

# Ein Beitrag zur Bewertung der Leistung dezentraler Materialflusssteuerungssysteme am Beispiel eines Steuerungsprototyps

DIPL.-INFORM. TOBIAS R. MAYER

LEHRSTUHL FÜR VERTEILTE INFORMATIONSSYSTEME, UNIVERSITÄT PASSAU

DIPL.-ING. SERGEY LIBERT

LEHRSTUHL FÜR FÖRDER- UND LAGERWESEN, TECHNISCHE UNIVERSITÄT DORTMUND

PROF. DR. MICHAEL TEN HOMPEL

LEHRSTUHL FÜR FÖRDER- UND LAGERWESEN, TECHNISCHE UNIVERSITÄT DORTMUND

**Das Steuerungsparadigma Internet der Dinge in der Intralogistik propagiert hoch modulare und flexible dezentrale Steuerungssysteme auf Basis von RFID-Technologie und Softwareagenten. Ein produktiver Einsatz solcher Systeme verlangt die Erfüllung von Anforderungen, die für intralogistische Anlagen und Systeme in der industriellen Praxis vorliegen. Eine Untersuchung der Echtzeitanforderung in dezentralen Steuerungssystemen wird im Rahmen des Sonderforschungsbereiches (SFB) 696 „Logistics on demand“ (Teilprojekt C5 – „Realtime Logistics“) durchgeführt. Eine systematische Leistungsuntersuchung bestehender und neuer Steuerungen ist ein wichtiger Aspekt dieser Forschungsarbeit. In dem vorliegenden Beitrag wird ein Ansatz zur Bewertung der Leistung einer dezentralen Steuerung vorgestellt. Die Aufnahme und Analyse von Leistungsdaten wird durch den Einsatz eines selbst entwickelten Messinstrumentes unterstützt.**

## 1. Einleitung

Die angewandte Forschung im Bereich automatisierter Materialflusssysteme weist in der letzten Zeit eine starke Neigung zur Entwicklung dezentraler Steuerungskonzepte auf (vgl. [Win08], [HLS06], [Wil06]). Unter den erwarteten Systemvorteilen werden besonders häufig bessere Skalierbarkeit und Anpassungsfähigkeit sowie die Wiederverwendbarkeit von Systemkomponenten genannt (vgl. [Lor03], [BSc01]). Eine Erhöhung der Wirtschaftlichkeit soll dabei durch kürzere Inbetriebnahmezeiten und aufwandarme Erweiterbarkeit der Steuerung garantiert werden (vgl. [GCK08]). Zwischen der Vision einer perfekten dezentralen Materialflusssteuerung und der Umsetzung dieser Vision in der industriellen Praxis besteht jedoch eine Lücke. Der in diesem Artikel vorgeschlagene Einsatz zur Leistungsbewertung dezentraler Materialflusssteuerungssysteme trägt zum Schließen dieser Lücke bei.

### 1.1. Internet der Dinge in der Intralogistik

Nach dem Grundgedanken des Steuerungsparadigmas *Internet der Dinge in der Intralogistik* (vgl. [GtH10]) übernimmt das zu transportierende Gut selbst die steuernde Rolle und nutzt die Transportdienste sowie weitere Dienste einer Förderanlage. Dabei wird auf moderne Technologien wie Radiofrequenzidentifizierung (RFID) gesetzt. Die auftragsrelevanten Informationen befinden sich prozessnah auf dem RFID-Tag des entsprechenden Behälters oder des Transporthilfsmittels. Eine Möglichkeit der Realisierung eines solchen Steuerungssystems ist der Einsatz von Softwareagenten. Das in dieser Arbeit untersuchte dezentrale Steuerungssystem ist ein Multiagentensystem, in dem Behälteragenten untereinander sowie mit Förderstreckenagenten in Verhandlung stehen, um auf Anlagenressourcen zugreifen zu können. Das Konzept sieht eine Verteilung der Steuerungshardware (und der Steuerungssoftware) entlang der Fördertechnik vor. Durch diese Verteilung soll eine hohe Modularität der Steuerung erreicht werden. Dies stellt aber zusätzliche Anforderungen an die Echtzeitfähigkeit und die Leistung entsprechender Steuerungskomponenten.

### 1.2. Die Echtzeitproblematik und Steuerungsleistung als Forschungsgegenstände

Die Echtzeitfähigkeit der Materialflussteuerung ist eine wichtige Anforderung, welche im Gegensatz zum dezentralen Steuerungsprinzip steht (vgl. [HLL07], [Ste03]). Der Grund für diese Konfrontation liegt in der Natur verteilter Systeme mit dezentraler Datenhaltung und -verarbeitung. Einfache Entscheidungen in diesen Systemen können anhand lokaler Informationen getroffen werden. Die Dauer der Entscheidungsfindung ist dann direkt von den implementierten Algorithmen sowie der Rechnerleistung abhängig. Komplexe Entscheidungen

wie Routensuche oder Planung von Transportressourcen erfordern eine zusätzliche Kommunikation unter den Systemteilnehmern. Die Entscheidungsdauer hängt somit auch von dem Durchsatz des Kommunikationsnetzwerkes und der Effizienz verwendeter Kommunikationsprotokolle ab.

Für eine verteilte Datenverarbeitung werden zunehmend kompakte Rechnersysteme auf PC-Basis (Industrie-PC bzw. IPC) eingesetzt. Im Gegensatz zu einer Speicherprogrammierbaren Steuerung (SPS) kann die Echtzeitfähigkeit solcher Systeme nicht immer garantiert werden. Im Fall einer verteilten Steuerung auf IPC-Basis hängt die Steuerungsreaktionszeit von vielen Faktoren ab. Dazu zählen beispielsweise die Leistung einzelner Rechner und die Anzahl der laufenden Programme bzw. die Komplexität der realisierten Steuerungsfunktionen. Abhilfe schafft die Auslegung der Steuerungshardware mit einem Leistungsvorrat, was allerdings zu unbegründet hohen Beschaffungskosten führt. Eine Unterdimensionierung kann dagegen dazu führen, dass das Steuerungssystem seine Aufgabe nicht erfüllt.

Vor diesem Hintergrund wird die Ermittlung der Leistung bestehender und geplanter Materialflusststeuerungen zu einer wichtigen Aufgabe. Die Heterogenität möglicher dezentraler Lösungsansätze und breite Auswahl der Gerätetechnik macht diese Aufgabe nicht leichter. Eine Begegnung des Problems stellt diese Arbeit vor.

### 1.3. Leistung einer Materialflusststeuerung

Aufgrund der Vielseitigkeit des Begriffes Leistung (vgl. [Svo76] [Oss73] [Fer78]) hat eine allgemeingültige Definition kaum praktische Bedeutung. Im Rahmen dieser Arbeit steht die Leistung dezentraler Rechner- und Softwaresysteme in der Materialflusststeuerung im Vordergrund der Betrachtung.

Die Steuerungsleistung wird in der Praxis oft als Qualität der Steuerung im Bezug auf die zu realisierende Systemfunktion betrachtet. Die technische Grenzleistung einer intralogistischen Anlage ist dabei ein Maßstab zur Bewertung und zum Vergleich der Steuerungsqualität und somit der Steuerungsleistung (vgl. [Gud05]). Die Steuerungsleistung wird in diesem Fall auf die Qualität implementierter Steuerungsstrategien begrenzt und mittels Simulationsverfahren untersucht (vgl. z. B. [Rad00], [RFo08]). Eine Überprüfung der Leistung einer realisierten Steuerung erfolgt dann erst bei den Integrationstests oder sogar im Betrieb.

Eine andere Sicht auf die Steuerungsleistung vermutet die Betrachtung der Leistung einzelner steuerungstechnischer Komponenten. Hierbei werden die Leistungswerte für industrielle Rechner- und Kommunikationssysteme experimentell ermittelt (vgl. [TVa99], [Kle96], [LMT99], [Ros02]). Rückschlüsse auf die Qualität des gesamten Steuerungssystems werden jedoch nicht gezogen.

