_{n_x}\}$ fasst kontinuierliche Zustandsvariablen und $\mathbb{U} = \{u_1, \dots, u_{n_u}\}$ Steuervariablen des betrachteten Systems zusammen. Durch die Abbildungen *init*, *jump* und *event* werden einer Kante jeweils eine Anfangsbedingung, Sprungbedingung und Ereignisse zugeordnet. Innerhalb eines Knoten q entwickelt sich das System gemäß der Dynamik $\dot{x} = f_q(x(t), u(t))$, die durch *flow* zugeordnet ist, unter Berücksichtigung der Nebenbedingungen $g_q(x(t), u(t)) \leq 0$, zugeordnet durch Invarianten *inv*.

HA werden zum Aufbau und zur Beschreibung mehrschichtiger Steuerungsarchitekturen auf Basis physikalischer Bewegungsdynamik eingesetzt [3]. In vielen Anwendungen stellen sie die Verbindung zwischen den Mehrfahrzeugsystemen und der Theorie hybrider Systeme dar, z.B. [26]. Hybride Automaten existieren in verschiedenen Varianten, die sich u.a. anhand einer gegebenenfalls vorhandenen Taktung und anhand des behandelbaren Grades an Nichtlinearität der algebraischen Ausdrücke und Differentialgleichungen unterscheiden. Eine Verbindung zur Optimalsteuerung wird durch die Erweiterung des HA durch die Abbildungen *cost_e* und *cost_q* erreicht, die den Knoten und Kanten Lauf- bzw. Schaltkosten zuordnet.

2.2 Anwendung auf Mehrfahrzeugsysteme

Ein Team von n_v kooperierenden Fahrzeugen wird betrachtet, zur Veranschaulichung o.B.d.A., in \mathbb{R}^2 . Gesucht sind die Steuerungen unter Einhaltung der Bewegungsdifferentialgleichungen und physikalischen Beschränkungen, so dass ein gewünschtes Systemverhalten (z.B. Formationwechsel, taktische Kooperation) optimiert wird.

Die Knoten q eines hybriden Automaten repräsentieren hier bestimmte Bewegungsmuster, Rollenverteilungen und Forderungen nach Formationen. Kanten stellen Veränderung dieser diskreten Zustände, z.B. durch Kopplung, Massenverlust oder Taktikwechsel, dar. Ein diskreter Zustand

$$q_k := q(t), t_{k-1} < t \leq t_k, k = 1, \dots, n_s$$

(mit $t_0 = 0, t_{n_s} = t_f$) der den aktiven Knoten markiert, kann seinen Wert zu einem (variablen oder festen) Schaltzeitpunkt t_k ändern. Die Zahl der maximalen Schaltzeitpunkte n_s im System sei dabei zunächst als fest angenommen. Die Übergänge von x^i und u^i folgen der durch die *jump*-Bedingung an Kante $e \in E$ zugeordneten Vorschrift

$$\begin{pmatrix} x(t_{s,k} + 0) \\ u(t_{s,k} + 0) \end{pmatrix} = j_e(x(t_k - 0), u(t_k - 0)), \quad (1)$$

mit $t \pm 0 := \lim_{\epsilon \rightarrow 0, \epsilon > 0} t \pm \epsilon$. So lassen sich sowohl stetige (z.B. Gangschaltung) als auch sprunghafte Übergänge (z.B. bei Kollision) von Ort und Geschwindigkeit beschreiben.

Jedem Zustand $q \in Q$ wird durch *flow* eine Dynamik

$$\dot{\mathbf{x}} := \begin{pmatrix} \dot{\mathbf{x}}^1 \\ \vdots \\ \dot{\mathbf{x}}^{n_v} \end{pmatrix} = \begin{pmatrix} \mathbf{f}_q^1(\mathbf{x}^1(t), \mathbf{u}^1(t)) \\ \vdots \\ \mathbf{f}_q^{n_v}(\mathbf{x}^{n_v}(t), \mathbf{u}^{n_v}(t)) \end{pmatrix} = \mathbf{f}_q(\mathbf{x}, \mathbf{u}), \quad (2)$$

mit $q \in Q$ und $(\mathbf{u}^1, \dots, \mathbf{u}^{n_v})^T =: \mathbf{u} \in \mathbb{U}$ zugeordnet. Dabei entspricht $\dot{\mathbf{x}}^i(t) = \mathbf{f}_q^i(\mathbf{x}^i(t), \mathbf{u}^i(t))$ der Bewegungsdynamik des Fahrzeugs i mit kontinuierlichem Systemzustand $\mathbf{x}^i(t) \in \mathbb{R}^{n_{x,i}}$ (Position, Geschwindigkeit, Orientierung, ...) und kontinuierlichen Steuervariablen $\mathbf{u}^i(t) \in \mathbb{R}^{n_{u,i}}$ (Beschleunigung, Lenkwinkel, ...). Abhängig vom Detaillierungsgrad des physikalischen Modells können diese Bewegungsdifferentialgleichungen von einfachen kinematischen bis hin zu komplexen nichtlinearen Fahrzeugdynamikmodellen reichen. Der diskrete und kontinuierliche Anfangszustand zur Startzeit sei hier bekannt ($q(t_0) = q_0$, $\mathbf{x}^i(0) = \mathbf{x}_0^i$), kann aber i. Allg. genauso wie die Endzeit t_f und Endzustände frei oder fest gewählt sein.

Zustände und Steuervariablen unterliegen Beschränkungen

$$\mathbf{g}_q^i(\mathbf{x}^i(t), \mathbf{u}^i(t)) \leq 0, \quad (3)$$

die sich meist direkt aus der Physik des Fahr- oder Flugzeugs (z.B. maximale Bewegungsradien, Höchstgeschwindigkeiten) und dessen Umgebung ergeben.

Die Komponenten \mathbf{x}^i verschiedener Fahrzeuge des Systems sind eng aneinander gekoppelt; etwa durch physikalische Bedingungen, wie die Kollisionsvermeidung

$$\forall i_1, i_2 \in \{1, 2, \dots, n_v\}, i_1 \neq i_2 : \mathbf{g}_q^{coll}(\mathbf{x}^{i_1}, \mathbf{x}^{i_2}) \leq 0, \quad (4)$$

durch Formations- und Konnektivitätsforderungen (vgl. Abschnitt 4.2.3), oder durch eine gemeinsame Aufgabe, welche in einer Zielfunktion $J(\mathbf{x}, \mathbf{u})$ oder durch Endbedingungen $(\mathbf{x}(t_f), q(t_f)) \in \mathbb{X}_f \times Q_f$ beschrieben wird.

