1 Einleitung

1.1 Synchronisation von Multiagentensystemen

Diese Dissertation beschäftigt sich mit *Multiagentensystemen*, welche aus mehreren Teilsystemen (*Agenten*) mit einer gemeinsamen Regelungsaufgabe bestehen (Abb. 1.1). Es werden Methoden zum Entwurf *vernetzter Regler* erarbeitet, welche die Teilsysteme asymptotisch einer gemeinsamen Trajektorie folgen lassen. Die Bewegung der Teilsysteme auf einer gemeinsamen Trajektorie $y_{\mathrm{s}}(t)$

$$\lim_{t\to\infty} |y_i(t) - y_{\mathrm{s}}(t)| = 0, \quad i = 1, 2, \ldots, N$$

wird Synchronisation genannt. Diese Arbeit untersucht, unter welchen Bedingungen ein linearer vernetzter Regler lineare bzw. affine Agenten P_i auch dann synchronisieren kann, wenn die Agenten gestört sind oder Modellunbestimmtheiten aufweisen.

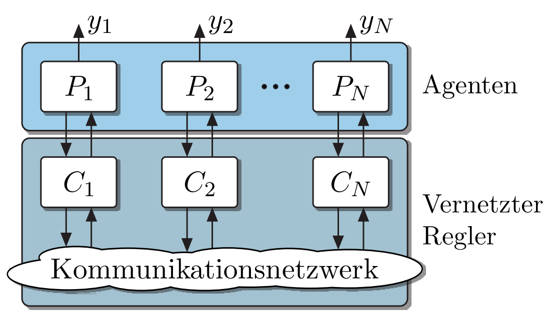

Abb. 1.1: Multiagentensystem

Multiagentensysteme. Abbildung 1.1 zeigt die Struktur von Multiagentensystemen bestehend aus einem Kommunikationsnetzwerk, *lokalen Reglern* C_i, ($i = 1, 2, \ldots, N$) und den Agenten P_i, welche die physikalischen Systeme widerspiegeln. Das Kommunikationsnetzwerk und die lokalen Regler ergeben zusammen den vernetzten Regler. Ein System als Multiagentensystem zu betrachten ist sinnvoll, wenn viele Teilsysteme, die nicht oder nur schwach physikalisch gekoppelt sind, eine gemeinsame Regelungsaufgabe erfüllen.

Anwendungsbeispiele. Die folgenden zwei Anwendungsbeispiele zeigen besondere praktische Randbedingungen von Multiagentensystemen. Bei Fahrzeugkolonnen kann das Kommunikationsnetzwerk über visuelle Grenzen hinweg genutzt werden, um den Verkehrsfluss zu verbessern und Kollisionen zu vermeiden (Abb. 1.2). Aufgrund einer hohen Anzahl an Fahrzeugen und dem Fortbewegen der Kolonne muss eine gemeinsame Regelung aller Fahrzeuge vernetzt erfolgen, aber im Allgemeinen ohne einen Koordinator auskommen. Die breite Variation unterschiedlicher Fahrzeuge und äußerer Einflüsse, wie z.B. Wind und Steigungen, müssen berücksichtigt werden. Auch wenn Fahrzeuge unterschiedliche Zielgeschwindigkeiten haben, sollen sie temporär eine Kolonne bilden, um Energie zu sparen.

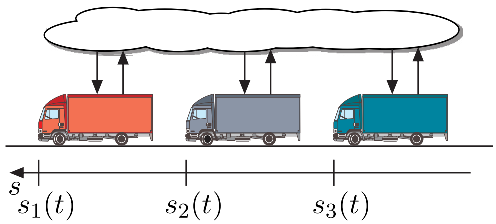

Abb. 1.2: Fahrzeugkolonne

Das Bewässerungssystem in Abb. 1.3 mit mehreren Fahrgestellen bewegt ein Rohrsystem über ein Feld. Dabei ist es wichtig, dass die Fahrgestelle trotz unterschiedlicher Bodengegebenheiten möglichst synchron nebeneinander fahren, um Beschädigungen am Rohrsystem zu verhindern. Es reicht nicht aus, die Fahrgestellmotoren nur zeitgleich ein- und auszuschalten, sondern es ist ein vernetzter Geschwindigkeitsregler erforderlich.