Für eine ganzheitliche Betrachtung der Steuerungsleistung ist es sinnvoll, Zeitgrößen einer Steuerung in einen Zusammenhang mit der Qualität der Steuerungsentscheidung zu bringen. Mit dieser Aufgabe beschäftigt sich Teilprojekt C5 – „Realtime Logistics“<sup>1</sup> im Rahmen des Sonderforschungsbereiches 696 „Anforderungsgerechte Logistik“. Das hier entwickelte Realtime-Logistics-Modell (RTL-Modell) bietet eine einheitliche Darstellung von Zeitfaktoren in der Stetigfördertechnik und deren Steuerung (siehe [LHN07], [LNH08]). Als Eingabewerte für das Modell dienen die Leistungsdaten konkreter Automatisierungsgeräte und Steuerungskomponenten.

### 1.4. Ziele und Vorgehensweise dieser Arbeit

Die aktuelle Arbeit stellt eine Fortführung des Forschungsvorhabens im Rahmen des SFB 696 dar und hat als Ziel, einen Ansatz zur Ermittlung und Analyse technischer Zeitgrößen der Steuerungssoftware und -hardware in realen Systemen zu erarbeiten. Eine besondere Herausforderung besteht darin, dass intralogistische Materialflusstsysteme sowie die entsprechenden Steuerungen hoch individuell gebaut sind. Diese Tatsache erschwert einen quantitativen Leistungsvergleich verschiedener Steuerungsrealisierungen und gibt die Motivation für diese Arbeit.

Der zu entwickelnde Ansatz soll nicht nur für eine architekturübergreifende Leistungsbewertung benutzt werden, sondern auch zur Erkennung grundsätzlicher Probleme in dezentralen Steuerungssystemen dienen. Damit sind nicht logistische Zielsetzungen wie Durchsatz o.ä. gemeint, sondern Leistungsaspekte der Steuerungsrealisierung. Das Untersuchungsobjekt dieser Arbeit setzt beispielsweise Standardtechnologien wie Ethernet ein, die nicht zwangsläufig als echtzeitfähig einstuftbar sind. Die Verwendung solcher Technologien in Bereichen mit Echtzeitanforderungen kann im Betrieb zu Verzögerungen führen, die die Zuverlässigkeit des Steuerungssystems gefährden.

Der Aufbau dieses Beitrages orientiert sich an der Vorgehensweise, die für die Leistungsanalyse produktiver Softwaresysteme verwendet wird. Zur Ermittlung der Steuerungsleistung wird zunächst ein Untersuchungsobjekt benötigt. Als solches Objekt dient in dieser Arbeit ein Steuerungsprototyp, der nach Prinzipien vom Internet der Dinge entwickelt wurde. Das Untersuchungsobjekt wird im nachfolgenden Abschnitt präsentiert. Danach wird die Experimentplanung in einzelnen Schritten erklärt. Dieser Erklärung folgt die Vorstellung des selbstentwickelten Werkzeugs, welches zur Erfassung von Messdaten und Visualisierung der Ergebnisse eingesetzt wird. Anschließend werden die Messergebnisse diskutiert. Ein Fazit und Ausblick über die Weiterführung dieser Arbeit schließt den Artikel ab.

<sup>1</sup> <http://www.sfb696.uni-dortmund.de>

## 2. Das Untersuchungsobjekt

Die zu untersuchende Steuerung ist in der Stückgutstetigförderanlage des Lehrstuhls für Förder- und Lagerwesen<sup>2</sup> (FLW) der TU Dortmund installiert. Im Folgenden werden die beiden Systeme – die zu steuernde Anlage und ihre Steuerung – kurz vorgestellt.

### 2.1. Das Stetigfördersystem

Bei dem zu steuernden System handelt es sich um eine experimentelle Anlage, die zur Erprobung verschiedener dezentraler Steuerungskonzepte verwendet wird (Abbildung 1). Die Anlage ist in zwei fördertechnisch miteinander verbundene Ebenen aufgeteilt und besteht aus insgesamt 37 angetriebenen Förderelementen, die zur Steuerungszwecken mit ca. 80 Sensoren ausgestattet sind. Die Gesamtlänge der Fördertechnik beträgt über 120 Meter.

Bei dem Stückgut handelt es sich um Mehrzweckbehälter, die mittels RFID-Technik identifizierbar sind. Zehn Identifizierungsstationen an allen wichtigen Entscheidungspunkten der Anlage sorgen für berührungsloses Schreiben und Lesen von steuerungsrelevanten Daten<sup>3</sup>.



Abbildung 1: Stetigförderanlage des Lehrstuhls FLW der TU Dortmund

Die Steuerungshardware besteht aus sieben Industrie-PCs (IPCs), die entlang der Fördertechnik installiert und per Fast-Ethernet (100Mbit/s) miteinander verbunden sind. Die IPCs<sup>4</sup> übernehmen die Rolle von SPS bei einer klassischen Steuerungsarchitektur. Die Anbindung von Sensoren und Antrieben an das Rechnersystem erfolgt über das K-Bus-System. Als Betriebssystem kommt ein Linux mit einem 2.4.x Kernel und einer Echtzeiterweiterung zum Einsatz. Diese Tatsache ermöglicht Realisierung von Steuerungsprogrammen in Hochprogrammiersprachen wie Java oder C++.

### 2.2. Materialflusssteuerung Paket Royale

*Paket Royale* ist eine prototypische Steuerungsrealisierung, welche die Idee einer dezentralen Steuerung nach dem Internet der Dinge in der Intralogistik verwirklicht (vgl. [FLH+09]). Die in Java entwickelte Steuerungssoftware basiert auf Standardtechnologien wie XML<sup>5</sup> und Web Services und wird durch einen hohen Modularisierungsgrad sowie Flexibilität charakterisiert. Zwei Arten von Steuerungskomponenten repräsentieren entsprechend die Anlagensteuerung und die Materialflusssteuerung.

Eine Anlagensteuerungskomponente realisiert die Kernfunktionen des jeweiligen Förderelements. Diese Kernfunktionen (Stellen einer Weiche, Benachrichtigen über Sensorereignisse etc.) stehen den anderen Steuerungskomponenten in Form von Diensten zur Verfügung. Der Abruf dieser Dienste erfolgt via Web Services, welche per DPWS<sup>6</sup> (Device Profiles for Web Services) verarbeitet werden.

Eine dezentrale Materialflusssteuerung implementiert testweise Betriebsstrategien, die bei Bedarf erweitert werden können. Die Komponenten der Materialflusssteuerung sind in Form von Softwareagenten realisiert. Ein solcher Agent berechnet Steuerungsentscheidungen und nutzt die Anlagendienste für die Umsetzung. Im Einsatz auf der Förderanlage des Lehrstuhls FLW wird jedem Förderelement oder Stückgut jeweils ein Steuerungsagent zugeordnet. Die Fördertechnikagenten arbeiten zyklusorientiert. In einem Steuerungszyklus werden die Sensor-

<sup>2</sup> <http://www.flw.mb.uni-dortmund.de>

<sup>3</sup> Bei der RFID-Anlage handelt es sich um ein UHF-System (865 MHz). Die an den Behältern befestigten RFID-Tags sind passive Transponder mit Speicherkapazität von 512 Bit.

<sup>4</sup> Es handelt sich dabei um Rechner mit 266 MHz Cyrix Prozessoren und einem Arbeitsspeicher von 128 MB. Der Datenspeicher ist durch eine CompactFlash Speicherkarte mit 1024 MB realisiert.

<sup>5</sup> eXtensible Markup Language (XML) ist eine Metasprache zur Beschreibung strukturierter Informationen.