2.3 Optimalsteuerung

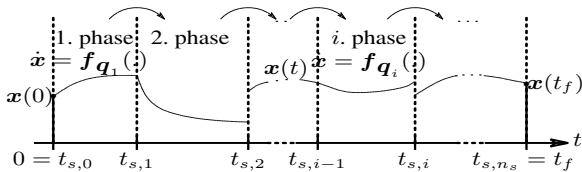


Bild 1: Kontinuierliche Zustandsgröße in n_s Phasen, Phasenübergänge treten an diskreten Schaltzeitpunkten $t_{s,k}$ auf. Die Folge der q_i ist dabei vorab nicht vorgegeben.

Für jede zulässige Sequenz $(q_k)_{k=1}^{n_s}$ diskreter Systemzustände, wobei der Startzustand $\mathbf{x}(0)$ und auch der Steuerverlauf $\mathbf{u}(t)$, $0 \leq t \leq t_f$, gegeben sind, kann die Systemtrajektorie $\mathbf{x}(t)$, $0 \leq t \leq t_f$, unter nicht zu einschränkenden Voraussetzungen aus Gl. (2), selbst unter Berücksichtigung von Sprung- und Schaltbedingungen (1), eindeutig bestimmt werden.

Als Summe der zur Sequenz (q_k) gehörigen Lauf- und Schaltkosten resultieren die Gesamtkosten des Optimalsteuerungsproblems

$$\min_{\mathbf{u}, (q_k)} J, \quad (5)$$

$$J = \varphi_{n_s}(\mathbf{x}(t_f), t_f) + \sum_{k=1}^{n_s-1} \varphi_k(\mathbf{x}(t_k - 0), \mathbf{x}(t_k + 0)) + \sum_{k=1}^{n_s} \int_{t_{k-1}}^{t_k} L_k(\mathbf{x}(t), \mathbf{u}(t), t) dt,$$

mit reellwertigen Funktionen φ_k, L_k unter den Nebenbedingungen, die aus den Bewegungsgleichungen (2), den Anfangsbedingungen $\mathbf{x}(0) = \mathbf{x}_0$ und der Schaltbedingung (1) sowie Beschränkungen an den Endzustand

$$0 = \mathbf{r}_{\text{eq}, q_{n_s}}(\mathbf{x}(t_f)), \quad 0 \leq \mathbf{r}_{\text{iq}, q_{n_s}}(\mathbf{x}(t_f))$$

und an die Zustands- und Steuervariablen aus Gln. (3), (4),

$$\mathbf{u}_{q_i}^{LB} \leq \mathbf{u}(t) \leq \mathbf{u}_{q_i}^{UB}, \quad \mathbf{x}_{q_i}^{LB} \leq \mathbf{x}(t) \leq \mathbf{x}_{q_i}^{UB},$$

mit konstanten Schranken $\mathbf{u}_{q_i}^{LB}, \dots, \mathbf{x}_{q_i}^{UB}$ bestehen.

Eine mathematische Formulierung der Menge aller zulässigen Sequenzen (q_k) ist nötig, um daraus - z.B. mit einem Branch-and-Bound-Verfahren (B&B) - diejenige Sequenz mit geringsten Kosten J zu finden. Dazu können die durch die Graphenstruktur gegebenen möglichen Abläufe als Relationen, unter Verwendung binärer Variablen $b_q \in \{0, 1\}$ als lineare Gleichungen und Ungleichungen geschrieben werden. Die logische Verknüpfung der Nebenbedingungen kann z.B. durch den Big-M Ansatz [25] oder durch Convex-Hull-Relaxierungen [15] erreicht werden. Welcher Ansatz im speziellen besser geeignet ist hängt u.a. vom Verhalten der Nebenbedingungen bei Relaxierung ab. Hier sind Nebenbedingungen, die sich aus der Convex-Hull-Relaxierungen ergeben, strenger als die aus Big-M [12]. Die Kosten der einzelnen Systemzustände werden analog verknüpft. Die Berücksichtigung spezieller Strukturen des Multigraphen (z.B. Entkoppeln an Brücken) kann darüber hinaus bei der Umwandlung des HA in ein gemischt binäres Optimalsteuerungsproblem (MBOCP) helfen, unnötige Berechnungskomplexität zu vermeiden.

3 Numerische Berechnung

3.1 Direkte Kollokation und Branch-and-Bound

Der hier betrachtete, allgemeine nichtlineare numerische Lösungsansatz besteht aus einer Zerlegung des MBOCP in ein gekoppeltes diskret-dynamisches Optimierungsproblem mit äußerer und innerer Schleife (vgl. [5, 24]).

In der inneren Iteration werden dynamische Optimierungsprobleme betrachtet, bei denen die nichtlineare Zustandsdynamik in mehreren Phasen definiert ist (Abb. 1). Für jede Phase $[t_{k-1}, t_k]$ wird ein adaptives Zeitdiskretisierungsgitter eingeführt. Entlang dieses Zeitgitters werden die kontinuierlichen Zustandsvariablen $\mathbf{x}(t)$ und Steuervariablen $\mathbf{u}(t)$ approximiert [24]. In der vorliegenden Arbeit werden für die Zustandsvariablen stetig differenzierbare stückweise kubische und für die Steuervariablen stetige stückweise lineare Polynome verwendet. Die Kollokationsbedingungen werden an Lobatto Punkten, alle weiteren Nebenbedingungen auf den Punkten des

Diskretisierungsgitters gefordert. Auf diese Weise wird das Optimalsteuerungsproblem mit (teilweise relaxierten) binären Variablen in ein großes, dünn besetztes, nicht-lineares, beschränktes Optimierungsproblem transformiert, welches numerisch mit einem SQP-Verfahren gelöst wird. Ist eine Lösung gefunden, wird in Bereichen mit starker Verletzung der Kollokations- und Nebenbedingungen das Diskretisierungsgitter verfeinert.

In der äußeren Iteration wird eine Suche im diskreten Lösungsraum mit B&B durchgeführt. I.Allg. geht durch eine vorübergehende Relaxierung binärer Variablen an inneren Knoten des Suchbaums die physikalische Bedeutung verloren, was das Verfahren aber nicht einschränkt, sofern eine numerische Lösung existiert.

Deren Effizienz hängt maßgeblich von guten Startschätzungen sowie dem Bereitstellen guter unterer und oberer Schranken der Zielfunktion (5) ab [24]. Um gute Startlösungen zu gewährleisten kommen bisher Homotopieverfahren zum Einsatz, welche die benötigten Lösungen beim Durchlaufen des Suchbaums ineinander überführen.

