

Institut für Formale Methoden der Informatik
Abteilung Algorithmik
Universität Stuttgart
Universitätsstraße 38
D-70569 Stuttgart

Bachelorarbeit Nr. 30

Lagerplatzhaltung – Formulierung und Lösung als diskretes Optimierungsproblem

Stefanie Götz

Studiengang:	Softwaretechnik
Prüfer:	Prof. Dr. Stefan Funke
Betreuer:	Prof. Dr. Stefan Funke Tobias Appelmann (AEB GmbH)
begonnen am:	01. Oktober 2012
beendet am:	31. Januar 2013
CR-Klassifikation:	H.4.1, H.4.2, J.4

Inhaltsverzeichnis

1. Einleitung	1
1.1. Motivation und Aufgabenstellung	1
1.2. Gliederung	2
2. Grundlagen	3
2.1. Bekannte Schnellläuferstrategien	3
2.2. Bezugsgröße: Artikel vs. Lagereinheiten	4
3. Konzept	7
3.1. Das Basisproblem	8
3.2. Statische Optimierung	10
3.2.1. Lineares Lager ohne Nachlieferungen	10
3.2.2. Realistisches Lager ohne Nachlieferungen	12
3.2.3. Realistisches Lager mit Nachlieferungen	16
3.2.4. Strategische Parameter	21
3.2.5. Behandlung von Sonderfällen	23
3.3. Dynamische Optimierung	24
4. Implementierung	27
4.1. Design	27
4.2. Umsetzung und Annahmen	28
4.3. Algorithmen	31
4.3.1. Schwerste stabile Menge im Intervallgraphen	31
4.3.2. Färbe-Algorithmus für Intervallgraphen	32
5. Evaluation	35
5.1. Empirische Untersuchungen	35
5.1.1. Im Lager ohne Nachlieferungen	36
5.1.2. Im Lager mit Nachlieferungen	36
5.2. Schlussfolgerungen	39
6. Zusammenfassung und Ausblick	41
6.1. Fazit	41
6.2. Anknüpfungspunkte	41
A. Anhang	43
A.1. Beweise	43
A.1.1. Optimalität der Sortier-Strategie im linearen Lager	43

A.1.2.	Ein Split ist besser als zwei Splits	43
A.1.3.	Keine Reorganisation wegen Split bei Einlagerung von hinten im realistischen Lager	44
A.1.4.	Job-Scheduling-Strategie erzielt optimale Frequentierung	45
A.1.5.	Optimale Lösung besteht aus maximalen stabilen Mengen	45
A.2.	Anlagen zu den durchgeführten Testreihen	46
A.2.1.	Verteilung der Bestellgrößen pro Betrachtungszeitraum	46
A.2.2.	Anzahl disjunkter Bestellungen pro Betrachtungszeitraum	50
A.2.3.	Ergebnisse im Lager ohne Nachlieferungen	50
A.2.4.	Ergebnisse im Lager mit Nachlieferungen	51
B.	Glossar	53
	Literaturverzeichnis	55

1. Einleitung

1.1. Motivation und Aufgabenstellung

Ziel einer effizienten Lagerhaltung ist die derartige Platzierung von Lagereinheiten, dass die Gesamtkosten für vorhersehbare Zugriffe auf diese Lagereinheiten (zum Beispiel bei der Kommissionierung von Bestellungen) möglichst niedrig sind.

Beispiel:

Gegeben sei eine Bestellung *A*, der vier Lagereinheiten zugeordnet sind. Die Abbildungen 1.1 und 1.2 zeigen zwei Möglichkeiten der Platzierung dieser vier Lagereinheiten im Lager.

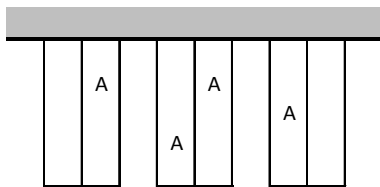


Abbildung 1.1.: Hohe Kommissionierkosten

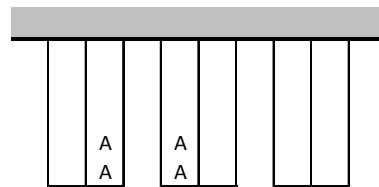


Abbildung 1.2.: Niedrige Kommissionierkosten

Um alle zu Bestellung *A* gehörenden Lagereinheiten einzusammeln, muss in dem von Abbildung 1.1 dargestellten Fall unabhängig von der Startposition offensichtlich ein deutlich weiterer Weg zurückgelegt werden als bei einer Platzierung der Lagereinheiten wie in Abbildung 1.2. Dieser Weg muss entweder von einem Menschen oder einer Maschine zurückgelegt werden, deren Arbeit bezahlt werden muss.

Das Ziel, durch geschickte Einlagerungsstrategien die Kommissionierkosten zu minimieren, ist nicht neu und wird in der entsprechenden Fachliteratur schon seit Jahrzehnten diskutiert. Ein relativ detaillierter Überblick über die relevante Fachliteratur findet sich beispielsweise bei Glass [Gla08, S. 41-45]. Laut einer Studie der IWL AG [MN10] boten 2010 in der Folge bereits über 90% der untersuchten Lagerverwaltungssysteme Zonierungsstrategien an. Diese wurden Stand damals jedoch nur von etwas mehr als 50% der Kunden tatsächlich eingesetzt. [MN10, S. 10] Hinzu kommt, dass die Konfiguration der Einlagerungsstrategie eines Lagerverwaltungssystems seinen Betreibern häufig gar nicht präsent ist, obwohl der signifikante Einfluss dieser Strategien auf die Leistungsfähigkeit des Lagerverwaltungssystems weithin anerkannt ist. [Gla08, S. 39] Den Grund für diese Diskrepanz zwischen dem aktuellen Stand der Wissenschaft und dessen praktischer Berücksichtigung beim Einsatz solcher Steuerungsverfahren sieht Glass unter anderem in der mangelnden Verfügbarkeit der Fachliteratur und

in der unterschiedlichen Interpretation von Schnellläuferstrategien selbst unter Fachleuten. [Gla08, S. 39]

Auffällig ist außerdem, dass die Studie der IWL AG die Lagerverwaltungssysteme nur auf die Bereitstellung zonenbasierter Schnellläuferstrategien untersucht. Diese Strategien stellen die am weitesten verbreitete Form von Schnellläuferstrategien dar. Sie setzen die Klassifizierung von Artikeln anhand verschiedener Parameter voraus. Die dabei entstehenden Artikelklassen können dann wiederum den gegebenen Zonen zugeordnet werden.

Die weite Verbreitung dieser klassenbasierten Strategien ist laut Glass allerdings weniger in ihrer Überlegenheit als vielmehr historisch begründet. [Gla08, S. 45] Besonders bei Veränderungen im Sortiment oder starken Schwankungen bezüglich des Lagerfüllgrades haben diese Strategien erkennbare Schwächen. Mit Hilfe der heute verfügbaren Rechenleistung und den inzwischen erreichten Erkenntnissen können jedoch auch wesentlich differenziertere Strategien umgesetzt werden, die mit diesen Herausforderungen besser umgehen können. [Gla08, S. 51f.]

Aufgabe dieser Arbeit ist die Formulierung dieses Ziels als diskretes Optimierungsproblem; die Entwicklung, Implementierung und Bewertung von Algorithmen zu dessen Lösung sowie die empirische Untersuchung anhand historischer Daten.

1.2. Gliederung

Die Arbeit ist folgendermaßen gegliedert:

Kapitel 1 – Einleitung Im ersten Kapitel werden die Aufgabenstellung und das Ziel der Arbeit vorgestellt.

Kapitel 2 – Grundlagen Das zweite Kapitel beschreibt die Ausgangssituation und liefert eine kurze Zusammenfassung der Ergebnisse der Hintergrundrecherchen zum Thema.

Kapitel 3 – Konzept Kapitel drei befasst sich mit der Herangehensweise an die gestellte Aufgabe und mit der Entwicklung der Lösungsstrategien für das Optimierungsproblem.

Kapitel 4 – Implementierung Das vierte Kapitel beschreibt die Implementierung der diskutierten Optimierungsstrategien im Rahmen eines Smalltalk-Prototyps.