<sup>6</sup> <http://www.ws4d.org>

zustände überprüft, eine Steuerungsentscheidung wird getroffen und eine Anweisung an die Anlagensteuerung geschickt.

Die Laufzeitumgebung für Steuerungsagenten wird Berechnungsgerät (*ComputationDevice*) genannt. Während die Komponenten der Anlagensteuerung stets prozessnah auf den entsprechenden IPCs arbeiten, können die Berechnungsgeräte entweder verteilt auf den IPCs oder zentral auf einem einzigen Rechner installiert werden. Im Testsystem sind aufgrund der geringen Rechnerleistung auf den IPCs zunächst nur Dienste der Anlagensteuerung installiert. Die Berechnungsgeräte werden dagegen auf einem Standard-PC mit einem 1.866 MHz Pentium-M Prozessor und 1024 MB Arbeitsspeicher gestartet. Dieser Rechner wird im Weiteren Agentenrechner genannt. Darüber hinaus wird dieser bei allen Experimenten zur Leistungsuntersuchung als Messstation verwendet.

### 3. Systematische Vorgehensweise

Eine systematische Herangehensweise zur Leistungsbewertung nach Jain [Jai91] sieht einen sequenziellen Ablauf vor. Diesem Ablauf zufolge müssen nach der Festlegung des Untersuchungsobjektes Metriken identifiziert werden. Diese Metriken stellen in unserem Fall die quantifizierbaren Leistungswerte des Steuerungssystems dar und werden im nachfolgenden Abschnitt beschrieben. Danach wird der Experimentaufbau erklärt und das Datenerfassungskonzept vorgestellt. Anschließend werden Untersuchungsszenarien erläutert.

#### 3.1. Metriken zur Bewertung der Steuerungsleistung

Metriken sind quantifizierbare Kennwerte, die zu Bewertung und Vergleich verschiedener Systeme verwendet werden. Bei der aktuellen Untersuchung wird zwischen Metriken der Rechnersysteme und Metriken der Steuerung unterschieden.

*Metriken der Rechnersysteme* charakterisieren Auslastung von Rechnerressourcen. Dazu zählen die Auslastungsdaten von Prozessoren, Arbeitsspeicher sowie der Netzwerkbandbreite. Diese verdeutlichen die Inanspruchnahme des jeweiligen Computersystems bei der Ausführung des Steuerungsprogramms. Für eine detaillierte Auslastungsanalyse ist dabei die Differenzierung einzelner ausgeführter Rechnerprozesse erforderlich.

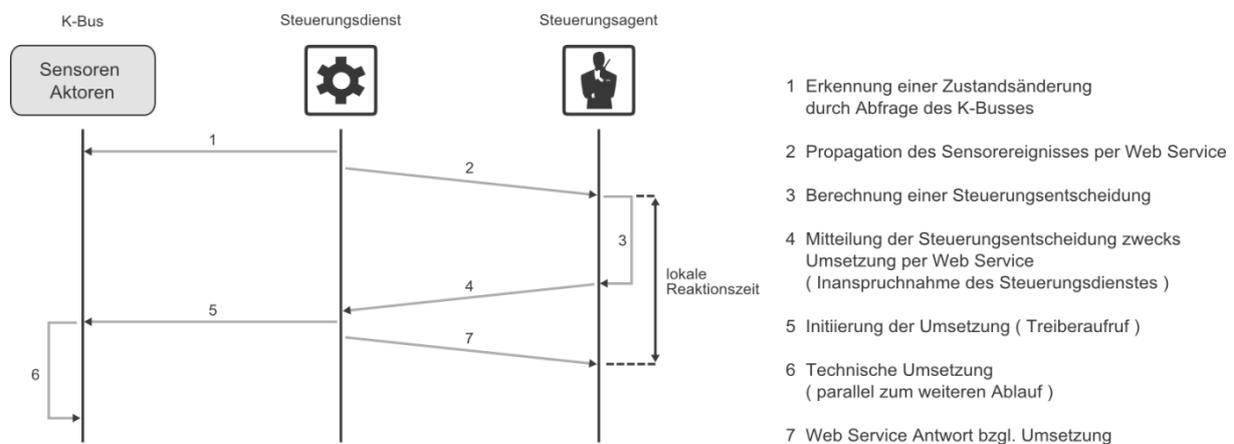


Abbildung 2: Zusammensetzung der lokalen Reaktionszeit

*Metriken der Steuerung* umfassen steuerungsspezifische Leistungswerte. Zur Beschreibung und Quantifizierung der Leistung von Materialflusssteuerungen wird der Begriff lokale Reaktionszeit eingeführt. Die lokale Reaktionszeit umfasst die Zeit, die zur Erfüllung eines Steuerungsbedarfs in einem Entscheidungsknoten benötigt wird. Der Zeitraum beginnt bei dem ersten Bekanntwerden eines solchen Bedarfs, schließt alle steuerungsbezogenen Aktivitäten ein und endet mit der Umsetzung der Steuerungsentscheidung. Dank einem hohen Abstraktionsgrad kann diese Metrik auf eine Vielzahl von Steuerungsapplikationen angewendet werden.

Im Fall der Leistungsbewertung einer Steuerungssoftware umfasst die lokale Reaktionszeit den beschriebenen Zeitraum, jedoch exklusive der technischen Umsetzung (z.B. Schalten eines Antriebes oder Stellen einer Weiche). In diesem Fall setzt sich die lokale Reaktionszeit wie in Abbildung 2 skizziert zusammen.

Anhand Abbildung 2 wird deutlich, dass bei einem zyklischen Steuerungsablauf die Dauer des Verarbeitungszyklus genau der lokalen Reaktionszeit entspricht. Im Weiteren wird der Begriff lokale Reaktionszeit gleichbedeutend mit der Dauer des Berechnungszyklus eines Steuerungsagenten verwendet. Für die Erfassung der lokalen Reaktionszeit müssen sich die Steuerungsagenten die Zyklusdauer merken und diese dem

Messsystem zur Verfügung stellen. Der Aufbau des entsprechenden Experimentes wird im folgenden Abschnitt beschrieben.

### 3.2. Experimentaufbau

Der Aufbau des Experimentes zur Messdatenerfassung ist durch die dezentrale Struktur der zu untersuchenden Steuerung geprägt. Das Messsystem ist in zwei Hierarchieebenen gegliedert (vgl. Abbildung 3). Die Erfassung von Auslastungswerten erfolgt durch die Datenerfassungssoftware, die prozessnah auf den IPCs sowie auf dem Agentenrechner aktiv ist. Eine per Netzwerk angebundene Auswertungssoftware ruft die Messdaten periodisch ab, ermöglicht Visualisierung und unterstützt die Auswertung dieser Daten. Das Datenerfassungstool und das Auswertungstool sind beide Teile des Systems *Monitor Royale*, einer Eigenentwicklung, welche in Abschnitt 4 genauer erläutert wird.

Die Schnittstelle zwischen den Steuerungsagenten und dem Datenerfassungstool ist über Dateisystem realisiert. Ein Steuerungsagent trägt die Zeitwerte in eine Datei ein. Das Datenerfassungstool liest die Daten alle 1000 ms aus und nimmt zusätzlich die Auslastungswerte des entsprechenden Rechnersystems auf. Ein solches Messobjekt umfasst die zuvor erwähnten Auslastungswerte und – aufgrund variabler Laufzeiten der Agentenzyklen - eine Liste mit lokalen Reaktionszeitwerten. Einzelne Messobjekte sind mit einer Identifikationsnummer eindeutig gekennzeichnet. Als solcher wird der Zeitstempel der durchgeführten Messung verwendet.