3.2 Stückweise lineare zeitdiskrete Systeme

Durch enorme Effizienzverbesserungen in den letzten Jahren bei der numerischen Lösung gemischt-ganzzahliger linearer Optimierungsprobleme (MILP) wurden lineare hybride Systeme, insb. stückweise lineare (PWA) Systeme zur approximativen Optimalsteuerung interessant. Zur Wegplanung einfacher kooperativer Roboter- und Fahrzeugsysteme wurden MILP-Formulierungen bereits eingesetzt, z.B. [8]. Diverse Approximationsansätze von Optimalsteuerungsproblemen mit MILP wurden vorgestellt, Bsp. [16, 4, 23]. Zur effizienten Optimierung ist dabei die Diskretisierung und Linearisierung unter enger Berücksichtigung der Optimalitätskriterien zu wählen. So ist etwa eine zeitdiskrete Linearisierung vor allem bei fester Endzeit t_f effizient.

Neben der Effizienzsteigerung von MBOCP-Lösern durch Integration linearisierter Modelle sind MILP-Modelle auch im Blick auf Echtzeitsteuerungen von Bedeutung. Im Rahmen einer modell-prädiktiven Regelung (MPC) können Offline-Lösungen effizient eingesetzt werden [7, 22], so dass in Echtzeit ein lineares Reglergesetz in Abhängigkeit vom aktuellen Zustand stets zur Verfügung steht.

In dem nun vorgestellten Ansatz wird Gl. (2) um (i. Allg. mehrere) Betriebspunkte $(\mathbf{x}^*, \mathbf{u}^*)$ linearisiert

$$\mathbf{f}_q(\mathbf{x}, \mathbf{u}) \approx \left. \frac{\partial \mathbf{f}_q}{\partial \mathbf{x}} \right|_{\mathbf{x}^*, \mathbf{u}^*} (\mathbf{x} - \mathbf{x}^*) + \left. \frac{\partial \mathbf{f}_q}{\partial \mathbf{u}} \right|_{\mathbf{x}^*, \mathbf{u}^*} (\mathbf{u} - \mathbf{u}^*) + \mathbf{f}_q(\mathbf{x}^*, \mathbf{u}^*) \quad (6)$$

Die Anzahl und Verteilung der $(\mathbf{x}^*, \mathbf{u}^*)$ ist dabei in erster Linie anhand des Grads der Nichtlinearität von \mathbf{f}_q und der gewünschten Genauigkeit der Approximation abzuwägen.

Auf festem Zeitgitter und mit einer Euler-Diskretisierung lässt sich Gl. (2) schließlich in folgende Form überführen,

$$\mathbf{x}_{k+1} = \mathbf{A}_q \cdot \mathbf{x}_k + \mathbf{B}_q \cdot \mathbf{u}_k + \mathbf{d}_q.$$

Logische Bedingungen werden durch lineare Ungleichungen eingebracht, die Verknüpfung algebraischer Beschränkungen mit diskreten Variablen über einen Big-M-Ansatz (s.a. [20]). Ferner erlaubt die feste Taktung eine intuitive, direkte Modellierung.

Im Kontext hybrider Automaten tritt jede zusätzliche lineare Systembeschreibung als Knoten q in Erscheinung, zusammen mit zusätzlichen Invarianten, welche die Umgebung von $(\mathbf{x}^*, \mathbf{u}^*)$ als Gültigkeitsbereich der Linearisierung beschreiben. Dies erhöht die diskrete Struktur des Gesamtmodells und bringt die Methode insbesondere bei starken Nichtlinearitäten an ihre Grenzen.

Die Zielfunktion (5) des MBOCP kann ebenfalls durch Linearisierung transformiert werden, oder in eigener Formulierung in das resultierende MILP aufgenommen werden.

4 Repräsentative Benchmarkprobleme und exemplarische Ergebnisse

4.1 Simultane Rollenverteilung und Physik-basierte Trajektorienplanung

Anhand eines Benchmark-Szenarios (vgl. Abb. 3) aus dem Roboterfußball soll die vorgestellte Methodik demonstriert werden: *„Wie können zwei Roboter bei vorhandenem Gegenspieler einen Ball innerhalb einer festen Zeitspanne dem Tor möglichst nahe bringen?“*

Physikalische Bewegungsmöglichkeiten mobiler Roboter unterscheiden sich je nach Bauprinzip signifikant. Optimale taktische Bewegungen können somit nur unter Berücksichtigung der individuellen Bewegungsdynamik erreicht werden. Durch die dynamische Interaktion mit der Umwelt (Ball, Gegenspieler) und durch offensichtliche Vorzüge kooperativen Verhaltens (z.B. Spielen eines Doppelpasses) werden entscheidende charakteristische Merkmale der Mehrroboter-Interaktion hier berücksichtigt.

Aus den Grundfertigkeiten einzelner Fußballroboter $q_R \in Q_R = \{\text{goto pos, catch, dribble, kick}\}$ werden für ein Rollenverhalten verschiedene dieser Aktionen kombiniert. Auch für den Ball werden diskrete Bewegungsmuster unterschieden, z.B. $q_B \in Q_B = \{\text{free, contact, rebound}\}$. Zur Veranschaulichung ziehen wir uns hier auf ein Minimalbeispiel zurück, welches vier grundsätzliche Rollenverteilungen (①, ..., ④) gemäß Abb. 2) vorsieht. Ballannahme und Schüsse werden dabei als Schaltung modelliert; ein Fangen setzt die Ballposition nach dem Schaltvorgang der Roboterposition gleich, während ein Schuss die aktuelle Ballgeschwindigkeit um einen konstanten Faktor erhöht.