Kapitel 5 – Evaluation In Kapitel fünf werden die erzielten Ergebnisse bewertet und auf ihre praktische Verwendbarkeit analysiert.

Kapitel 6 – Zusammenfassung und Ausblick Kapitel sechs fasst die Ergebnisse der Arbeit zusammen und zeigt Anknüpfungspunkte für weiterführende Arbeiten auf.

Begriffserklärungen sind im Glossar aufgelistet.

2. Grundlagen

Für die langfristige Minimierung der Wege im Lager bietet die Fachliteratur verschiedene Strategien an, die im Folgenden kurz im Überblick vorgestellt werden.

2.1. Bekannte Schnellläuferstrategien

Gleichverteilungs-Strategie

Die Gleichverteilungs-Strategie (auch: Purely Randomized Assignment [CS91]) platziert alle Lagereinheiten zufällig auf den freien Plätzen. Dabei wird jeder freie Platz mit der gleichen Wahrscheinlichkeit gewählt (Gleichverteilung).

Diese Strategie ist keine echte Schnellläuferstrategie. Sie wird aber oft zum Vergleich und als Bewertungsgrundlage für andere Strategien herangezogen.

Kürzester-Weg-Strategie

Die Kürzester-Weg-Strategie (auch: Closest Open Location [CS91]) wählt für jede eingehende Lagereinheit den günstigsten freien Platz im Lager.

Diese Strategie schneidet bei geringen Lagerfüllgraden besser ab als die Gleichverteilungs-Strategie, weil weiter entfernte Plätze in diesem Fall kaum bis gar nicht genutzt werden. Die Kürzester-Weg-Strategie schneidet jedoch zunehmend schlechter ab je größer die Ungleichverteilung der Verweildauern im Sortiment. Besonders bei hohem Lagerfüllgrad kann sie nämlich dazu führen, dass lange liegende Lagereinheiten günstige Plätze über einen längeren Zeitraum blockieren. In dieser Zeit verlagert sich die Mehrheit der Lageroperationen auf den Bereich der teureren Plätze. Dadurch können Einbußen gegenüber der Gleichverteilungs-Strategie von bis zu 30% entstehen. [Gla08, S. 90]

n-Zonen-Strategie

Hierunter sind alle Strategien zusammengefasst, die das Lager in konstant viele Zonen unterteilen und die Lagereinheiten auf Basis ihrer Eigenschaften einer dieser Zonen zuordnen. Für die Platzwahl innerhalb der jeweiligen Zonen wird üblicherweise entweder die Gleichverteilungs- oder die Kürzester-Weg-Strategie eingesetzt.

Je nachdem, ob Lagereinheiten ausschließlich in der für sie vorgesehenen Zone gelagert werden dürfen (exklusive Zonenzuordnung), oder ob sie unter bestimmten Umständen auch eine Lagerung in anderen Zonen erlaubt ist (nicht-exklusive Zonenzuordnung), kann eine solche Strategie neben der Beeinflussung der Kommissionierkosten auch Einfluss auf die benötigte Gesamtkapazität des Lagers haben. Bei exklusiver Zonenzuordnung beispielsweise muss die Kapazität einer Zone jederzeit dem maximal darin zu lagernden Bestand an Lagereinheiten entsprechen. Nicht-exklusive Zonenzuordnung führt im Vergleich zu niedrigeren Kapazitätsanforderungen.

Praktisch liegt bei diesen Strategien eine der größten Herausforderungen in der Bestimmung der Zonengrenzen. [Gla08, S. 49]

Dynamische Zonierung

Die Strategie der dynamischen Zonierung ist das Ergebnis der Dissertation von Michael Glass. [Gla08] Nach dieser Strategie wird die Verweildauerverteilung der Lagereinheiten über die Gleichsetzung der jeweiligen Quantile so auf die Kommissionierkostenverteilung der Lagerplätze abgebildet, dass Schnelläufer auf günstigen und Langsamdreher auf teureren Plätzen platziert werden.

Anstatt konkrete Zonengrenzen festzulegen, nimmt die Strategie also eine implizierte Zonierung anhand der Verweildauerquantile vor und stellt somit eine Weiterentwicklung der n-Zonen-Strategien dar, die in der Lage ist, sich flexibel an veränderte Umstände anzupassen. Für eine genauere Beschreibung der Strategie siehe Kapitel 3.2.3.

2.2. Bezugsgröße: Artikel vs. Lagereinheiten

Fast alle Arbeiten zu Schnelläuferstrategien verfolgen eine artikelbasierte Betrachtungsweise, bei der für alle Lagereinheiten eines Artikels die gleiche Verweildauer angesetzt wird. [Gla08, S. 48] Ein Vorteil der Betrachtung auf Artekelebene ist, dass hier die Daten zur erwarteten Verweildauer üblicherweise bereits vorliegen, weil sie auch für die Ermittlung der optimalen Bestellmenge benötigt werden.

Glass jedoch argumentiert in seiner Arbeit, dass eine Betrachtung auf der Ebene von Lagereinheiten für verweildauerbasierte Strategien sinnvoller ist als eine artikelbasierte. Das begründet er mit den signifikanten Unterschieden zwischen den Verweildauern verschiedener Lagereinheiten eines Artikels. Dieser Unterschied zwischen Lagereinheiten desselben Artikels kann oft sogar größer sein als der Unterschied zwischen den durchschnittlichen Verweildauern unterschiedlicher Artikel. [Gla08, S. 53]

Der Vorschlag zu dieser Betrachtungsweise, bei der jeder Lagereinheit eine individuelle Verweildauer zugeordnet wird, stammt aus der Arbeit von Goetschalckx und Ratliff [GR90]. Sie beweisen, dass im perfekt balancierten Lager diejenige Strategie optimal ist – sowohl in Bezug auf die im Lager zurückgelegten Wege als auch in Bezug auf die nötige Lagerkapazität –

die für jede Periode $t = 1, 2, \dots$ den Lagereinheiten mit Verweildauer t die günstigsten freien Plätze zuweist. [GR90, S. 1123] Die Zuweisung der freien Plätze muss dabei aufsteigend nach Verweildauer erfolgen, beginnend mit dem kleinsten t .

Glass demonstriert die Überlegenheit der lagereinheitsbasierten Betrachtungsweise anhand verschiedener Simulationen und zeigt im direkten Vergleich, dass durch die Umstellung von der artikelbasierten auf die lagereinheitsbasierte Betrachtungsweise im Allgemeinen größere Verbesserungen erzielt werden können als durch die Einführung zusätzlicher Zonen oder durch eine kontinuierliche Betrachtung. [Gla08, S. 87f.]

Obwohl diese Art der Betrachtung aufgrund ihres hohen Detailgrads beispielsweise den Einsatz eines Referenzzeitraums für die Erstellung der Zukunftsprognosen erschwert, wird daher im Folgenden ebenfalls eine lagereinheitsbasierte Betrachtungsweise zugrunde gelegt.

3. Konzept

Die Behandlung der Lagerhaltung als diskretes Optimierungsproblem erfordert die Betrachtung des Problems aus algorithmischer Sicht. Um eine Lösung für das Optimierungsproblem finden zu können, muss also zuerst das Optimierungsproblem selbst konkretisiert und präzise formuliert werden.

Zunächst werden dafür die Parameter isoliert, die die Rahmenbedingungen des Problems bestimmen. Dann werden in einem ersten Schritt Lösungsansätze für das vereinfachte Szenario eines linearen Lagers entwickelt, also für ein Lager mit nur einem Regal. Abbildung 3.1 zeigt ein Beispiel eines solchen Lagers. Nachlieferungen werden in diesem Szenario nicht berücksichtigt. Im weiteren Vorgehen wird dann versucht, diese Ansätze auf zunehmend komplexere Problemstellungen zu übertragen, die durch sukzessive Anpassung der Parameter Schritt für Schritt an ein realistisches Szenario angenähert werden.