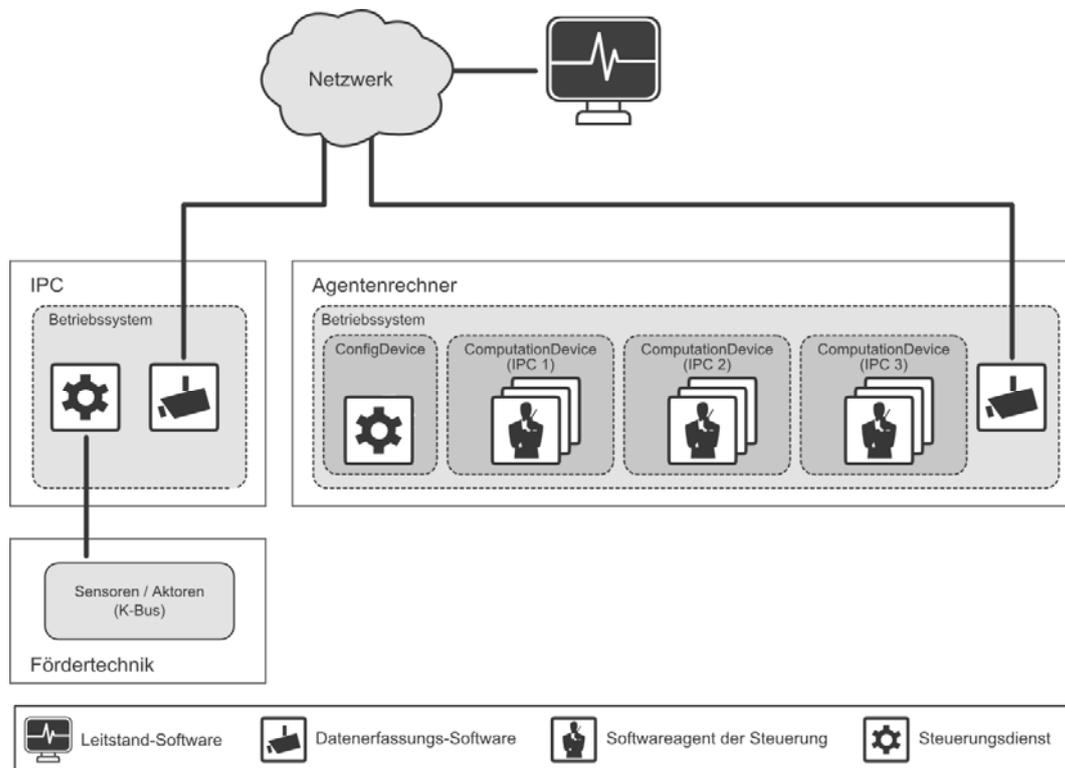


Abbildung 3: Experimentaufbau der Messdatenerfassung

### 3.3. Untersuchungsszenarien

Zur Erfassung von Last- und Leistungswerten wurden Untersuchungsszenarien entwickelt, welche typische Kernaufgaben (dezentraler) Materialflusssteuerungen repräsentieren. Diese Szenarien können in verschiedenen Systemen realisiert und zum Leistungsvergleich benutzt werden. Als variable Größen werden dabei die Anzahl aktiver Steuerungsentitäten im System und der lokale Durchsatz (Anzahl des transportierten Stückguts pro Zeiteinheit durch einen Förderabschnitt) verwendet.

- *Szenario 0: Leistung des Monitoring-Systems*  
Das Messsystem selbst kann in einem Rechnersystem eine Last verursachen. In diesem Szenario werden entsprechende Leistungswerte ermittelt, damit diese bei allen folgenden Szenarien berücksichtigt werden können.
- *Szenario 1: Steuerungsauslastung im Leerlauf*  
In diesem Szenario wird das Systemverhalten ohne Arbeitslast untersucht. Dezentrale Steuerungsentitäten sind aktiv, es entsteht jedoch kein Durchsatz.

- *Szenario 2: Anlauf von Steuerungsentitäten*  
In einer dezentralen Steuerung können Steuerungsentitäten einzeln oder gruppenweise dem System hinzugefügt werden. In diesem Szenario ist nur eine Teilmenge von Steuerungsentitäten zu einem bestimmten Zeitpunkt aktiv. Zur Laufzeit wird eine vorgegebene Anzahl von Steuerungsentitäten aktiviert und die Auslastung von Steuerungsrechnern wird ermittelt.
- *Szenario 3: Abschalten von Steuerungsentitäten*  
Dieses Szenario ähnelt sich an das Szenario 2 mit einer Ausnahme. Hier wird bei einem laufenden System eine vorgegebene Anzahl von Steuerungsentitäten deaktiviert. Die Systemleistung wird dabei untersucht.
- *Szenario 4: Leistung der Verteilsteuerung (ein Paket)*  
Eine Grundfunktion der Steuerung intralogistischer Anlagen ist die Zielsteuerung, die durch fördertechnische Verteilelemente realisiert wird. In diesem Testszenario ist eine für diese Aufgabe zuständige Steuerungsentität aktiv sowie eine Reihe weiterer an der Verhandlung (z.B. Routensuche) beteiligten Entitäten. Die Steuerungsleistung wird beim Vorbeifahren eines Transportguts durch dieses Verteilelement ermittelt.
- *Szenario 5: Leistung der Sammelsteuerung (ein Paket)*  
Realisierung von Vorfahrtstrategien bei sämtlichen Zusammenführungen ist eine weitere Grundaufgabe der Materialflusssteuerung. Die Steuerungsleistung eines Sammelelementes unter einer einfachen Arbeitslast wird in diesem Szenario untersucht.
- *Szenario 6: Verhalten unter Arbeitslast*  
In diesem Szenario wird im Testbetrieb die Arbeitslast der Förderanlage variiert. Die Steuerungsleistung sowie die Auslastung von Netzwerk- und Rechnerressourcen werden dabei für einzelne lokale Steuerungen aufgenommen und anschließend analysiert. Dieses Szenario schließt die Szenarien 1, 4 und 5 ein.

## 4. Monitoring-Software Monitor Royale

Den Benutzern und Administratoren von Rechnersystemen steht eine Vielzahl verschiedener Monitoring-Systeme zur Auswahl. Darunter fallen einfache Programme wie Linmark<sup>7</sup> und Panalyser<sup>8</sup> aber auch komplexere wie OProfiler<sup>9</sup> oder PerfSuite<sup>10</sup> des U.S. National Center for Supercomputing Applications. Trotz dieser Vielzahl existiert aktuell kein Werkzeug, welches die Anforderungen im Rahmen dieser Forschungsarbeit erfüllen kann. Unter Anderen sind hier insbesondere folgende Anforderungen zu nennen: Verfügbarkeit der Daten per Netzwerk, abstrakte Datenschnittstelle zur Aufzeichnung individueller Daten sowie die funktionale Erweiterbarkeit. Diesem Bedarf beugend wurde die Monitoring-Software Monitor Royale entwickelt, die als Open-Source-Software veröffentlicht<sup>11</sup> wurde.

Monitor Royale wurde speziell für die Leistungsuntersuchung in dezentralen Steuerungssystemen entwickelt und mit dem System Paket Royale getestet. Dank seiner generischen Architektur ist Monitor Royale auch für die Leistungsuntersuchung verschiedener anderer Soft- und Hardwaresysteme einsetzbar. Das Werkzeug ist in C++ entwickelt und hat Linux-/Unix-Systeme als Zielplattform. Monitor Royale besteht aus zwei wesentlichen Komponenten: Monitoring Agent (Datenerfassung) und Evaluation Station (Visualisierung und Analyse). Ein Monitoring-System kann je nach Bedarf aus mehreren solcher Komponenten bestehen. Im allgemeinen Fall ist eine m-zu-n-Beziehung realisierbar. Die Programmteile sind untereinander lose gekoppelt. Die Kommunikation erfolgt per RPC (engl. remote procedure call) über eine TCP/IP-Verbindung. Neben dem Datenabruf per Netzwerk ist die Nutzung einer Datendatei möglich.