I. Allg. enthält Gl. (2) kinematische oder kinetische Fahrzeugmodelle des nötigen Detaillierungsgrads. Zur Ver-

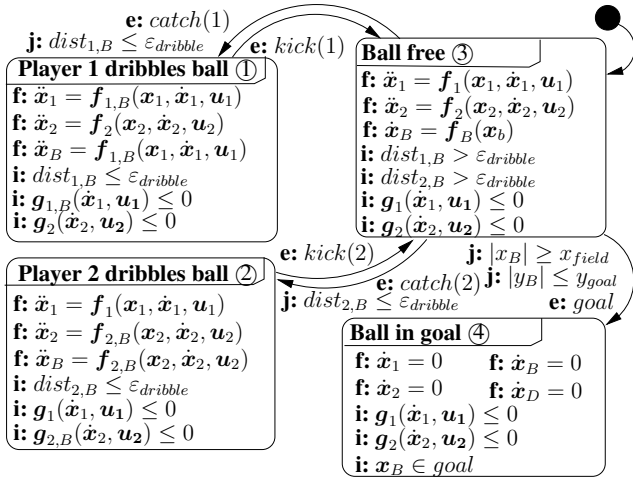


Bild 2: Automatenmodell des Soccer-Benchmarkproblems.

anschaulichung werden hier o.B.d.A. Punktmassenmodelle

$$\dot{x}^\diamond = (\dot{x}^\diamond, \dot{y}^\diamond, \dot{v}_x^\diamond, \dot{v}_y^\diamond)^\top = (v_x^\diamond, v_y^\diamond, u_x^\diamond, u_y^\diamond)^\top \quad (7)$$

$\diamond \in \{1, 2\}$ für die Dynamik der Roboter verwendet. Die Steuerung $u_{x/y}^\diamond$ repräsentiert die Kraft zur Beschleunigung. Die reduzierte Mobilität eines dribbelnden Roboters \diamond ist dabei durch eine Beschränkung an dessen Maximalgeschwindigkeit modelliert. Die Geschwindigkeit des frei rollenden Balles nimmt durch Reibung ab ($0 < \alpha_r < 1$)

$$\dot{x}_B = f_B(x^B) = \begin{pmatrix} v_x^B \\ v_y^B \\ -\alpha_r \\ -\alpha_r \end{pmatrix} \quad \left| \quad \dot{x}_B = f_\diamond(x^\diamond, u^\diamond) \right. \quad \left. \begin{array}{l} \text{Ball rollt frei} \\ \text{Ballführen} \end{array} \right.$$

$$v_x^{\diamond 2} + v_y^{\diamond 2} \leq v_{max}^2 \quad \left| \quad v_x^{\diamond 2} + v_y^{\diamond 2} \leq (0.7 \cdot v_{max})^2 \right.$$

Weiterhin können verschiedene Gegner in eine solche Formulierung eingebracht werden. Ein kontinuierlicher Zustand $x^G = (x^G, y^G, v_x^G, v_y^G)^\top$ wird dazu analog zu x^\diamond definiert. Falls die Bewegung bekannt ist, kann dieser als ein reaktives bewegliches Hindernis über die Formulierung von Zustandsbeschränkungen berücksichtigt werden.

Kollisionsvermeidungsbedingungen (4) der Form

$$d_c - \sqrt{(x^{i_1} - x^{i_2})^2 + (y^{i_1} - y^{i_2})^2} \leq 0 \quad (8)$$

($i_1, i_2 \in \{1, 2, B, G\}$, $i_1 \neq i_2$) werden jeweils in die Knoten $q \in \{\textcircled{1}, \textcircled{2}, \textcircled{3}\}$ aufgenommen.

In [11] wurden bereits Ergebnisse dieses Beispiels mit den Methoden aus 3.1 vorgestellt, so dass wir hier exemplarisch nur ein Resultat der MILP-Modellierung vorstellen.

Auf einem fixierten, äquidistanten Zeitgitter mit $N + 1$ Punkten und Gitterabstand T_s erhalten wir aus obigem Punktmassenmodell die lineare Approximation ($x_k^i := x^i(t_k)$, $i \in \{1, 2, B, G\}$)

$$\begin{aligned} x_{k+1}^i &= x_k^i + T_s \cdot v_{x,k}^i, & v_{x,k+1}^i &= v_{x,k}^i + T_s \cdot u_{x,k}^i \\ y_{k+1}^i &= y_k^i + T_s \cdot v_{y,k}^i, & v_{y,k+1}^i &= v_{y,k}^i + T_s \cdot u_{y,k}^i. \end{aligned} \quad (9)$$

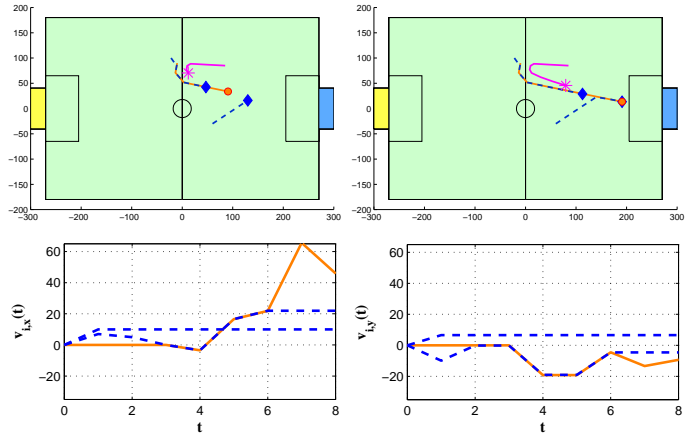


Bild 3: Ergebnis einer MILP-Formulierung: Roboter 1 (---♦) nimmt den Ball (—○) an, passt zu Roboter 2 welcher den Ball zum Tor schießt. Der Gegenspieler (—✱) folgt dem Ball. Unten: Resultierende optimale Geschwindigkeiten.

Um neben dem festen Zeitgitter keine weiteren Strukturvorgaben an die Lösung zu setzen, erhöhen wir die Zahl der Phasen von n_s auf $N + 1$, so dass diskrete Zustände $q(t)$ (bzw. entsprechende binäre Variablen $b^q(t)$) ihren Wert genau zu den Zeitschritten t_k ändern können.

Die reaktive Dynamik des Gegenspielers ist hier so angelegt, dass dessen Geschwindigkeitsvektor stets in Richtung der aktuellen Ballposition weist. Die Kollisionsvermeidung (8) werden jeweils durch ein Bündel von $n_d \geq 4$ linearen Beschränkungen approximiert ($j = 1, \dots, n_d$):

$$\sin\left(\frac{2j}{n_d} \pi\right) (x^{i_1} - x^{i_2}) + \cos\left(\frac{2j}{n_d} \pi\right) (y^{i_1} - y^{i_2}) \leq d_c. \quad (10)$$

Die Frage, wie nahe der Ball dem Tor gebracht werden kann, spiegelt sich in der Zielfunktion wider,

$$-x_{N+1}^B + 0.7 |y_{N+1}^B| + \varepsilon \sum_{i \in \{1, 2\}} \sum_{k=1}^N |u_{x,k}^i| + |u_{y,k}^i|. \quad (11)$$

Der letzte Summand in (11) wurde mit geringer Gewichtung $\varepsilon = 0.01$ aufgenommen, um verbliebene Freiheitsgrade des linearen Problems zu beseitigen. Weitere Details zum MILP-Modell finden sich in [20]. Die Lösung (Abb. 3) wurde für 13 Zeitpunkte in 12.0 sec. berechnet¹.