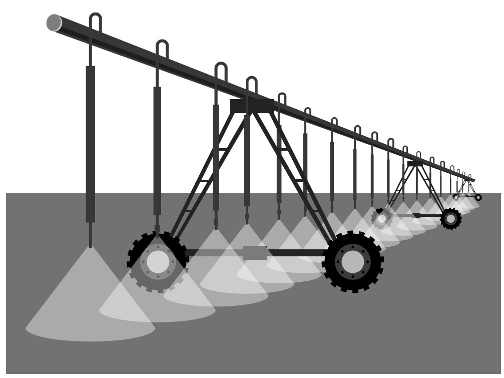

Abb. 1.3: Bewässerungssystem

Die beiden Anwendungsbeispiele zeigen die folgenden **praktischen Randbedingungen** auf, unter denen Synchronisationsaufgaben gelöst werden müssen:

- In der Regel müssen viele Teilsysteme miteinander kooperieren.

- Multiagentensysteme haben typischerweise keinen Koordinator, der alle Agenten gemeinsam steuert bzw. deren Regelungen koordiniert.
- Die Synchronisationsaufgabe muss auch unter Störungen erfüllt sein.
- Modellunsicherheiten in den Agentenmodellen oder dem vernetzten Regler müssen berücksichtigt werden.
- Während des Übergangsverhaltens müssen auch Forderungen erfüllt sein, um z.B. Schäden zu vermeiden.

1.2 Synchronisationsprobleme

Die *asymptotische Synchronisation* ist eine typische Aufgabe für Multiagentensysteme. Dabei sollen die Ausgänge $y_i(t)$, $(i = 1, 2, \ldots, N)$ der Agenten asymptotisch einer gemeinsamen Trajektorie $y_{\mathrm{s}}(t)$ folgen, welche auch als *synchrone Trajektorie* bezeichnet wird.

Definition 1.1 (Asymptotische Synchronisation). Ein Multiagentensystem heißt asymptotisch synchronisiert, wenn es für beliebige Anfangszustände $\boldsymbol{x}_{i0}$, $(i = 1, 2, \ldots, N)$ aller Agenten eine Funktion $y_{\mathrm{s}}(t)$ gibt, mit der die Beziehung

$$\lim_{t \to \infty} |y_i(t) - y_{\mathrm{s}}(t)| = 0, \quad i = 1, 2, \ldots, N \tag{1.1}$$

gilt.

Die angestrebte synchrone Trajektorie $y_{\mathrm{s}}(t)$ ist keine von außen vorgegebene Trajektorie, sondern ergibt sich durch die kooperierenden Agenten und hängt von den Anfangszuständen $\boldsymbol{x}_{i0}$ ab. Dabei darf die synchrone Trajektorie nicht asymptotisch verschwinden

$$y_{\mathrm{s}}(t) \overset{t \to \infty}{\not\longrightarrow} 0.$$

Andernfalls tritt die *triviale Synchronisation* auf, bei der alle Ausgänge $y_i(t)$ asymptotisch verschwinden, wie es bei jedem asymptotisch stabilen System der Fall ist.

Robuste Synchronisation. Haben die Agenten oder der vernetzte Regler Parameterunsicherheiten, wird die *robuste Synchronisation* betrachtet. Die Unsicherheiten treten als Abweichungen von nominellen Parametern $\hat{\boldsymbol{A}}$ auf und können beispielsweise durch eine Norm abgeschätzt werden:

$$\boldsymbol{A} = \hat{\boldsymbol{A}} + \Delta \boldsymbol{A}, \quad ||\Delta \boldsymbol{A}|| \leq \delta, \quad \delta > 0.$$

Der Bezug zu den Agenten und dem vernetzten Regler wird in Kapitel 4 beschrieben.

Definition 1.2 (Robuste Synchronisation). Ein Multiagentensystem heißt robust synchronisiert, wenn das Ziel der asymptotischen Synchronisation (1.1) auch für normbeschränkte Parameterunsicherheiten mit $\delta > 0$ erfüllt ist.

Die Idee der robusten Synchronisation ist es, die für die Agenten bekannten nominellen Modelle weiterhin zu nutzen, um einen vernetzten Regler für die asymptotische Synchronisation zu entwerfen. Da jedoch die Modelle parameterunsicher sind, soll der erhaltene vernetzte Regler ebenfalls das parameterunsichere Multiagentensystem asymptotisch synchronisieren. Gesuchte Synchronisationsbedingungen für parameterunsichere Multiagentensysteme müssen demnach die robuste Synchronisation und die darin inbegriffene asymptotische Synchronisation des Nominalsystems gewährleisten.

Praktische Synchronisation. Kann keine robuste Synchronisation erzielt werden, wird die *praktische Synchronisation* betrachtet. Bei dieser folgen die Agenten der synchronen Trajektorie $y_{\mathrm{s}}(t)$ mit einer Toleranz $\bar{\varepsilon}$, statt ihr exakt zu folgen.