Der *Monitoring Agent* ist der Programmteil, der auf den zu überwachenden Systemen zum Einsatz kommt. Aufgezeichnet werden neben den Auslastungsinformationen von Prozessor, Arbeitsspeicher und Netzwerk auch Informationen über aktive Prozesse. Die Auslastungsinformationen des Rechnersystems werden aus dem proc-Dateisystem<sup>12</sup> des Betriebssystems entnommen. Die Leistungsdaten der Steuerung werden aus einer Schnittstellendatei ausgelesen. Diese Datei beinhaltet Schlüsselwert-Paare von einzelnen Messobjekten und kann individuell gestaltet werden. Die Nutzung einer Textdatei erhöht die Flexibilität der Software. Jedes Programm mit der Fähigkeit zu Dateisystemoperationen ist in der Lage, der Monitoring-Software Daten zwecks Aufzeichnung zukommen zu lassen.

<sup>7</sup> <http://linmark.sourceforge.net>

<sup>8</sup> <http://panalyser.sourceforge.net>

<sup>9</sup> <http://oprofile.sourceforge.net>

<sup>10</sup> <http://perfsuite.ncsa.uiuc.edu>

<sup>11</sup> <http://monitorroyale.sourceforge.net>

<sup>12</sup> Unter proc-Dateisystem versteht man bei Unix-Betriebssystemen eine Verzeichnisstruktur, in der betriebssysteminterne Zustände in Form von Dateien abgebildet werden.

*Evaluation Station* ist die grafische Benutzerschnittstelle von Monitor Royale. Diese Komponente dient gleichermaßen der Kontrolle des Monitorings wie auch der Verwaltung und Visualisierung erfasster Daten. Das Visualisierungswerkzeug ermöglicht eine on-line Verfolgung der Datenaufnahme. Weitere integrierte Werkzeuge helfen bei der Analyse aufgenommener Daten. Abbildung 4 zeigt die Bedienoberfläche des Programms. Eine Vielzahl von integrierten Funktionen ermöglicht verschiedene Darstellungsformen wie beispielsweise Skalierung der Zeit- bzw. der Wertachse oder Auswahl bestimmter Messdatenbereiche. Eine gleichzeitige Darstellung mehrerer Messgrößen in einem Diagramm ermöglicht eine Vergleichsanalyse. Darüber hinaus werden verschiedene statistische Informationen berechnet und als Zahlenwerte aber auch als Diagramme dargestellt. Die Benutzerfreundlichkeit gewinnt durch ergonomisches Oberflächendesign und intuitive Piktogramme zur Programmsteuerung.

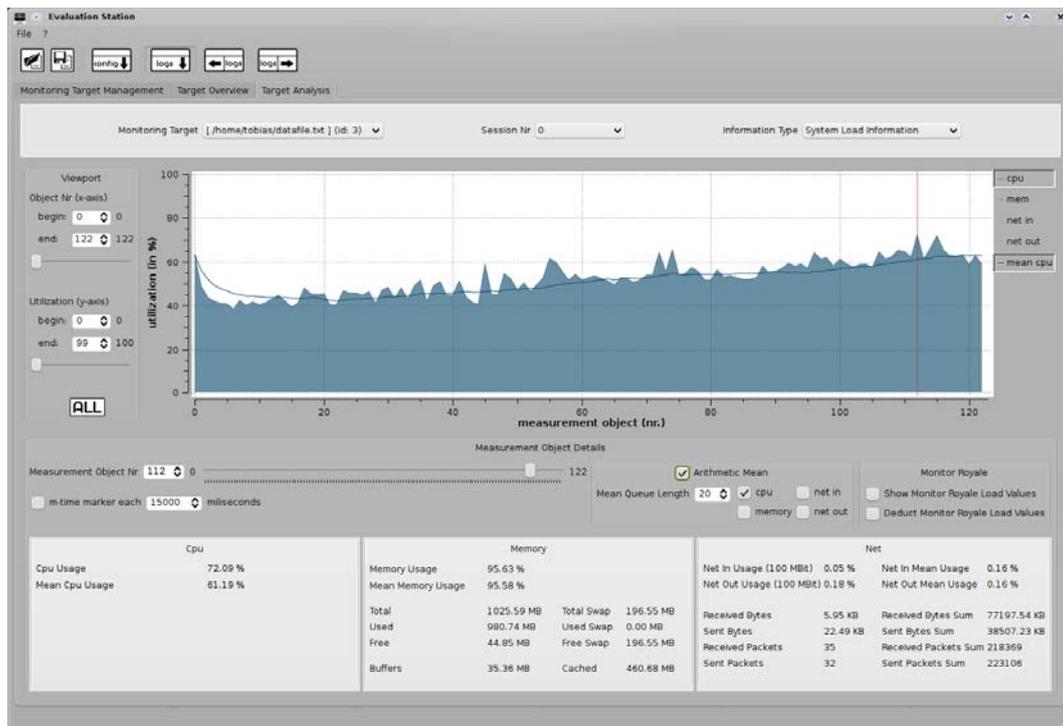


Abbildung 4: Visualisierung der Systemlastinformationen durch Evaluation Station

## 5. Ergebnisse

Das Steuerungssystem Paket Royale wurde an den vorgestellten Szenarien ausführlich getestet. Die Testdurchläufe betrachteten bei bestimmten Szenarien bis zu 2 Stunden. Das untersuchte Steuerungssystem setzt nicht echtzeitfähige Technologien, wie Ethernet und PC-basierte Steuerungshardware, im zeitkritischen Bereich der Anlagen- und Materialflusssteuerung ein. Die ermittelten Leistungskennzahlen sollen daher zum Verständnis von Problematiken beitragen, die in solchen Systemen durch ungewünschte Verzögerungen der Steuerungsreaktion eventuell zu Stande kommen. Vor diesem Hintergrund werden im Folgenden die Messergebnisse präsentiert und diskutiert, die in Tests nach Szenarien 1 und 6 entstanden sind. Die Betrachtung von diesen beiden Szenarien ist dadurch begründet, dass diese dem Systemverhalten im Normalbetrieb entsprechen. Dabei beinhaltet Szenario 6 die Szenarien 4 und 5 und stellt somit einen allgemeineren Fall dar. Szenarien 2 und 3 haben eine Änderung der Fördertopologie (z. B. beim Anlauf der Steuerung) zur Folge. Sie können als Spezialfälle aufgefasst werden, die nur sehr selten auftreten und daher für die Untersuchung von Leistungsgpässen im normalen Betrieb vernachlässigbar sind.

### 5.1. Steuerungsauslastung im Leerlauf (Szenario 1)

Im Leerlaufbetrieb werden keine Lastobjekte durch das Fördersystem transportiert. Die Aktivität der Steuerungsagenten reduziert sich lediglich auf die Durchführung des Steuerungszyklus inklusive einer Abfrage der Prozesszustände. Die entsprechend geringe Auslastung des Agentenrechners kann aus Abbildung 5 entnommen werden. Dabei werden die Messreihen für CPU-Auslastung (oben) und Datenspeichernutzung (unten) beispielhaft für einen Zeitraum von ca. 60 Sekunden dargestellt. Die unteren Messreihen auf den beiden Diagrammen (ca. 6% der CPU-Last und 2,6% der Speichernutzung) entsprechen den drei Agentenlaufzeit-

umgebungen (ComputationDevice) sowie einem Konfigurationsprogramm (ConfigDevice), die auf dem Agentenrechner gleichzeitig laufen. Die resultierende Auslastung des Agentenrechners beträgt 15 bis 20 % für die CPU und 8,89 % für den Speicher. Die bestehende CPU-Last ist auf die stets durchgeführten Steuerungszyklen in den Steuerungsagenten zurückzuführen. Die gemessenen Zykluszeiten der Agenten liegen vollständig unterhalb der definierten Aufzeichnungsschwelle von 1 ms. Der Speicherverbrauch von 91,22 MB ist maßgeblich auf die virtuellen Maschinen von Java (JVM) zurückzuführen. Netzwerkkommunikation ist neben regelmäßigen Abonnement-Erneuerungen für Sensorereignisse bei diesem Szenario nicht vorhanden.