4.2 Simultane Wegpunktzuweisung und Trajektorienplanung bei Überwachungsmissionen

Folgendes Kernproblem der Überwachung, etwa von Verkehrsknotenpunkten oder von Waldbrand gefährdeten Gebieten, wird untersucht: *Eine Gruppe Luftfahrzeuge startet gemeinsam, um bestimmte Ziele zu überfliegen und schließlich zum Ausgangspunkt zurückzukehren. Wie sehen die zugehörigen optimalen Bewegungstrajektorien aus?*

¹ In allen Beispielen: Intel Pentium-4, 2.66 GHz; 1 GB RAM; MILP-Löser CPLEX [14]; SQP-Verfahren SNOPT [10]

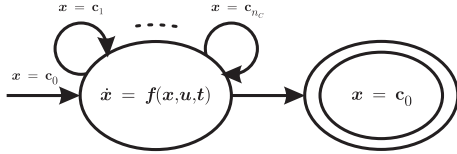


Bild 4: Automatenmodellierung des mmTSP: Nach dem Passieren eines ersten Wegpunktes c_l wechselt der Zustand solange in Zustand $go-to-c_l$ zurück, bis alle Punkte erreicht wurden.

Dieses kombinatorische Optimalsteuerungsproblem ist durch die Kopplung diskreter Entscheidungen und kontinuierlicher Trajektorienoptimierung charakterisiert. Im Gegensatz zum klassischen Handlungsreisendenproblem müssen die Orte hier entlang differenzierbarer Trajektorien überflogen werden. Somit hängen die Kosten eines Wegabschnitts stets von den Reihenfolgen *aller* Vorgänger- und Nachfolgeknoten ab. Das Problem ist als *multiple motorized travelling salesmen problem* (mmTSP) bekannt.

Es wird exemplarisch eine Auswahl von $n_c = 5$ Wegpunkten betrachtet, die jeweils von mindestens einem der $n_v = 2$ kooperierenden Fahrzeuge besucht werden müssen. Die Zahl möglicher Rundreisen beträgt $(n_c + 1)! = 720$, diejenigen eingeschlossen, bei denen einem Fahrzeug alle Orte zugeordnet werden. Ein Automatenmodell kann in einfacher Struktur beschrieben werden (Abb. 4).

Auch hier reduzieren wir die i. Allg. nichtlineare Fahrzeugdynamik zur Demonstration auf ein Modell zweier Punktmassen i (7), die zur Zeit $t_0 = 0$ starten und zur festen Endzeit $t_f = 300$ zum Ausgangspunkt zurückkehren

$$\begin{aligned} x^i(0) = 0 = x^i(t_f), \quad y^i(0) = 0 = y^i(t_f), \\ v_x^i(0) = 0 = v_x^i(t_f), \quad v_y^i(0) = 0 = v_y^i(t_f). \end{aligned} \quad (12)$$

Beschränkungen an Zustands- und Steuergrößen sind als einfache Schranken $|x^i| \leq 70$, $|y^i| \leq 70$, $|v_x^i| \leq 15$, $|v_y^i| \leq 15$, $|u_x^i| \leq 100$, $|u_y^i| \leq 100$ berücksichtigt.

Als Verbindung der Phasen an den Schaltzeitpunkten t_k lauten die Rundreise-Beschränkungen

$$\bigvee_{i=1}^{n_v} \bigvee_{l=1}^{n_c} [x^i(t_k) - c_l = 0].$$

Unter Verwendung der binären Variablen b_k^v und des binärwertigen Vektors $\mathbf{b}_k \in \{0, 1\}^{n_c}$ kann dies für $n_v = 2$ ($k = 1, \dots, n_s - 1$) durch

$$b^v \begin{pmatrix} x^1(t_k) \\ y^1(t_k) \end{pmatrix} + (1 - b^v) \begin{pmatrix} x^2(t_k) \\ y^2(t_k) \end{pmatrix} - \mathbf{C} \mathbf{b}_k = \mathbf{0}, \quad (13)$$

ausgedrückt werden, mit $\mathbf{C} = (\mathbf{c}_1, \dots, \mathbf{c}_{n_c}) \in \mathbb{R}^{2 \times n_c}$, $\mathbf{b}_{k_1}^T \mathbf{b}_{k_2} = 0$ für $k_1 \neq k_2$ und $\mathbf{b}_{k_1}^T \mathbf{b}_{k_1} = 1$. Zahlreiche Symmetrien (n_v gleiche Fahrzeuge, Reihenfolge vorwärts und rückwärts...) führen zu mehreren gleichwertigen Optima. Zum Testen der allgemeinen Methodik verzichten wir hier bewusst darauf, diese problemspezifische Charakteristik zur Effizienzsteigerung zu entfernen.

4.2.1 Lösung mit B&B und direkter Kollokation

Eine Lösung des MBOCP mit der Methodik aus Abschnitt 3.1 ist in Abb. 5 dargestellt. Mit einer Rechenzeit von

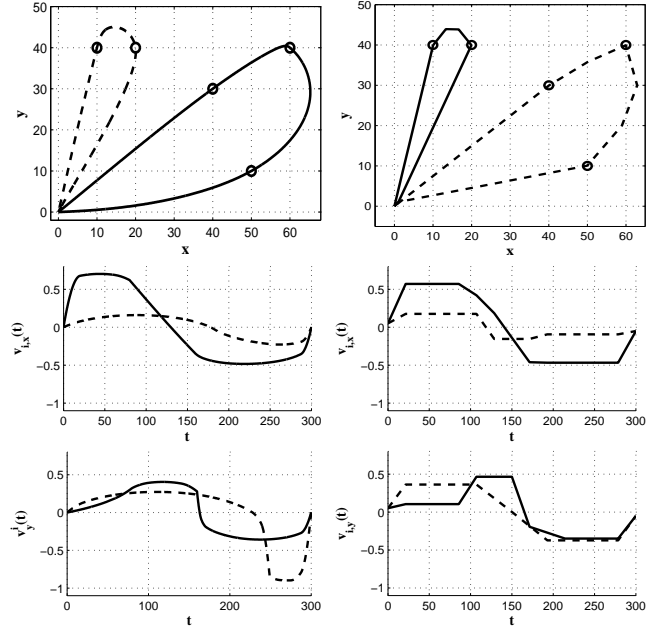


Bild 5: Optimale Pfade und Geschwindigkeitsverläufe. Der Vergleich der Lösungen des Originalproblem (links) und des MILP-Modells (rechts) zeigt den approximativen Charakter.