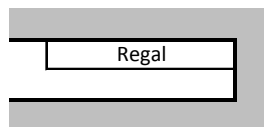


Abbildung 3.1.: Lineares Lager

Um eine Bewertungsgrundlage für die verschiedenen entwickelten Strategien zu schaffen, werden diese anschließend anhand empirischer Untersuchungen verglichen und auch der historischen Situation gegenübergestellt.

Im Rahmen der Untersuchungen wird zwischen statischer und dynamischer Optimierung unterschieden. Die statische Optimierung (Kapitel 3.2), auf der der Fokus der vorliegenden Arbeit liegt, befasst sich mit dem Optimierungsproblem unter der Annahme vollständiger Information.¹ Da die Ein- und Auslagerungszeitpunkte aller Lagereinheiten hier a priori bekannt sind, muss der Algorithmus nur ein einziges Mal auf den vorhandenen Daten ausgeführt werden, um das Problem für einen beliebigen Betrachtungszeitraum zu lösen.

Die dynamische Optimierung hingegen (Kapitel 3.3) geht davon aus, dass zukünftige Ereignisse nicht oder nur unzureichend vorhergesagt werden können.² Solche Strategien müssen daher darauf ausgelegt sein, flexibel auf Abweichungen von den Prognosen zu reagieren.

¹Dieser Begriff eines statischen Systems wird beispielsweise auch von Goetschalckx und Ratliff so verwendet. [GR90, S. 1125]

²Diese Systeme werden bei Goetschalckx und Ratliff als „adaptiv“ bezeichnet. [GR90, S. 1125]

Sie sind daher praxisnäher als die Strategien der statischen Optimierung. Die dynamische Optimierung wird in dieser Arbeit jedoch nur insofern behandelt, dass die in Kapitel 3.2 diskutierten Strategien auf ihre Einsetzbarkeit als dynamische Optimierungsstrategien untersucht werden.

3.1. Das Basisproblem

Formuliert als algorithmisches Problem lautet die zu lösende Aufgabe wie folgt:

Problembeschreibung Gegeben sei eine Menge $L = \{l_1, l_2, \dots, l_m\}$ von m Lagereinheiten sowie ein Lager $W = \{p_1, p_2, \dots, p_n\}$ mit n Plätzen. Des Weiteren sei gegeben eine Sequenz $\mathcal{B} = B_1, B_2, \dots, B_k$ von disjunkten Bestellungen $B_i \subseteq L$.

Gesucht ist eine injektive Funktion $\phi : L \rightarrow W$, welche jeder Lagereinheit einen Platz im Lager zuordnet, sodass die Kosten, alle Bestellungen zu bedienen, minimal sind.

Es gelte:

$$c(\mathcal{B}) = \sum_{i=1}^k c(B_i) \quad (3.1)$$

Die Kosten $c(B_i)$, die für die Kommissionierung einer Bestellung B_i anfallen, seien dabei definiert als die Summe der bei der Kommissionierung von B_i im Lager zurückgelegten Wege.

Entsprechend dieser Definition der Kostenfunktion sind vor allem Kommissionierstrategie und Lagerstruktur entscheidend für die Höhe der Kommissionierkosten. Andere Parameter, wie zum Beispiel die Kosten oder das Beschleunigungsverhalten etwaiger eingesetzter Maschinen, die den Preis des zurückgelegten Wegs bestimmen, stellen nur einen konstanten Faktor vor dem Ergebnis dar.

Bestimmende Parameter für die Strategien sind daher:

- Die Komplexität der Lagerstruktur (Aufbau)
- Die Wahl der Kommissionierstrategie (basierend auf Einzel-/Mehrfachspielen)
- Die angenommene Bestandsentwicklung (Lager mit/ohne Nachlieferungen)

Bezüglich der Lagerstruktur sind in der Praxis verschiedene Varianten zu finden. Die Abbildungen 3.2 bis 3.5 zeigen hierzu einige typische Beispiele. Dabei ist offensichtlich, dass sich die möglichen Wege im Lager zwischen den verschiedenen Lagertypen stark unterscheiden. Da der Lösungsansatz für das Optimierungsproblem jedoch maßgeblich von diesen möglichen Wegen und damit von der Lagerstruktur abhängt, müssen für jeden Lagerstrukturtyp eigene Untersuchungen angestellt werden.

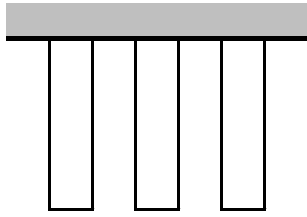


Abbildung 3.2.: Typ 1a: Einseitig begehbare Gassen

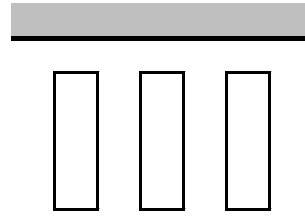


Abbildung 3.3.: Typ 1b: Beidseitig begehbare Gassen

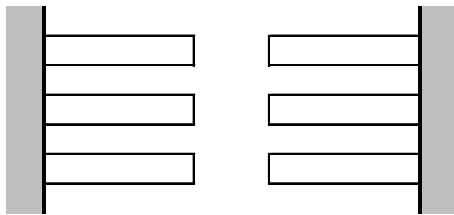


Abbildung 3.4.: Typ 2a: Einseitig begehbare Gassen

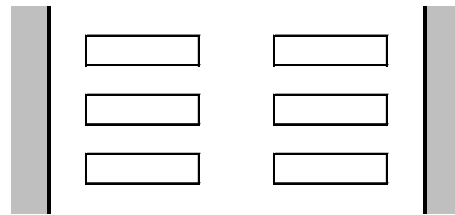


Abbildung 3.5.: Typ 2b: Beidseitig begehbare Gassen

Die vorliegende Arbeit orientiert sich in ihren Annahmen an dem Lager, aus dem die historischen Daten stammen, die als Input für die Algorithmen und zum historischen Vergleich herangezogen werden. Dieses Lager ist vom Typ 1a. Die folgenden Lösungsansätze beschränken sich daher ebenfalls auf Lager mit einseitig begehbaren Gassen.

Weiter wird angenommen, dass die Differenz zwischen den Kommissionierkosten für zwei benachbarte Plätze p_i und p_j mit $c(p_i) < c(p_j)$ innerhalb des Lagers immer gleich sei. Normiert gilt daher

$$c(p_j) - c(p_i) = 1 \quad (3.2)$$

Gleiches gelte für den Weg vom Kommissionierplatz zum ersten Platz in jeder Gasse.

Im Folgenden werden verschiedene Varianten des Basisproblems untersucht. Dabei werden die oben genannten Annahmen zur Lagerstruktur als gegeben vorausgesetzt und das Szenario, beginnend mit einem linearen Lager ohne Nachlieferungen, sukzessive um folgende Parameter erweitert:

- Mehrfachspiele (z.B. gemeinsame Ausfassung aller Lagereinheiten einer Bestellung)
- Komplexere Lagerstruktur
- Nachlieferungen

In den dabei aufgeführten Beispielen sind die Plätze innerhalb einer Gasse immer aufsteigend nach ihrer Entfernung zum Kommissionierplatz nummeriert. Für diese Beispiele gelte daher, dass die Kommissionierkosten eines Lagerplatzes seiner Nummer entsprechen.

3.2. Statische Optimierung

Die statische Optimierung setzt im Gegensatz zur dynamischen Optimierung (Kapitel 3.3) vollständige Information voraus. Da die Ein- und Auslagerungszeitpunkte aller Lagereinheiten aus dem Betrachtungszeitraum hier a priori bekannt sind, kann eine Lösungsstrategie diese Informationen über zukünftige Lagerbewegungen bereits von vorn herein berücksichtigen. Dieser Wissensvorsprung ist vor allem in einem Lager mit Nachlieferungen interessant (siehe hierzu Kapitel 3.2.3).