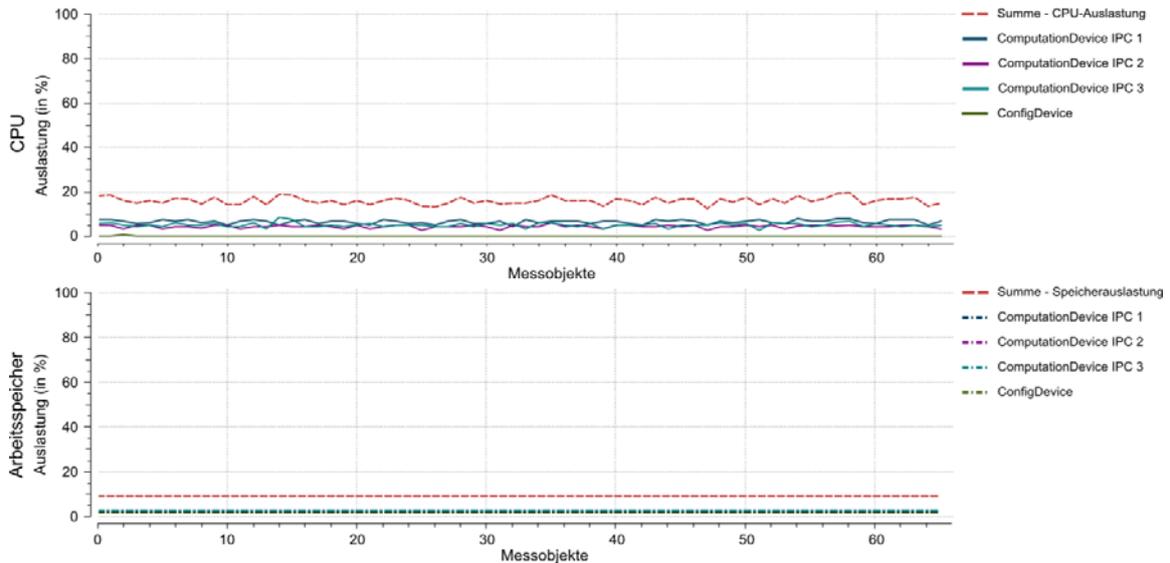


Abbildung 5: CPU- und Speicherauslastung des Agentenrechners für Szenario 1 (0 transportierte Pakete)

## 5.2. Verhalten unter Arbeitslast (Szenario 6)

Im Szenario 6 wird das Steuerungssystem unter variiertter Arbeitslast untersucht. Bei diesem Experiment wird nur der untere Kreislauf der Förderanlage in Betrieb genommen. Aufgrund einer vorhandenen Überholstrecke muss die Steuerung eine Wegentscheidungsfunktion sowie eine Vorfahrtstrategie realisieren. Die Anzahl der im System rotierenden Lastobjekte wurde von 0 auf 12 schrittweise erhöht. Die in Abbildung 6 dargestellten Messdaten zeigen die Aufnahme bei der maximalen Arbeitslast über einen Zeitraum von ca. 300 Sekunden. Bei dem Experiment kreisten die 12 Pakete mit einer durchschnittlichen Geschwindigkeit von ca. 1 m/s über die Förderanlage. Bei einem Abstand von durchschnittlich ca. 5 m pro Paket wurde somit ein stetiger Paketstrom von ca. 720 Paketen pro Stunde simuliert.

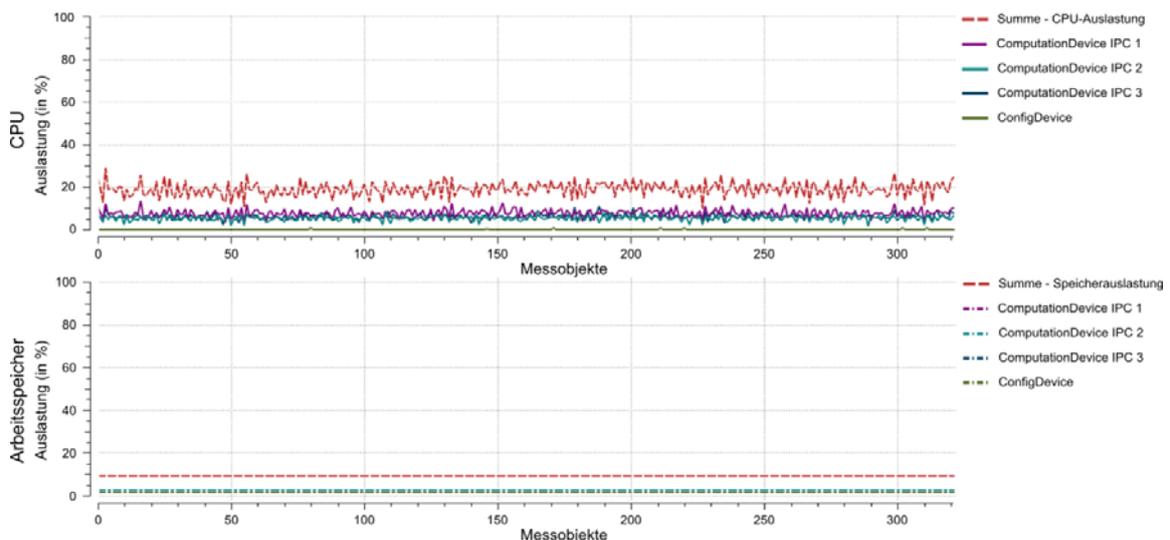


Abbildung 6: CPU- und Speicherauslastung des Agentenrechners für Szenario 6 (12 transportierte Pakete)

Vergleicht man die Auslastung des Agentenrechners mit diesen aus dem Szenario 1, so kann nur eine geringe Steigerung festgestellt werden. Genauere Untersuchungen haben ergeben, dass die erhöhte Prozessorlast primär

auf die Verarbeitung von Web Service Aufrufe zurückzuführen ist. Die Skalierbarkeit ist daher stark von der Anzahl von Web Service Nachrichten abhängig. Diese hängt wiederum von der Anzahl der Sensoren ab, deren Auslösung eine Ereignispropagation per Web Service verursacht. Der Speicherverbrauch dagegen unterscheidet sich nicht von dem aus dem Szenario 1. Dies wird dadurch erklärt, dass die Initialisierung aller Entitäten (Steuerungsdienste, Fördereragenten) bereits abgeschlossen ist. Ein erhöhter Speicherverbrauch durch eine erhöhte Anzahl an Steuerungsagenten ist für 12 Pakete nicht erkennbar.

Die Netzwerkauslastung des Agentenrechners ist für Szenario 6 gering. Der arithmetische Durchschnitt beträgt 0,42 % für eingehenden und ausgehenden Verkehr zusammen (Abbildung 7). Vergleicht man diesen Wert mit den Leistungsbewertungen von Ethernet (vgl. [RWN+95]), so können Kollisionen beim Medienzugriff vernachlässigt werden. Das nicht-deterministische CSMA/CD Zugriffsverfahren wirkt sich in der Anwendung mit Echtzeitanforderung in diesem Fall nicht störend aus. Die Skalierbarkeit kann durch Subnetzbildung weiter erhöht werden.