17 min. wurde bei sehr groben Startwerten für

$$\min \int_0^{300} \sum_{i=1}^2 \sqrt{u_x^i(t)^2 + \varepsilon} + \sqrt{u_y^i(t)^2 + \varepsilon} dt \quad (14)$$

($\varepsilon = 0.1$) ein Wert von 7.4605 errechnet. Es ist $\sqrt{u^2 + \varepsilon}$ eine differenzierbare Näherung der Betragsfunktion.

In einem weiteren Test konnte (14) mit numerisch günstigerem Integranden $\sum u_x^i{}^2 + u_y^i{}^2$ in 9 min. zu 12.7086 minimiert werden. Für die Auswertung einer vorgegebenen Aufteilung und Reihenfolge dauert die Lösung des (rein kontinuierlichen) Mehrphasen-Optimalsteuerungsproblems etwa 3.3 sec., so dass sich das Gesamtkonzept in Anbetracht des unterlagerten kombinatorischen Charakters effizient zeigt.

Da das SQP-Verfahren der inneren Schleife nur eine lokale Optimalität garantieren kann, ist auch insgesamt die globale Optimalität nicht gesichert.

4.2.2 Lösung der approximativen MILP-Modellierung

Das MILP-Modell basiert auf einem Zeitgitter mit $N + 1 = 15$ äquidistant verteilten Punkten $t_0 = 0, \dots, t_{N+1} := 300$ ($T_s = 300/14$) und einfacher Dynamik (9).

Die Schranken an Zustands- und Stellvariablen werden analog zu 4.2.1 gewählt. Mittels *Big-M*-Ansatz übersetzen

sich die Rundreisebedingungen zu ($M = 140$)

$$\begin{aligned}
& \forall c_j, \forall k, \forall i : | -x_k^i - c_{j,x} | + b_j^i(t_k) \cdot M \leq \varepsilon + M \\
& \forall c_j, \forall k, \forall i : | -y_k^i - c_{j,y} | + b_j^i(t_k) \cdot M \leq \varepsilon + M \\
& \forall j : \sum_{k=1}^{N+1} \sum_{i=1}^{n_v} b_j^i(t_k) = 1 \\
& \sum_{j=1}^{n_c} \sum_{k=1}^{N+1} \sum_{i=1}^{n_v} b_j^i(t_k) = n_c .
\end{aligned} \tag{15}$$

Als eine zu (14) vergleichbare Zielfunktion setzen wir

$$T_s \cdot \sum_{i=1}^{n_v} \sum_{k=1}^N (|u_{x,k}^i| + |u_{y,k}^i| + 0.1 \cdot (|\Delta_{u_{x,k}^i}| + |\Delta_{u_{y,k}^i}|)) , \tag{16}$$

wobei Δ_{u^i} die Änderung von u^i pro Zeitschritt beschreibt.

Ohne Angabe von Startwerten ergab sich mit einer Rechenzeit von 19s die Lösung in Abb. 5 mit einem Zielfunktionswert von 7.2005. Der approximative Charakter der Lösung ist hier deutlich erkennbar, was auch weitere Beispielrechnungen bestätigten. Da die Lösung eines MILP stets dessen globales Minimum darstellt, zeigt sie sich als gute Ausgangsbasis zur Kombination mit den Ansätzen aus 3.1.

4.2.3 Erweiterung: Unterlagerte Graphenstruktur

Um die Leistungsfähigkeit von MILP-Modellen weiter zu untermauern, wird die Problemstellung abschließend um die praxisrelevante Bedingung der Einhaltung von Formationstopologien erweitert. Die Notwendigkeit, Formationen zu bilden, kann durch die kooperative Aufgabe selbst oder indirekt, z.B. über Funk-Konnektivitätsforderungen, gegeben sein. Die Frage nach optimalen Sequenzen von Netzwerktopologien und Formationen fällt unmittelbar in den Bereich der vorgestellten Methoden. Weiterführende Literatur dazu findet sich in [21].

Für jedes Fahrzeug i_1 der Problemstellung dieses Kapitels 4.2 betrachten wir eine distanzabhängige Nebenbedingung, welche die eigene Position mit der Position des Fahrzeugs i_2 in Beziehung setzt. Dabei beschränken wir uns hier auf den symmetrischen Fall und definieren ($i_1 \neq i_2$)

$$[a_{i_1, i_2} = -1] : \Leftrightarrow \begin{cases} C_{i_1}(\mathbf{x}^{i_1}, \mathbf{x}^{i_2}) \leq 0 \\ \vee C_{i_2}(\mathbf{x}^{i_2}, \mathbf{x}^{i_1}) \leq 0 \end{cases} \tag{17}$$

für den Fall, dass eine Nachbarschaftsbeziehung zwischen i_1 und i_2 vorliegt. Für $i_1 = i_2$ setzen wir

$$a_{i_1, i_1} = - \sum_{\substack{i_2=1 \\ i_2 \neq i_1}}^{n_v} a_{i_1, i_2} .$$

Die Eigenwerte der quadratischen Matrix $\mathcal{A} := (a_{i_1, i_2})_{i_1, i_2 \in \mathbb{Z}^{n_v \times n_v}}$ („Laplace-Matrix“) lassen Aussagen über die Qualität des Graphen zu. Insbesondere

$$[\lambda_2 > 0 \Leftrightarrow \text{Graph ist zusammenhängend}] \tag{18}$$

ist hier von Bedeutung, wobei λ_2 der zweitgrößte Eigenwert („algebraische Konnektivität“) von \mathcal{A} ist [9].

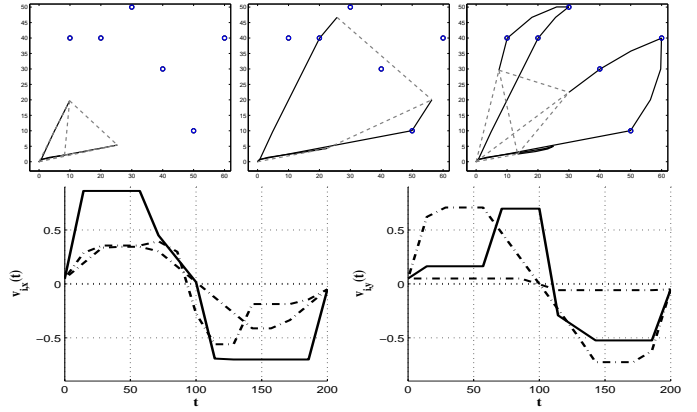


Bild 6: Ergebnisse einer MILP-Formulierung: Optimale Wege Geschwindigkeitsverläufe der Fahrzeuge mit sich verändernder Graphenstruktur. Eines der Fahrzeuge verharrt im Ursprung.