Bei einer auf Einzelspielen basierenden Kommissionierstrategie wird jede Lagereinheit einzeln kommissioniert. Die Kosten, um eine Bestellung B_i unter einer gegebenen Platzierung ϕ zu bedienen, sind daher in diesem Fall

$$c(B_i) := \sum_{l \in B_i} c(\phi(l)) \quad (3.3)$$

Unter Annahme der Kommissionierung aus einem Lager ohne Nachlieferungen, bei dem jede Lagereinheit Teil einer Bestellung ist, ist die Lösung für eine solche Strategie trivial: Jeder Platz im Lager muss genau einmal besucht werden; zu welcher Bestellung die dort liegende Lagereinheit gehört, ist irrelevant. In einem voll belegten Lager wäre selbst eine völlig zufällige Platzierung eine optimale Lösung.

Für die Problemvarianten ohne Nachlieferungen wird daher im Folgenden nur der Fall von Mehrfachspiel-basierten Kommissionierstrategien betrachtet. Für die Variante mit Nachlieferungen beginnen die Überlegungen bei Einzelspiel-basierten Strategien.

Jeder Problemvariante ist ein eigenes Unterkapitel gewidmet, das mit der Problembeschreibung beginnt und anschließend die erarbeiteten Lösungsstrategien diskutiert. Für die Varianten im realistischen Lager wird außerdem eine empirische Untersuchung mit Vergleich zu historischen Daten durchgeführt.

3.2.1. Lineares Lager ohne Nachlieferungen

Problembeschreibung Gegeben sei ein lineares Lager mit einer Kommissionierstrategie basierend auf Mehrfachspielen: Alle Lagereinheiten einer Bestellung werden gemeinsam kommissioniert. Die Kosten, um eine Bestellung B_i unter einer gegebenen Platzierung ϕ zu bedienen, seien daher

$$c(B_i) := \max\{c(\phi(l)) : l \in B_i\} \quad (3.4)$$

Lösungsansätze Am besten veranschaulicht man sich die Problemstellung zunächst anhand eines Beispiels.

Beispiel:

Gegeben sei ein Lager mit $n = 6$ Plätzen. Die Kardinalität der Bestellungen A bis C sei gegeben wie folgt: $|A| = 3$, $|B| = 2$, $|C| = 1$. Dabei sei die Kardinalität $|B_i|$ einer

Bestellung B_i definiert als die Anzahl der Lagereinheiten, die dieser Bestellung zugeordnet sind. Die folgende Tabelle zeigt die Platzierung der zu A , B und C gehörigen Lagereinheiten:

Platz	1	2	3	4	5	6
Regal 1	A	B	A	A	B	C

$$c(A) = 4, c(B) = 5, c(C) = 6, c(B) = 15$$

Durch die „eingeschobene“ Lagereinheit von Bestellung B verteuert sich die Kommissionierung von Bestellung A . Bei gemeinsamer Lagerung der zusammengehörigen Lagereinheiten wird das Ergebnis schon besser:

Platz	1	2	3	4	5	6
Regal 1	A	A	A	B	B	C

$$c(A) = 3, c(B) = 5, c(C) = 6, c(B) = 14$$

Werden die Lagereinheiten jeder Bestellung zusammenhängend platziert, entsprechen die Kosten der Kommissionierung einer Bestellung B_i gemäß (3.2) und (3.4) also den Kosten der Kommissionierung ihrer Vorgängerbestellung B_j zuzüglich der Kardinalität $|B_i|$ der Bestellung selbst:

$$c(B_i) = c(B_j) + |B_i| \quad (3.5)$$

wobei

$$c(B_j) \geq 0 \quad (3.6)$$

Theoretisch kommt außerdem die Anzahl eventuell dazwischen frei gelassener Plätze hinzu. Da das Freilassen von Lücken jedoch ganz offensichtlich nicht zu einer Reduktion der Kosten beitragen kann, wird dieser Summand nicht berücksichtigt.

Da die Kardinalität der Bestellung nicht beeinflussbar ist, muss also das Ziel sein, jeweils die Kosten der Vorgängerbestellung zu minimieren.

Lemma 1. *Die optimale Lösung wird erreicht, indem man die Bestellungen aufsteigend der Größe nach sortiert und das Lager entsprechend der Reihenfolge der sortierten Bestellungen von vorne her auffüllt, beginnend mit den Lagereinheiten der kleinsten Bestellung. (Beweis in Anhang A.1.1.)*

Werden nicht alle Plätze im Lager benötigt, bleiben diese automatisch hinten übrig.

Zwischen gleich großen Bestellungen gibt es offensichtlich keine Größendifferenz; ihre Positionen in der Reihenfolge von Bestellungen können vertauscht werden, ohne dass sich eine Kostenänderung ergibt. In diesem Fall ist die optimale Lösung daher nicht eindeutig.

3.2.2. Realistisches Lager ohne Nachlieferungen

Problembeschreibung Gegeben sei ein Lager mit mehreren Gassen. Die Kommissionierstrategie sei die gleiche wie im linearen Fall. Die Kosten für die Kommissionierung einer Bestellung, deren zugehörige Lagereinheiten alle in der gleichen Gasse liegen, entsprechen daher den durch Gleichung (3.4) gegebenen Kosten.

Lösungsansätze Das oben definierte Lager stellt im Grunde ein Lager dar, das sich aus mehreren linearen Lagern zusammensetzt. Intuitiv bietet sich daher eine Übertragung der für den linearen Fall entwickelten Lösung an.

Sortier-Strategie von vorne

Diese Strategie sortiert die Bestellungen nach ihrer Kardinalität und befüllt das Lager von vorne, beginnend mit dem günstigsten Platz und der kleinsten Bestellung. Diese Strategie liefert jedoch keine optimale Lösung.

Wie oben angedeutet kann es wegen der Aufteilung des Lagers auf mehrere Gassen passieren, dass nicht alle zu einer Bestellung gehörigen Lagereinheiten in derselben Gasse untergebracht werden können. Je mehr Bestellungen bereits platziert sind, desto weniger Plätze sind am Ende jeder Gasse noch frei. Gleichzeitig werden die zu platzierenden Bestellungen zunehmend länger.

Im Fall des Splits einer Bestellung B_i in t Teile sind die Kommissionierkosten unter Umständen aber deutlich höher:

$$c(B_i) = \sum_{k=1}^t c(B_{i_k}) \quad (3.7)$$

wobei

$$c(B_{i_k}) \geq |B_{i_k}| \quad (3.8)$$

Beispiel:

Gegeben sei ein Lager mit $n = 8$ Plätzen, verteilt auf zwei Gassen. Die Kardinalität der Bestellungen A bis D sei gegeben wie folgt: $|A| = |B| = 1$, $|C| = 2$, $|D| = 4$. Bei der Platzierung ergibt sich ein Split der Bestellung D in die Teile D_1 und D_2 .

Sortier-Strategie von vorne:

Platz	1	2	3	4
Gasse 1	B	D_1	D_1	D_1
Gasse 2	A	C	C	D_2

$$c(\mathcal{B}) = 13$$

Nach dem Split widerspricht das Ergebnis den Vorgaben der Strategie selbst. Diese verlangt, dass alle Bestellungen innerhalb einer Gasse der Kardinalität nach sortiert sein müssen. Um das zu beheben, müssen die Gassen mit den Teilbestellungen gegebenenfalls neu sortiert werden. Diese nachträgliche Sortierung kann das Ergebnis entsprechend der Argumentation aus Kapitel 3.2.1 nie verschlechtern. Das Ersetzen der gesplitteten Bestellung durch ihre Teile und eine Wiederholung des Einlagerungsvorgangs unter Berücksichtigung der beiden „neuen“ Bestellungen könnte hingegen weitere Splits erfordern.

Im obigen Fall würden durch die nachträgliche Sortierung in der zweiten Gasse die Bestellungen C und D_2 vertauscht und so die Gesamtkosten auf $c(\mathcal{B}) = 12$ gesenkt. Damit liegen die Kosten jedoch immer noch über dem Optimum.

Ergebnis nach Neusortierung:

Platz	1	2	3	4
Gasse 1	B	D_1	D_1	D_1
Gasse 2	A	D_2	C	C

$$c(\mathcal{B}) = 12$$

Optimal wäre:

Platz	1	2	3	4
Gasse 1	D	D	D	D
Gasse 2	A	B	C	C

$$c(\mathcal{B}) = 11$$

Die Sortier-Strategie mit Einlagerung von vorne kann also selbst dann zu einem Split führen, wenn es eine Lösung ohne Split gäbe. Dies gilt auch im nicht voll belegten Lager.