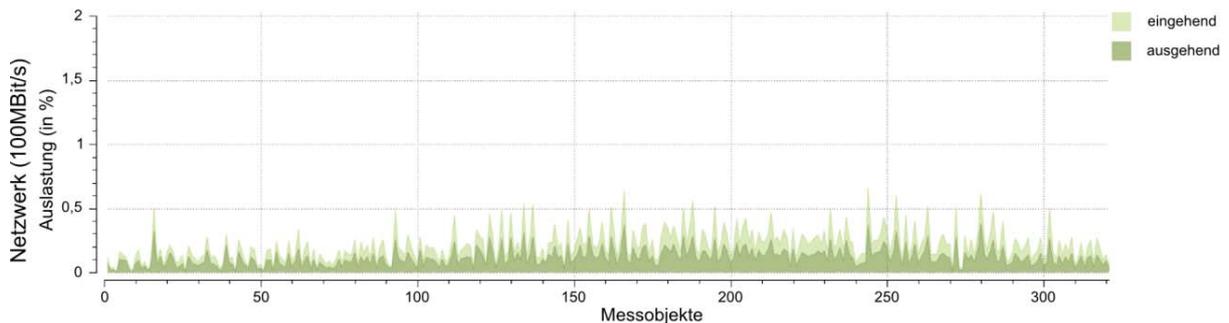


Abbildung 7: Netzwerkauslastung des Agentenrechners für Szenario 6 (12 transportierte Pakete)

Die Arbeitslast macht sich an der Erhöhung der lokalen Reaktionszeiten von Steuerungsagenten bemerkbar. Abbildung 8 stellt die erfassten Reaktionszeiten eines Weichenagenten dar. Die meisten Zykluszeiten sind im Bereich von 150-160 ms verteilt. Auffällige Werte am Anfang des erfassten Spektrums sowie die Werte, die unterhalb der Aufzeichnungsschwelle liegen repräsentieren die Zykluszeit im Leerlauf. Die Ausreißer mit Werten größer als 520 ms lassen sich auf die Arbeit des *Java Garbage Collectors*<sup>13</sup> zurückführen. Ignoriert man diese Ausreißer, ergibt sich eine durchschnittliche lokale Reaktionszeit von 167 ms. Die negative Wirkung der langen Reaktionszeit und insbesondere der Ausreißer konnte durch ein proaktives Verhalten von Steuerungs-

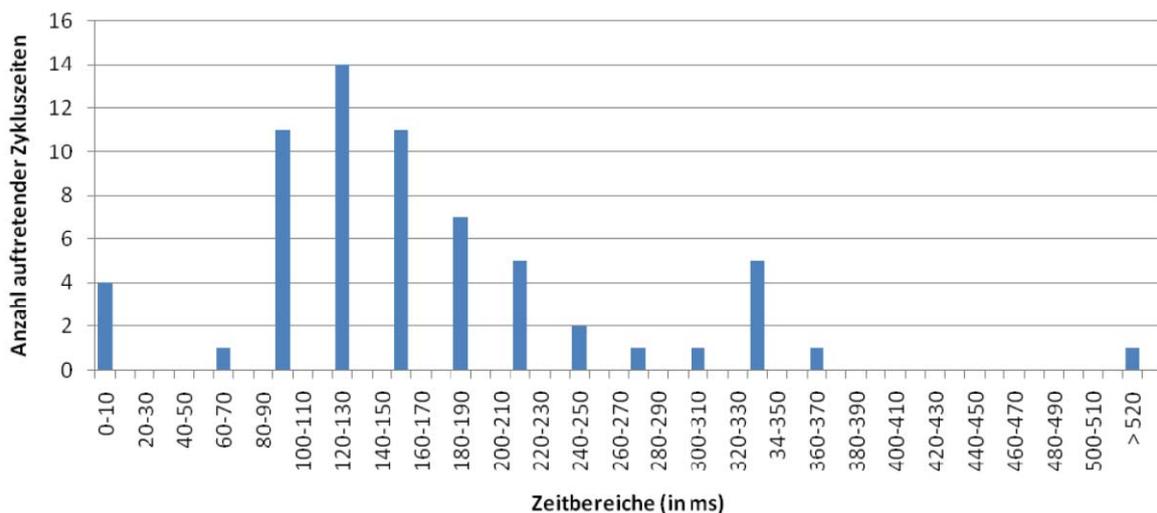


Abbildung 8: Zykluszeitverteilung des Steuerungsagenten für Fördererelement U1

agenten entschärft werden. Dabei werden nicht zeitkritische Steuerungsentscheidungen im Vorfeld vorbereitet. Ziel ist es, den Berechnungsaufwand zu kritischen Zeitpunkten zu reduzieren. Während ein Paket beispielsweise zu einem Entscheidungspunkt (Zusammenführung bzw. Verzweigung von Förderwegen) transportiert wird, nutzt

<sup>13</sup> Der Garbage Collector ist eine integrierte Routine der Java Laufzeitumgebung (Java Virtual Machine) mit der Aufgabe die Arbeitsspeichernutzung während Programmausführungen zu optimieren. Die Ausführung dieser Routine beansprucht die vorhandene Rechenleistung. Der Ausführungszeitpunkt und -dauer hängen von der jeweiligen Programmsituation sowie eingebauter Algorithmen des Garbage Collectors ab und sind nicht deterministisch vorhersagbar. Anwendungen mit Echtzeit-Anforderung wird u.a. empfohlen diese Speicherbereinigung präventiv zu unkritischen Zeitpunkten manuell zu initiieren.

der zuständige Behälteragent diese Beförderungszeit, um die weitere Transportrichtung zu berechnen oder die Vorfahrt zu regeln.

### 5.3. Fazit

Der hier beschriebene Ansatz hat sich nicht nur die Erfassung von Leistungskennzahlen zum Ziel gesetzt. Vielmehr dienen die vorgestellten Kennzahlen, Werkzeuge und die Vorgehensweise zur Analyse bestehender steuerungstechnischer Lösungen auf Probleme und Leistungsengpässe, die sich mithilfe von grafisch dargestellten Messergebnissen leicht identifizieren lassen. Bei den durchgeführten Experimenten konnten die vermuteten Engpässe aufgrund einer zeitlich nicht-deterministischen Nachrichtenübertragung nicht bestätigt werden. Die Ausreißer bei den Zykluszeiten von Steuerungsagenten konnten quantitativ gemessen werden und lassen sich auf die Arbeit vom Java Garbage Collector zurückschließen. Dieses Verhalten gefährdet die Zuverlässigkeit der Steuerung und muss beim Steuerungsdesign berücksichtigt werden. Der Hauptanteil der Zykluszeiten (durchschnittlich 167 ms) ist durch Versand und Verarbeitung von Web Service Nachrichten verursacht (vgl. Abbildung 2). Eine Optimierung verwendeter Bibliotheken oder Nutzung eines effizienteren Kommunikationsprotokolls wird die gesamte Reaktionszeit der Steuerung verbessern.

## 6. Zusammenfassung und Ausblick

Der vorliegende Beitrag stellt einen werkzeuggestützten Ansatz zur Bewertung und Vergleich von Leistungen dezentraler Steuerungssysteme vor. Der Ansatz beinhaltet einen Vorschlag zur Auswahl von Leistungsmetriken, Festlegung variabler Parameter der Untersuchung sowie Aufstellung und Durchführung von Testszenarien. Zur automatischen Datenerhebung wurde das Monitoring-System Monitor Royale entwickelt, welche als Open-Source-Software verfügbar ist. Dank seiner konfigurierbaren Messdatenschnittstelle auf Dateibasis und guten Skalierbarkeit kann das System Monitor Royale für Aufzeichnung und Visualisierung von Messdaten in verschiedenen Testszenarien eingesetzt werden. Der Ansatz wurde für die Untersuchung einer exemplarischen Materialflusssteuerung Paket Royale angewandt. Diese dezentrale Steuerung wurde am Lehrstuhl für Förder- und Lagerwesen (FLW) der TU Dortmund entwickelt und läuft in einer verteilten Hardwareumgebung. Bei der durchgeführten Untersuchung konnten Leistungsmetriken der Steuerung in mehreren Testszenarien quantitativ erfasst und analysiert werden.