Mit $\mathcal{A}(t) \in \mathcal{T}$ betrachten wir \mathcal{A} als Element einer endlichen Menge zulässiger Graphentopologien \mathcal{T} , welche mit Hilfe logischer Bedingungen beschrieben werden kann. Um eine geeignete Steuerung der Topologie zu gewährleisten, fordern wir, dass Wechsel in der Netzwerkstruktur nur an definierten diskreten Schaltzeitpunkten t_k stattfinden können, $\mathcal{A}^k := \mathcal{A}(t)$, $t_{k-1} \leq t \leq t_k$. In der Systemmodellierung mit HA bildet jedes der Elemente aus \mathcal{T} einen Knoten.

Unter Berücksichtigung aller Aspekte des Abschnitts 4.2 zeigt Abb. 6 exemplarisch ein Resultat der MILP-Modellierung. Das mmTSP wurde dazu auf $n_c = 6$ Wegpunkte und ein heterogenes System von $n_v = 4$ Fahrzeugen erweitert. Durch distanzabhängige Nachbarschaftsbeziehungen spannen diese stets einen zusammenhängenden Graphen auf, was durch eine lineare Formulierung von (18)

$$- \sum_{i_1=1}^{n_v} \sum_{i_2 > i_1}^{n_v} a_{i_1, i_2}(t_k) \leq 1, \sum_{i_1=1}^{n_v} \sum_{\substack{i_2=1 \\ i_2 \neq i_1}}^{n_v} a_{i_1, i_2}(t_k) \geq -3$$

in das MILP-Modell einging. Unterschiedliche Fahrodynamik wurde durch die Schranken $t_f = 200$, $|v_{x/y}^2| \leq 0.75$, $|v_{x/y}^3| \leq 0.75$, $|v_{x/y}^4| \leq 0.5$, $|u_{x/y}^2| \leq 0.4$, $|u_{x/y}^3| \leq 0.4$ und $|u_{x/y}^4| \leq 0.1$ charakterisiert.

Analog zu (10) (mit $n_d = 4$, $dist = 35$) und mit anschließender Big-M-Formulierung lässt sich (17) für

$$C_{i_1}(\mathbf{x}^{i_1}, \mathbf{x}^{i_2}) = \sqrt{(x^{i_1} - x^{i_2})^2 + (y^{i_1} - y^{i_2})^2} - dist$$

in lineare Nebenbedingungen umsetzen. Das Ergebnis (Abb. 6, Rechenzeit etwa 66 sec.) lässt den Einfluss auf die optimale Bewegung des Gesamtsystems erkennen.

5 Diskussion und Ausblick

Eine geschlossene Methode der Modellierung und Lösung hybrider Optimalsteuerungsprobleme für kooperative Mehrfahrzeugsysteme wurde mit zwei leistungsfähigen

numerischen Ansätzen vorgestellt und exemplarisch auf repräsentative Probleme angewendet.

Die Modellierung erfolgt dabei unter Verwendung hybrider Automaten, welche die enge Kopplung von diskreten und kontinuierlichen Systemzuständen geeignet abbilden. Das Automatenmodell wird schließlich zur Lösung in gemischt-ganzzahlige Optimierungsprobleme übersetzt.

Der Kollokationsansatz erlaubt es dabei, hybride Bewegungsdynamikmodelle mit hoher Nichtlinearität zu lösen. Die Performanz der numerischen Gesamtmethode hängt maßgeblich von Startschätzungen und Schranken für die Teilprobleme der inneren Schleife ab. I. Allg. kann das SQP-Verfahren dort nur lokale Optimalität garantieren.

Die Lösung zeitdiskreter MILP-Modelle hingegen erweist sich ohne Vorgabe von Startwerten als effizient und liefert ein garantiert globales Optimum. Sofern ein Übergang von nichtlinearer Systembeschreibung auf PWA-Modelle mit zufriedenstellender Genauigkeit bei moderater Zahl von Betriebspunktlinearisierungen erreicht werden kann, bietet sich die Methode somit zur Generierung von Startschätzungen an und kann damit neben der Effizienz des Kollokationsansatzes auch dessen Wahrscheinlichkeit erhöhen, ein tatsächlich globales Minimum zu berechnen. Insbesondere bei Nichtkonvexitäten im Modell, etwa durch Hindernisbeschränkungen, ist dieser Vorteil offensichtlich.

In weitergehenden Arbeiten werden konkrete Kombinationen der Ansätze auf verschiedenen Ebenen der numerischen Berechnung untersucht, z.B. der Einsatz von MILP-Lösungen als Startschätzungen nichtlinearer Teilprobleme oder zur Verbesserung der Suchvorschrift im B&B-Baum.

Die Lösbarkeit zunehmend realitätsnaher Problemstellungen stellt den mittelfristigen Einsatz der vorgestellten Verfahren in realen Systemen in Aussicht. Weiterführende Fragen der Dezentralisierung, der Übertragbarkeit von MILP-Modellen auf MPC-Regler und damit verbundene Fragen der Robustheit werden dazu untersucht.

Die Erweiterungen der Modellierung auf *hierarchische* hybride Automaten, die Nebenläufigkeit und Synchronisation beschreiben, sind ebenfalls Gegenstand aktueller Untersuchungen. Hierbei verspricht die Kombination von Methoden der Systemanalyse/-verifikation mit Optimierungsverfahren eine weitere Verbesserung der Algorithmen und es eröffnen sich neue Möglichkeiten der dynamischen mehrschichtigen Verwaltung von Zustandsbeschränkungen.

Danksagung

Teile dieser Arbeit wurden durch die Deutsche Forschungsgemeinschaft (DFG) im Rahmen des GRK 1362 „Cooperative, adaptive and responsive monitoring in mixed mode environments“ gefördert.

Literatur

- [1] A. Bemporad and N. Giorgetti. Logic-based methods for optimal control of hybrid systems. *IEEE Trans. Automatic Control*, 51:963–976, 2006.