Beispiel:

Gegeben sei ein Lager mit $n = 8$ Plätzen, verteilt auf zwei Gassen. Die Kardinalität der Bestellungen A bis C sei gegeben wie folgt: $|A| = |B| = 2$, $|C| = 3$.

Sortier-Strategie von vorne:

Platz	1	2	3	4
Gasse 1	A	A	C	C
Gasse 2	B	B	C	-

$$c(\mathcal{B}) = 11$$

Optimal wäre:

Platz	1	2	3	4
Gasse 1	C	C	C	-
Gasse 2	A	A	B	B

$$c(\mathcal{B}) = 9$$

Das generelle Erzwingen möglichst langer zusammenhängender Sequenzen von Lager-einheiten der gleichen Bestellung bei der Platzierung löst dieses Problem jedoch nicht. So wäre im folgenden Fall, in dem sich ein Split nicht vollständig vermeiden lässt, eine frühere Trennung günstiger.

Beispiel:

Gegeben sei ein Lager mit $n = 8$ Plätzen, verteilt auf zwei Gassen. Die Kardinalität der Bestellungen A bis D sei gegeben wie folgt: $|A| = |B| = |C| = 1$, $|D| = 5$.

3. Konzept

Längstmögliche Sequenz:

Platz	1	2	3	4
Gasse 1	D	D	D	D
Gasse 2	A	B	C	D

$$c(\mathcal{B}) = 14$$

Optimal wäre:

Platz	1	2	3	4
Gasse 1	B	D	D	D
Gasse 2	A	C	D	D

$$c(\mathcal{B}) = 12$$

Das gleiche Ergebnis wie links erhält man, wenn man das Lager von hinten befüllt, beginnend mit dem teuersten Platz und der größten Bestellung. Auch dieser Ansatz führt damit in einem solchen Fall offensichtlich nicht zur optimalen Lösung.

Die Ergebnisse der Untersuchungen zu den Fällen mit Split legen die Vermutung nahe, dass ein möglichst ausgeglichener Split erstrebenswert ist, also eine ungefähre Halbierung der zu splittenden Bestellung. Es reicht jedoch nicht, immer nur die größte Bestellung für den Split in Erwägung zu ziehen.

Beispiel:

Gegeben sei ein Lager mit $n = 10$ Plätzen, verteilt auf zwei Gassen. Die Kardinalität der Bestellungen A bis C sei gegeben wie folgt: $|A| = |B| = 3$, $|C| = 4$. Ein Split ist daher nicht vermeidbar.

Split größte Bestellung:

Platz	1	2	3	4	5
Gasse 1	C	C	A	A	A
Gasse 2	C	C	B	B	B

$$c(\mathcal{B}) = 14$$

Optimal wäre:

Platz	1	2	3	4	5
Gasse 1	A	A	B	B	B
Gasse 2	A	C	C	C	C

$$c(\mathcal{B}) = 13$$

Dennoch gilt :

Lemma 2. *In einem Lager mit zwei Gassen ist eine Lösung mit einem Split immer echt besser als eine Lösung mit zwei Splits. (Beweis in Anhang A.1.2.)*

In einem größeren Lager sollte also über jedes Paar von Gassen maximal eine Bestellung gesplittet werden. Bei g Gassen wären das insgesamt maximal $\binom{g}{2}$ Splits.

Sortier-Strategie von hinten

Beginnt man mit der größten Bestellung und befüllt das Lager von hinten, so werden die zu platzierenden Bestellungen zunehmend kleiner. Wird bei der Platzierung eine Bestellung gesplittet, ergibt sich hier im Gegensatz zur Platzierung von vorne kein Bedarf für eine nachträgliche Reorganisation.

Lemma 3. *Der verbleibende Teil einer gesplitteten Bestellung kann wie eine eigene Bestellung betrachtet und entsprechend seiner Kardinalität platziert werden, ohne dass sich die Lösung dadurch verschlechtert. (Beweis in Anhang A.1.3.)*

Wie oben gezeigt führt die Einlagerung von hinten jedoch unter Umständen zu ungünstigen Split-Situationen.

Ist das Lager nicht voll belegt, würde man intuitiv einfach die teuersten Plätze sperren und erst davor mit der Einlagerung beginnen. Das kann aber dazu führen, dass Splits erzwungen werden, wo keine nötig wären. Stattdessen ist es sinnvoller, das Ergebnis bei Einlagerung von hinten zu berechnen und am Ende nach vorne „aufzurutschen“, also vorne frei gebliebene Plätze ans Ende der Gasse zu verschieben.

Beispiel:

Gegeben sei ein Lager mit $n = 8$ Plätzen, verteilt auf zwei Gassen. Die Kardinalität der Bestellungen A bis C sei gegeben wie folgt: $|A| = |B| = 1$, $|C| = 4$.

Sperrung der teuersten Plätze:

Platz	1	2	3	4
Gasse 1	C	C	C	-
Gasse 2	A	B	C	-

$$c(\mathcal{B}) = 9$$

Optimal wäre:

Platz	1	2	3	4
Gasse 1	C	C	C	C
Gasse 2	A	B	-	-

$$c(\mathcal{B}) = 7$$

Trotzdem führt offensichtlich keine der beiden Varianten der Sortier-Strategie in jedem Fall zur optimalen Lösung.

Lösung per dynamischer Programmierung

Um die tatsächlich optimale Lösung zu berechnen, würde man naiv einfach alle Möglichkeiten durchprobieren, bei denen keine oder nur eine Bestellung gesplittet wird, und daraus die günstigste Lösung auswählen. Bei g Gassen und k Bestellungen wären das g^k Möglichkeiten. Gibt es keine solche Lösung, müsste als nächstes nach einer Lösung mit zwei Splits gesucht werden, usw. Diese Vorgehensweise würde sowohl für voll belegte als auch nicht voll belegte Lager die optimale Lösung ermitteln. Die Laufzeitkomplexität allerdings läge bei $\mathcal{O}(g^k)$.

Legt man eine Rechenleistung von 10^9 Instruktionen (= 1GHz) zugrunde, würde das bereits bei 2^{50} Möglichkeiten mehrere Tage dauern – vorausgesetzt, dass die Kosten einer Möglichkeit tatsächlich in einer Instruktion berechnet werden können. Geschickter ist deshalb der Einsatz eines dynamischen Programms, das bereits berechnete Ergebnisse speichert, um sie wiederverwendbar zu machen und damit weitere Berechnungen zu beschleunigen.

Abbildung 3.6 zeigt die Tabelle für das dynamische Programm für ein Lager mit zwei Gassen. Jeder Eintrag $\square_{i,j}$ in der Tabelle entspricht den minimalen Kosten für die Platzierung der Bestellungen $B_1, \dots, B_i$ unter der Bedingung, dass in der ersten Gasse j Plätze belegt sind.

		j				
		1	2	...	n-1	n
i	1					
	2					
	...					
	k-1					
	k					

Abbildung 3.6.: Tabelle für ein Lager mit zwei Gassen

Sei $c(B_i, j, g)$ der Preis für die Platzierung der Bestellung B_i ab dem j -ten Platz in Gasse g . Dann gilt für die obige Tabelle

$$\square_{i,j} = \min(\square_{i-1,j-|B_i|} + c(B_i, j - |B_i| + 1, 1), \square_{i-1,j} + c(B_i, \sum_{m=1}^{i-1} |B_m| - j + 1, 2)) \quad (3.9)$$

Für jede weitere Gasse kommt in der Tabelle eine weitere Dimension hinzu. Mit Hilfe des dynamischen Programms kann die Laufzeitkomplexität des Algorithmus also von $\mathcal{O}(g^k)$ auf $\mathcal{O}(n^g \cdot k)$ reduziert werden, wobei n die Anzahl der Plätze in einer der Gassen darstellt.