Der vorgestellte Ansatz betrachtet allgemeine Kernfunktionalitäten einer nach Internet der Dinge für die Intralogistik entwickelten Materialflusssteuerung, um eine architekturübergreifende Bewertung zu ermöglichen. Die nach diesem Ansatz ermittelten Kennzahlen des Steuerungssystems Paket Royale eignen sich für die Leistungsbewertung und Leistungsvergleich unter verschiedenen Einsatzbedingungen. Bei den durchgeführten Experimenten konnten zeitliche Engpässe im Verhalten der untersuchten Steuerung aufgedeckt werden. Diese Ergebnisse bilden eine Grundlage für eine anschließende Optimierung des Steuerungsdesigns. Durch die Anwendung auf ein spezifisches Steuerungssystem konnte die Einsetzbarkeit des allgemeinen Ansatzes sowie die Aufdeckung steuerungstechnischer Leistungsengpässe gezeigt werden. Das Ziel eines architekturübergreifenden Bewertungsansatzes zum quantitativen Leistungsvergleich verschiedener Steuerungsrealisierungen wurde somit erfüllt.

Dieser Beitrag ist im Rahmen des Teilprojektes „Realtime Logistics“ des Sonderforschungsbereiches 696 entstanden. Ein weiterer Entwicklungsschritt ist die Integration der vorgeschlagenen Untersuchungsmethode in das Realtime-Logistics-Modell zur ganzheitlichen Beschreibung und Analyse vom Zeitverhalten dezentraler Steuerungssysteme in den intralogistischen Anlagen. Die mit dem System Monitor Royale erfassten Zeit- und Leistungsdaten dienen dabei als Eingabewerte für das RTL-Modell. Um dieses Vorhaben zu ermöglichen sind weitere Experimente zur Erfassung von Messdaten an realen Steuerungssystemen erforderlich.

## Literatur

- [BSc01] Bussmann, S.; Schild, K.: An Agent-based Approach to the Control of Flexible Production Systems. In: Proceedings of the 8th IEEE Int. Conf. on Emergent Technologies and Factory Automation (ETFA 2001), 2001. S. 481-488.
- [Fer78] Ferrari, D.: Computer systems and performance evaluation. Prentice Hall, Englewood Cliffs, 1978.
- [FLH+09] Feldhorst, S.; Libert, S.; ten Hompel, M.; Krumm, H.: Integration of a Legacy Automation System into a SOA for Devices. In Proc. of 14th IEEE International Conference on Emerging Technologies and Factory Automation, IEEE, IES 2009.
- [GCK08] Günthner, W. A.; Chisu, R.; Kuzmany, F.: Internet der Dinge - Zukunftstechnologie mit Kostenvorteil. In: Fördern und Heben, Ausgabe 10, 2008, S. 556-558.
- [GtH10] Günthner, W. A. (Hrsg.); ten Hompel, M. (Hrsg.): Internet der Dinge in der Intralogistik. Springer, 2010.
- [Gud05] Gudehus, Timm: Logistik: Grundlagen - Strategien - Anwendungen. 3. Aufl., Springer, 2005.
- [HLL07] ten Hompel, M.; Libert, S.; Liekenbrock, D.: Analyse der Echtzeitproblematik bei der Steuerung von Stetigfördersystemen. In: Crostack, H.-A.; ten Hompel, M. (Hrsg. Bd.): 1. Kolloquium des Sonderforschungsbereich 696, Dortmund : Verlag Praxiswissen, 2007, S. 235-255.
- [HLS06] ten Hompel, M.; Libert, S.; Sondhof, U.: Dezentrale Steuerung für Materialflusssysteme. In: Logistics Journal, 2006, S. 1-9.
- [Jai91] Jain, R.: The Art of Computer Systems Performance Analysis. John Wiley & Sons, Inc., 1991.
- [Kle96] Klehmet, U.: Mathematisch-heuristische Leistungsbewertung des Medienzugangsverfahrens beim Profibus. In: Automatisierungstechnik, Ausgabe 3, 1996, S. 108-119.
- [LHN07] Libert, S.; ten Hompel, M.; Nettsträter, A.: Modell zur Beschreibung von Zeitfaktoren in der automatisierten Stetigfördertechnik. In: Logistics on Demand, 2007, S. 167-188.
- [LNH08] Libert, S.; Nettsträter, A.; ten Hompel, M.: Das RTL-Modell für dezentrale Materialflusssteuerung. In: Crostack, H.-A.; ten Hompel, M. (Hrsg. Bd.): Berichte aus dem SFB 696, Dortmund : Praxiswissen, 2008, S. 137-161.
- [LMT99] Lian, F.-L.; Moyne, J.; Tilbury, D.: Performance Evaluation of Control Networks: Ethernet, ControlNet, and DeviceNet. In: IEEE Control Systems Magazine, Band 21, Ausgabe 1, 1999, S. 66-83.
- [Lor03] Lorentz, K.: Ein Beitrag zur verteilten und adaptiven Materialflusssteuerung für Stückguttransport. Dissertation, Magdeburg Universität, Fakultät für Maschinenbau, 2003.
- [Oss73] Osswald, B.: Leistungsvermögensanalyse von Datenverarbeitungsanlagen. Toeche-Mittler Verlag, Darmstadt, 1973.
- [Rad00] Radtke, A.: Beitrag zur Entwicklung optimierter Betriebsstrategien für Sortiersysteme. Dissertation, Universität Dortmund, Fakultät Maschinenbau, 2000.
- [RFo08] Follert, G.; Roidl, M.: Evaluation of Routing Strategies for Decentralized Self-Organisation in Large Scale Conveyor Systems. In: Kimberly E. et al. (Hrsg. Bd.): Progress in Material Handling Research: 2008, Charlotte (NC, USA) : The Material Handling Institute of America, 2008.
- [Ros02] Rostan, M.: Zykluszeit ist nicht alles: zur Performance von Feldbussystemen in der Fertigung. : SPS/IPC/DRIVES Nürnberg, 13. Fachmesse und Kongress, Heidelberg, 2002, S. 99-108.
- [RWN+95] Rindos, A.; Woollet, S.; Nicholson, L.; Vouk, M.; Nog, S.; Kotz, D.: A Performance Comparison of TCP/IP and MPI on FDDI, Fast Ethernet, and Ethernet. Technical Report, Dartmouth College, Hanover, 1995.
- [Ste03] Steusloff, H.: Verteilte Echtzeitsysteme und eingebettete Systeme - Über Systeme und Technologien. : Verteilte Echtzeitsysteme, Fachtagung der GI-Fachgruppe 4.4.2 „Echtzeitprogrammierung und PEARL (EP)“, Berlin : Springer, 2003.
- [Svo76] Svobodova, L.: Computer Performance Measurement and Evaluation Methods: Analysis and Application. Elsevier, New York, 1976.
- [TVa99] Tovar, E.; Vasques, F.: Analysis of the Worst-Case Real Token Rotation Time in PROFIBUS Networks. In: Proceedings of the Fieldbus Conference (FeT '99), 1999.
- [Wil06] Wilke, M.: Wandelbare automatisierte Materialflusssysteme für dynamische Produktionsstrukturen. Dissertation, TU München, Fakultät für Maschinenwesen, 2006.
- [Win08] Windt, K.: Ermittlung des angemessenen Selbststeuerungsgrades in der Logistik - Grenzen der Selbststeuerung. In: Nyhuis, P. (Hrsg. Bd.): Beiträge zu einer Theorie der Logistik, Berlin [u.a.] : Springer, 2008, S. 349-372.