- [2] A. Bemporad and M. Morari. Control of systems integrating logic, dynamics, and constraints. *Automatica*, 35(3):407–427, 1999.
- [3] J. Borges de Sousa, K. H. Johansson, J. Silva, and A. Speranzon. A verified hierarchical control architecture for coordinated multi-vehicle operations. *International Journal of Adaptive Control and Signal Processing*, 21(2–3):159–188, 2007. Special issue on autonomous adaptive control of vehicles.
- [4] F. Borrelli, P. Falcone, and C. D. Vecchio. Event-based receding horizon control for two-stages multi-product production plants. In *American Control Conference, June 14-16, 2006, Minneapolis, Minnesota, USA*, 2006.
- [5] M. Buss, M. Glocker, M. Hardt, O. von Stryk, R. Bulirsch, and G. Schmidt. Nonlinear hybrid dynamical systems: modeling, optimal control, and applications. In E. S. S. Engell, G. Frehse, editor, *Modelling, Analysis and Design of Hybrid Systems*, volume 279 of *Lecture Notes in Control and Information Sciences*, pages 311–335, Berlin, 2002. Springer.
- [6] L. Chaimowicz, V. Kumar, and M. Campos. A paradigm for dynamic coordination of multiple robots. *Autonomous Robots*, 17(1):7–21, 2004.
- [7] F. J. Christophersen. *Optimal Control and Analysis for Constrained Piecewise Affine Systems*. Dr. sc. ETH Zurich thesis, ETH Zurich, Zurich, Switzerland, Aug. 2006.
- [8] M. G. Earl and R. D’Andrea. Iterative milp methods for vehicle control problems. In *IEEE Conference on Decision and Control*, volume 4, pages 4369 – 4374, December 2004.
- [9] M. Fiedler. Algebraic connectivity of graphs. *Czech. Math. J.*, 23:298–305, 1973.
- [10] P. Gill, W. Murray, and M. Saunders. SNOPT: An SQP algorithm for large-scale constrained optimization. *SIAM J. Optim.*, 12:979–1006, 2002.
- [11] M. Glocker, C. Reinl, and O. von Stryk. Optimal task allocation and dynamic trajectory planning for multi-vehicle systems using nonlinear hybrid optimal control. In *Proc. 1st IFAC-Symposium on Multivehicle Systems*, pages 38–43, Salvador, Brazil, October 2-3 2006.
- [12] I. Grossmann and S. Lee. Generalized convex disjunctive programming: Nonlinear convex hull relaxation. *Computational Optimization and Applications*, 26(1):83–100, 2003.
- [13] T. Henzinger. The theory of hybrid automata. In *Proceedings of the 11th Annual Symposium on Logic in Computer Science*, pages 278–292, New Brunswick, NJ, 1996. IEEE Computer Society Press.
- [14] ILOG. *ILOG CPLEX 11.0, User’s Manual*, 2007.
- [15] S. Lee and I. E. Grossmann. New algorithms for nonlinear generalized disjunctive programming. *Computers and Chemical Engineering Journal*, 24:2125–2141, 2000.
- [16] H. Mehne, M. H. Farahi, and A. V. Kamyad. Milp modelling for the time optimal control problem in the case of multiple targets. *Optimal Control Applications and Methods*, 27(2):77 – 91, September 2006.
- [17] R. M. Murray. Recent research in cooperative control of multi-vehicle systems. *Journal of Dynamics, Systems, Measurement and Control*, 129:571 – 583, 2007.
- [18] S. Panek, O. Stursberg, and S. Engell. Optimization of timed automata models using mixed-integer programming. *Formal Modeling and Analysis of Timed Systems*, pages 73–87, 2004.
- [19] L. Parker. Distributed intelligence: Overview of the field and its application in multi-robot systems. *Journal of Physical Agents*, 2(2):5–14, 2008.
- [20] C. Reinl and O. von Stryk. Optimal control of cooperative multi-robot systems using mixed-integer linear programming. In C. I. de Mathematika, editor, *Proc. RoboMat 2007*, pages 145 – 151, Coimbra, Portugal, Sept. 17-19 2007.
- [21] C. Reinl and O. von Stryk. Optimal control of multi-vehicle systems under communication constraints using mixed-integer linear programming. In *Proc. 1st Int. Conf. Robot Communication and Coordination*, Athens, Greece, Oct. 15-17 2007.
- [22] A. Richards and J. How. Mixed-integer programming for con-

trol. In *Proceedings of the American Control Conference*, pages 2676–2683 vol. 4, June 2005.

- [23] O. Stursberg and S. Engell. Optimal control of switched continuous systems using mixed-integer programming. In *15th IFAC World Congress, Barcelona, 2002*.
- [24] O. von Stryk and M. Glocker. Numerical mixed-integer optimal control and motorized traveling salesmen problems. *APII-JESA (Journal européen des systèmes automatisés - European Journal of Control)*, 35(4):519–533, 2001.
- [25] H. P. Williams and S. C. Brailsford. *Advances in Linear and Integer Programming*, chapter Computational logic and integer programming, pages 249–281. Oxford University Press, Inc., New York, NY, USA, 1996.
- [26] S. Zelinski, T. J. Koo, and S. Sastry. Hybrid system design for formations of autonomous vehicles. In *Proc. 42nd IEEE Conf. Decision and Control*, volume 1, pages 1–6, 2003.

Manuskripteingang: ?. Oktober 2008.



Dipl.-Math. Christian Reinl ist Doktorand am Fachgebiet Simulation, Systemoptimierung und Robotik der TU Darmstadt mit Schwerpunkt Optimierung und optimale Steuerung kooperativer Systeme.

Adresse: Technische Universität Darmstadt; FB Informatik; 64289 Darmstadt, Tel: + 49-(0)6151-16-5212, E-Mail: reinl@sim.tu-darmstadt.de



Dr.-Ing. Markus Glocker hat 2006 an der TU Darmstadt über Methoden der nichtlinearen gemischt-ganzzahligen Optimalsteuerung promoviert und ist seither bei Trimble Terrasat im Bereich Algorithmik tätig.

Adresse: Trimble Terrasat GmbH, Haringstr. 19, 85635 Höhenkirchen; Tel: +49-(0)81802-274330, E-Mail: markus.glocker@trimble.com



Prof. Dr. Oskar von Stryk leitet das Fachgebiet Simulation, Systemoptimierung und Robotik am Fachbereich Informatik der TU Darmstadt; Hauptarbeitsgebiete: Optimierung und optimale Steuerung, Roboter- und Fahrzeugsystemdynamik, autonome laufende und biologisch inspirierte Roboter.

Adresse: Technische Universität Darmstadt; FB Informatik; 64289 Darmstadt, Tel: + 49-(0)6151-16-2513, E-Mail: stryk@sim.tu-darmstadt.de