Unter Annahme der oben genannten Voraussetzungen lässt sich ein Fall mit zwei Gassen à 50 Plätzen und 50 Bestellungen in weniger als einer Sekunde berechnen; schon bei sechs Gassen und 150 Bestellungen läge man aber bei über einer halben Stunde. Hinzu kommt, dass die gesamte Tabelle für jede Split-Möglichkeit erneut berechnet werden muss. Da die alleinige Betrachtung eines Splits der größten Bestellung nicht ausreicht, gibt es für x Splits etwa $\binom{|L|}{x}$ Möglichkeiten, wobei $|L|$ die Anzahl der zu platzierenden Lagereinheiten ist.

Ein Ansatz zur weiteren Reduktion der Rechenzeit wäre eine hierarchische Lösung des Problems mit Hilfe des dynamischen Programms. Dazu werden die Gassen zunächst in zwei Gruppen aufgeteilt und die Bestellungen mit Hilfe des Algorithmus auf diese beiden Gruppen verteilt. Daraufhin werden die Gruppen halbiert und die der Gruppe zugeordneten Bestellungen wieder mit Hilfe des Algorithmus auf die neuen Untergruppen verteilt. So fährt man fort, bis jede Gruppe nur noch aus einer Gasse besteht und nicht mehr weiter geteilt werden kann. Dadurch kann die Laufzeit weiter auf $\mathcal{O}(kN \log g)$ gedrückt werden, wobei N die Gesamtzahl der Plätze im betrachteten Lager repräsentiert.

3.2.3. Realistisches Lager mit Nachlieferungen

Im Lager mit Nachlieferungen ist das oberste Ziel der Lösung weniger die Minimierung der Kommissionierkosten für den Bestand zum Zeitpunkt der Optimierung, als vielmehr das Minimieren der Kommissionierkosten auf lange Sicht. Es kann sich daher lohnen, kurzzeitig

suboptimale Zustände zu akzeptieren, wenn dadurch langfristig ein besseres Gesamtergebnis erzielt werden kann.

Problembeschreibung Gegeben sei ein Lager mit mehreren Gassen. Die Kommissionierstrategie basiere auf Einzelspielen; die Kosten für die Kommissionierung einer Bestellung B_i seien also wie durch Gleichung (3.3) gegeben.

Lösungsansätze Die Kommissionierkosten hängen in erster Linie von den im Lager zurückgelegten Wegen ab. Um die Kosten auf lange Sicht zu minimieren, muss es also Ziel jeder Lösung sein, dass günstige Plätze im Lager langfristig häufiger besucht werden als teure, sodass der insgesamt im Lager zurückgelegte Weg minimiert wird.

Dynamische Zonierung

Die in Kapitel 2.1 vorgestellte Strategie der dynamischen Zonierung versucht dieses Problem zu lösen, indem sie dafür sorgt, dass günstige Plätze stärker frequentiert werden als teure. Dazu sortiert sie die Menge der Lagereinheiten nach ihren Verweildauern und die Menge der Lagerplätze nach ihren Kommissionierkosten. Daraufhin werden die beiden Listen normiert und die Quantile gleichgesetzt, sodass die $\alpha\%$ Lagereinheiten mit der kürzesten Verweildauer innerhalb der $\alpha\%$ günstigsten Lagerplätze platziert werden.

Um das Optimierungspotenzial bestmöglich ausschöpfen zu können, ist die möglichst exakte Bestimmung der Verweildauerverteilung ausschlaggebend. Entsprechend der in Kapitel 2.2 vorgestellten Argumentation betrachtet die Strategie daher jede Lagereinheit individuell.

Für jede Lagereinheit wird also zunächst ihre Verweildauer bestimmt und dann anhand der Verweildauer ihr Index j in der Liste der Lagereinheiten. Dieser Index wird mit Hilfe einer Normierung auf einen Index k in der Liste der Lagerplätze abgebildet. Abbildung 3.7 zeigt eine schematische Darstellung dieses Vorgehens.

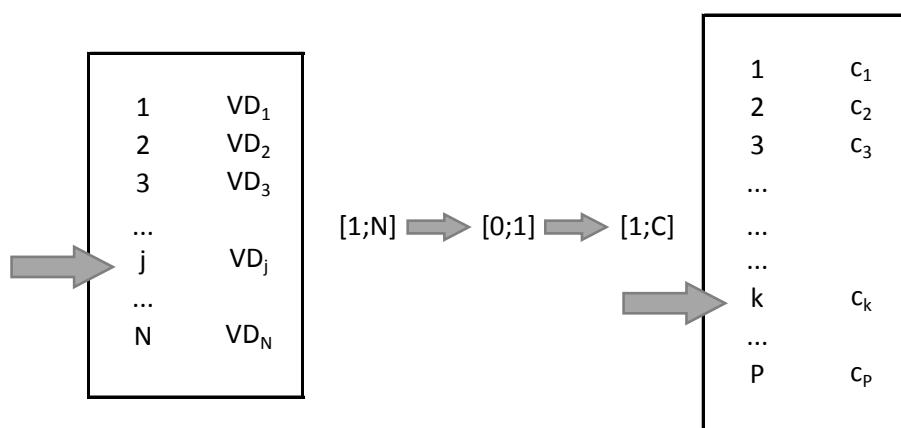


Abbildung 3.7.: Mapping Lagereinheit auf Lagerplatz
In Anlehnung an: [Gla08, S. 58]

3. Konzept

Die tatsächliche Auslastung günstiger Plätze ist jedoch nur im perfekt balancierten Lager optimal, da aufgrund des Mappings der Quantile automatisch Pufferzonen entstehen.

Beispiel:

Gegeben sei ein lineares Lager mit $n = 4$ Plätzen sowie zwei Lagereinheiten A und B mit den Verweildauern $VD_A = 2$ und $VD_B = 4$. Durch das Mapping der Quantile wird Lagereinheit A in der ersten Hälfte des Lagers platziert und Lagereinheit B in der zweiten.

Ergebnis:

Platz	1	2	3	4
Gasse 1	A		B	

$$c(\mathcal{B}) = 4$$

Optimal wäre:

Platz	1	2	3	4
Gasse 1	A	B		

$$c(\mathcal{B}) = 3$$

Je größer die Fluktuation des Bestands, desto häufiger entstehen solche Situationen wie oben beschrieben, weil der durchschnittliche Bestand dann relativ weit unterhalb der Lagerkapazität liegt, und desto schlechter wird das mit dieser Strategie erzielte Ergebnis. [Gla08, S. 85f] In der Realität muss also ein hoher durchschnittlicher Lagerfüllgrad vorherrschen, um ein gutes Ergebnis zu erreichen.

Job-Scheduling-Strategie³

Unter der Annahme vollständiger Information ist das Vorhalten von Pufferzonen nicht nötig, da zu jedem Zeitpunkt bekannt ist, wie viele Lagereinheiten auf Lager sind und wie lange diese jeweils im Lager bleiben. Die Job-Scheduling-Strategie nutzt dieses Wissen, um eine höchstmögliche Frequentierung der günstigen Plätze zu erzielen.

Dazu werden alle Lagereinheiten nach dem Zeitpunkt ihrer Kommissionierung sortiert. Beginnend mit dem günstigsten Platz wird nun Platz für Platz die maximale Menge von Lagereinheiten gesucht, die im gegebenen Zeitraum auf diesem Platz gelagert werden können.

Aus der Menge der noch nicht platzierten Lagereinheiten wird deshalb die Lagereinheit mit dem frühesten Kommissionierdatum und der längsten Verweildauer gewählt und dem aktuell bearbeiteten Platz zugewiesen. Alle Lagereinheiten, deren Lagerzeitraum sich mit dem der gewählten Lagereinheit überschneidet, werden verworfen, weil sie ja nicht auf demselben Platz gelagert werden können wie die soeben gewählte Lagereinheit. Dieses Vorgehen wird so lange wiederholt, bis alle Lagereinheiten entweder platziert oder verworfen sind. Dann kommt der nächste Platz an die Reihe.

³Benannt in Anlehnung an [Ski08, S. 9]

Je günstiger der Lagerplatz, desto mehr Lagereinheiten werden demnach darauf platziert. Dadurch erzielt diese Strategie unter Berücksichtigung der bereits platzierten Lagereinheiten nachweislich die höchstmögliche Frequentierung für jeden Platz. (Beweis in Anhang A.1.4.) Offensichtlich ist die Strategie dabei vom Lagerfüllgrad unabhängig.