Definition 1.3 (Praktische Synchronisation). Ein Multiagentensystem heißt praktisch synchronisiert mit der Toleranz $\bar{\varepsilon} \geq 0$, wenn es für beliebige Anfangszustände $\boldsymbol{x}_{i0}$, $(i = 1, 2, \ldots, N)$ aller Agenten eine Funktion $y_{\mathrm{s}}(t)$ gibt, mit der die Beziehung

$$\lim_{t \to \infty} |y_i(t) - y_{\mathrm{s}}(t)| \leq \bar{\varepsilon}, \quad i = 1, 2, \ldots, N \tag{1.2}$$

gilt.

Die praktische Synchronisation erlaubt einen bleibenden Synchronisationsfehler zwischen den Ausgängen $y_i(t)$ und der synchronen Trajektorie $y_{\mathrm{s}}(t)$. Dabei kann die praktische Synchronisation als Verallgemeinerung der asymptotischen Synchronisation (1.2) interpretiert werden. Für $\bar{\varepsilon} = 0$ ist die praktische Synchronisation (1.2) identisch mit der asymptotischen Synchronisation (1.1).

Synchronisationsziel. Welches dieser definierten Synchronisationsziele erfüllt werden kann, hängt von den betrachteten Agenten und dem vernetzten Regler ab. Die Synchronisationsziele können den Kapiteln dieser Dissertation wie folgt zugeordnet werden:

- Asymptotische Synchronisation in Kapitel 3, 5 und 6
- Robuste Synchronisation in Kapitel 4
- Praktische Synchronisation in Kapitel 5.

Jedes dieser Synchronisationsziele soll nur unter Verwendung der Agentenausgänge $y_i(t)$ als Reglereingänge erzielt werden. Dabei werden die Agenten über den vernetzten Regler *diffusiv gekoppelt*:

$$u_i(t) = \sum_{j=1}^{N} a_{\mathcal{G}ij}(y_i(t) - y_j(t)), \quad i = 1, 2, \dots, N.$$

Das heißt, die Stellgröße $u_i(t)$ hängt von den Ausgangsdifferenzen $y_i(t) - y_j(t)$ statt von einzelnen Ausgängen ab, was für das Synchronisationsziel (1.1) die Konsequenz

$$\lim_{t \to \infty} u_i(t) = 0, \quad i = 1, 2, \dots, N$$

impliziert, d.h. asymptotisch synchronisierte Agenten folgen der synchronen Trajektorie $y_s(t)$ als Eigenbewegung.

1.3 Literaturübersicht

Bereits im Jahr 1657 hat sich Christiaan Huygens als einer der ersten Wissenschaftler mit der Synchronisation beschäftigt. Er beobachtete, dass sich zwei am selben Holzbalken angebrachte Uhren synchronisieren [7]. Das gezielte Untersuchen des Synchronisationsproblems hat jedoch erst in den letzten Jahrzehnten an Interesse gewonnen, wie die Übersichtsarbeiten [12, 30, 46, 65] zeigen. Dies ist unter anderem auf die Digitalisierung und der damit steigenden Möglichkeit, Systeme miteinander zu vernetzen, zurückzuführen.

Früheren Veröffentlichungen im Bereich der vernetzten Systeme fokussieren sich auf das Konsensproblem, bei dem Integratorsysteme einen gemeinsamen Wert annehmen sollen [19, 58, 63]. Eine der Hauptfragen ist, welche Bedingungen das Kommunikationsnetzwerk erfüllen muss, um Konsens für alle Integratoren zu erreichen. Einen Überblick über das Konsensproblem liefern die Arbeiten [46, 57, 64].

Die Verallgemeinerung des Konsensproblems auf beliebige Systemdynamiken führt auf das Synchronisationsproblem, bei dem die Agentenausgänge einer gemeinsamen Trajektorie folgen sollen. Die früheren Arbeiten [56, 59, 71, 83] konzentrieren sich auf die asymptotische Synchronisation nichtlinearer oder chaotischer Oszillatoren. Später wurden für beliebige Agentendynamiken grundlegende Fragen nach der Synchronisierbarkeit und Synchronisationsbedingung untersucht [13, 14, 38, 39, 48, 81]. Die asymptotische Synchronisation wurde zunächst für identische Agenten betrachtet [34, 54, 67, 77]. Die Erweiterung auf individuelle Agenten zeigt die Notwendigkeit, dass alle Agenten eine gemeinsame Systemdynamik für die Synchronisation benötigen [40, 43, 82]. Eine weitreichende Übersicht über die Synchronisation identischer und individueller Agenten liefert [46]. Dort und in den zuvor angegebenen Referenzen wird außerdem untersucht, wie die synchrone Trajektorie der Agenten aussieht. In Leader-Follower-Systemen ist diese durch den Führungsagenten beschrieben [35, 55, 85], bei allen anderen Strukturen entsteht sie als Linearkombination der Agentenausgänge.