Die Wahl der Lagereinheit mit der längsten Verweildauer verhindert negative Auswirkungen auf den teureren Plätzen.

Beispiel:

Die Lagerzeiträume der Lagereinheiten A bis D seien gegeben wie folgt:

Tag 1	A			
Tag 2	A	B		
Tag 3		B		
Tag 4		B	C	
Tag 5			C	D
Tag 6			C	D

Lagereinheit A wird also an Tag 1 eingelagert, ist an Tag 1 und 2 auf Lager und wird nach Tag 2 ausgelagert. Lagereinheit B wird an Tag 2 eingelagert, ist an Tag 2, 3 und 4 auf Lager und wird nach Tag 4 ausgelagert, usw.

Nur unter Beachtung des Verweildauerkriteriums erzielt der Algorithmus hier das optimale Ergebnis.

Kürzeste Verweildauer zuerst:

Platz	1	2	3
Tag 1	A		
Tag 2	A	B	
Tag 3		B	
Tag 4		B	C
Tag 5	D		C
Tag 6	D		C

$$c(\mathcal{B}) = 2 \cdot 1 + 2 + 3 = 7$$

Längste Verweildauer zuerst:

Platz	1	2	3
Tag 1	A		
Tag 2	A	B	
Tag 3		B	
Tag 4	C	B	
Tag 5	C	D	
Tag 6	C	D	

$$c(\mathcal{B}) = 2 \cdot 1 + 2 \cdot 2 = 6$$

Abgesehen von der Berücksichtigung der Verweildauer entspricht der Teil des Algorithmus, der für jeden Platz ausgeführt wird, damit dem Algorithmus zur Bestimmung der größten stabilen Menge⁴ auf einem Intervallgraphen. Im Intervallgraph repräsentiert jeder Knoten ein Intervall. Jede Kante zwischen zwei Knoten bedeutet eine Überschneidung der beiden Intervalle. Die größte stabile Menge ist die größte Menge von Knoten, zwischen denen keine gemeinsame Kante besteht, also die größtmögliche Menge von Intervallen, die sich nicht überschneiden.

⁴engl.: Maximum Independent Set

3. Konzept

Die Berechnung der größten stabilen Menge auf einem Intervallgraphen läuft in $\mathcal{O}(n \log n)$, wobei n in diesem Fall die Anzahl der Intervalle bezeichnet. Die gewichtete Variante dieses Algorithmus ist in Kapitel 4.3.1 beschrieben.

Gibt es mehrere größte stabile Mengen, muss jedoch nicht jede davon zur optimalen Lösung führen.

Beispiel:

Die Lagerzeiträume der Lagereinheiten A bis D seien gegeben wie folgt:

Tag 1	A		
Tag 2	A	B	
Tag 3		B	D
Tag 4			C D
Tag 5			C D
Tag 6			D

Weil der Job-Scheduling-Algorithmus immer die Lagereinheit mit dem frühesten Kommissionierzeitpunkt wählt, erzielt er hier nicht das optimale Ergebnis.

Ergebnis:

Platz	1	2	3
Tag 1	A		
Tag 2	A	B	
Tag 3		B	D
Tag 4	C		D
Tag 5	C		D
Tag 6			D

$$c(\mathcal{B}) = 2 \cdot 1 + 2 + 3 = 7$$

Optimal wäre:

Platz	1	2	3
Tag 1	A		
Tag 2	A	B	
Tag 3	D	B	
Tag 4	D	C	
Tag 5	D	C	
Tag 6	D		

$$c(\mathcal{B}) = 2 \cdot 1 + 2 \cdot 2 = 6$$

Sowohl hier als auch im vorigen Beispiel führte also die Minimierung der insgesamt verwendeten Plätze zum Ziel. Die Verteilung der Lagereinheiten auf die minimale Anzahl von Plätzen könnte leicht mit Hilfe eines Färbe-Algorithmus für Intervallgraphen berechnet werden (beschrieben in Kapitel 4.3.2).

Dieser Ansatz ist aber auch keine Lösung, weil dabei die höchstmögliche Frequentierung der günstigen Plätze nicht mehr gewährleistet ist.

Beispiel:

Die Lagerzeiträume der Lagereinheiten A bis H seien gegeben wie folgt:

Tag 1	A		D				
Tag 2		B		D			
Tag 3			C	D			
Tag 4				D	E		
Tag 5					E	F	
Tag 6					E		G
Tag 7					E		H

In einem solchen Fall wäre die Minimierung der Plätze teurer als die Wahl der größten stabilen Menge.

Minimum belegter Plätze:

Platz	1	2	3
Tag 1	A	D	
Tag 2	B	D	
Tag 3	C	D	
Tag 4	E	D	
Tag 5	E	F	
Tag 6	E	G	
Tag 7	E	H	

$$c(\mathcal{B}) = 4 \cdot 1 + 4 \cdot 2 = 12$$

Optimal wäre:

Platz	1	2	3
Tag 1	A	D	
Tag 2	B	D	
Tag 3	C	D	
Tag 4		D	E
Tag 5	F		E
Tag 6	G		E
Tag 7	H		E

$$c(\mathcal{B}) = 6 \cdot 1 + 1 \cdot 2 + 1 \cdot 3 = 11$$

Klar ist jedoch:

Lemma 4. *In einer optimalen Lösung bildet für jeden Platz die Menge der darauf gelagerten Lagereinheiten eine maximale stabile Menge bezüglich der Lagereinheiten, die auf diesem und auf teureren Plätzen gelagert werden. (Beweis in Anhang A.1.5.)*

3.2.4. Strategische Parameter

In der Praxis sind außerdem häufig spezifische strategische Parameter vorgegeben, die sich ebenfalls auf die Leistungsfähigkeit von Optimierungsstrategien auswirken können. Im Folgenden werden die entwickelten Strategien daher auf ihre Anpassungsfähigkeit bezüglich besonders häufiger Vorgaben untersucht.

Mehrfachspiele

Wie in Kapitel 3.2.2 gezeigt ist es für eine auf Mehrfachspielen basierende Kommissionierstrategie am besten, wenn die Lagereinheiten, die in einem Kommissioniervorgang eingesammelt werden, in derselben Gasse und dort so nah wie möglich beieinander liegen. Damit stellt eine solche Strategie völlig andere Anforderungen an die Optimierung als eine auf Einzelspielen

basierende Strategie, für die solche Zusammenhänge vollkommen uninteressant sind, weil jeder Kommissioniervorgang immer nur eine einzige Lagereinheit umfasst.

Unter Einsatz der dynamischen Zonierung ist es nicht möglich, Lagereinheiten mit signifikant unterschiedlicher Verweildauer nahe beieinander zu lagern, ohne dass dadurch der Optimierungseffekt der Strategie gefährdet wird. Die Platzierung einer Lagereinheit mit hoher Verweildauer auf einem günstigen Platz könnte viele kürzer liegende Lagereinheiten auf teurere Plätze verdrängen und dadurch den Effekt der gemeinsamen Lagerung schnell zunichte machen. Die Platzierung einer Lagereinheit mit niedriger Verweildauer auf einem teureren Platz hätte zwar weniger zahlreiche Nebenwirkungen, kann durch den weiteren Weg bei der Einlagerung aber trotzdem mehr Kosten verursachen als durch die gemeinsame Kommissionierung gespart wird.

Bei Anwendung der Job-Scheduling-Strategie ist eine gemeinsame Platzierung ohne Auswirkungen auf andere Lagereinheiten prinzipiell möglich, sofern ein nahe gelegener Platz im gegebenen Zeitraum noch nicht belegt ist. Diese Strategie optimiert durch die Wahl der am längsten liegenden Lagereinheit aber nicht nur die Frequentierung günstiger Plätze, sondern auch deren Auslastung. Dadurch ist das Vorhandensein einer solchen Lücke zunehmend unwahrscheinlich, je höher die Verweildauer der zu verschiebenden Lagereinheit ist, und je günstiger die anvisierte Zielregion. Hinzu kommt natürlich auch hier die Differenz bei den Einlagerungskosten, die durch die Einsparungen bei den Kommissionierkosten kompensiert werden muss.

In beiden Fällen ist also die Berücksichtigung von Mehrfachspielen auf Basis vordefinierter Zusammenhänge (z.B. Zugehörigkeit zur selben Bestellung) nicht so recht mit der Optimierungsstrategie vereinbar. Einfacher wäre es, zusätzliches Optimierungspotenzial zu heben, indem man die Touren der Kommissionierer so anpasst, dass möglichst viele der gleichzeitig zu kommissionierenden Lagereinheiten in einer Tour eingesammelt werden können.

Berücksichtigung von Verbrauchsfolgekriterien

Die angewandten Verbrauchsfolgekriterien definieren den prognostizierten Kommissionierzeitpunkt für jede Lagereinheit. Dienen Daten aus der Vergangenheit als Zukunftsprognose, wie das zum Beispiel beim Zugrundelegen eines Referenzzeitraums der Fall wäre, und wurden in der Vergangenheit die gleichen oder strengere Verbrauchsfolgekriterien angewandt wie sie für die Zukunft gefordert sind, können die Daten aus dem Referenzzeitraum diesbezüglich ohne Anpassung übernommen werden. Die Auslagerungszeitpunkte der im Referenzzeitraum betrachteten Lagereinheiten resultieren direkt aus der Anwendung dieser Kriterien und werden durch die Reorganisation nicht verändert, daher entspricht die Verbrauchsfolge auch nach der Optimierung weiterhin der geforderten Reihenfolge.

Wird für die Zukunft eine Verbrauchsfolgestrategie mit anderen oder strengeren Kriterien gefordert, oder werden die Prognosen ohne Referenzzeitraum erstellt, müssen die Auslagerungszeitpunkte der Lagereinheiten entsprechend an die zukünftigen Verbrauchsfolgekriterien angepasst werden, bevor die Optimierung vorgenommen wird.

Mengenoptimale Ausfassung

Die Vorgabe der mengenoptimalen Ausfassung verlangt, dass angebrochene Lagereinheiten nur dann verwendet werden, wenn das Anbrechen einer weiteren Lagereinheit sich auch durch die Verwendung einer vollständigen Lagereinheit nicht vermeiden lässt.

Beispiel:

Liegen eine halbe und eine nicht angebrochene Lagereinheit eines Artikels auf Lager und es wird eine Menge bestellt, die einer vollen Lagereinheit entspricht, dann ist die nicht angebrochene Lagereinheit zuerst zu verwenden. Wird hingegen eine Menge bestellt, die einer dreiviertel Lagereinheit entspricht, darf die halbe Lagereinheit benutzt werden, weil die nächste volle Lagereinheit in jedem Fall angebrochen werden müsste.

Letztendlich handelt es sich also auch hier um ein Verbrauchsfolgekriterium, das wie die anderen Verbrauchsfolgekriterien nur bei der Prognose der Zugriffszeitpunkte auf die einzelnen Lagereinheiten zu berücksichtigen ist. Da die Zuordnung der Lagereinheiten zu Bestellungen im Rahmen der Optimierung nicht verändert wird, entspricht auch die Lösung nach wie vor der gegebenen Vorgabe.

3.2.5. Behandlung von Sonderfällen

Teilentnahmen

Die Entnahme von Teilbeständen aus einer Lagereinheit führt dazu, dass nicht nur vollständige sondern auch angebrochene Lagereinheiten im Lager sind. Die angebrochene Lagereinheit wird nach der Teilentnahme im System mit dem reduzierten Mengeninhalt geführt. Der Platz bleibt belegt, solange die Lagereinheit nicht endgültig ausgelagert wird.

Da auf diese Lagereinheiten im Lauf ihrer Lagerzeit mehrfach zugegriffen wird, müssen sie bei der Platzierung gesondert berücksichtigt werden. Eine ausschließliche Betrachtung des endgültigen Auslagerungszeitpunkts würde unter Umständen zu einer Platzierung auf einem zu teuren Platz führen. Gleichzeitig ist eine Abbildung als Menge kleinerer Lagereinheiten auch nicht zielführend, da die Teilmengen nicht separat platziert werden können.

Eine Zulagerung zu angebrochenen Lagereinheiten wird nicht betrachtet, da sich hier weitere Anforderungen ergeben würden, die das Problem unter Umständen stark verkomplizieren könnten. Beispiele hierfür sind eine geforderte Sorten- oder Chargenreinheit pro Platz, die Notwendigkeit der Einhaltung eines gemeinsamen Verfallsdatums und Ähnliches.

Unter Einsatz der dynamischen Zonierung bietet es sich an, die durchschnittliche Zeit zwischen zwei Zugriffen für die Bestimmung des Verweildauerquantils anzuwenden. Damit wird die Lagereinheit entsprechend der durchschnittlichen Verweildauer einer ihrer Teilmengen platziert, was in etwa dem Ergebnis entspricht, das man erreichen würde, wenn alle Teilmengen separat platziert werden könnten.

Bei Anwendung der Job-Scheduling-Strategie wäre ein ähnlicher Ansatz möglich. Dabei würde man mit Hilfe der durchschnittlichen Zeit zwischen zwei Zugriffen einen theoretischen ersten Zugriffszeitpunkt bestimmen, anhand dessen die Lagereinheit einem Lagerplatz zugewiesen werden könnte. Natürlich müsste bei der Ermittlung der stabilen Menge für den Platz wieder die Gesamtverweildauer der Lagereinheit berücksichtigt werden, damit keine Doppelbelegungen entstehen. Je weiter die Zugriffe auseinander liegen, und je länger die Lagereinheit insgesamt auf Lager ist, desto größer wird hier jedoch die Gefahr, dass hierbei nicht das optimale Ergebnis erreicht wird.

Besser ist es daher, jeder Lagereinheit ein Gewicht zuzuweisen, das der Anzahl der Zugriffe auf die Lagereinheit entspricht. Daraufhin kann der Algorithmus so angepasst werden, dass er statt der Anzahl der Lagereinheiten pro Platz (größte stabile Menge) das Gesamtgewicht der dort gelagerten Lagereinheiten maximiert (schwerste stabile Menge⁵). Dies entspricht der Anwendung des Algorithmus für die Bestimmung der schwersten stabilen Menge auf einem Intervallgraphen, der in Kapitel 4.3.1 beschrieben ist. Wie im ungewichteten Fall gilt auch hier, dass bei mehreren schwersten stabilen Mengen nicht jede Wahl zur optimalen Lösung führen muss (siehe Kapitel 3.2.3).

Misch-Lagereinheiten

Als Misch-Lagereinheit werden alle Lagereinheiten bezeichnet, die nicht artikelrein sind. Verlässt eine solche Lagereinheit das Lager in demselben Zustand, in dem sie eingelagert wurde, kann sie wie eine gewöhnliche Lagereinheit behandelt werden. Wird die Lagereinheit bei Kommissionierung auseinandergenommen oder werden Teilbestände zu unterschiedlichen Zeitpunkten kommissioniert, entspricht die Behandlung einer Lagereinheit mit Teilentnahmen.

Konsolidierung von Restbeständen

In manchen Fällen werden in der Praxis Restbestände aus verschiedenen Lagereinheiten konsolidiert, um den benötigten Stauraum für die Restmengen zu reduzieren. Solche Maßnahmen werden von den hier diskutierten Optimierungsstrategien automatisch mit berücksichtigt, solange die zusätzlichen Zugriffe auf die Lagereinheiten entsprechend in den Daten abgebildet sind.

3.3. Dynamische Optimierung

Im Gegensatz zur statischen Optimierung (Kapitel 3.2) basiert die dynamische Optimierung nicht auf der Annahme vollständiger Information, sondern geht davon aus, dass Informationen über die Zukunft höchstens in Form von Prognosen zur Verfügung stehen, sich also „zur

⁵engl.: Maximum Weight Independent Set

Laufzeit“ ändern können. Um trotzdem ein gutes Ergebnis zu erzielen, können zwei Ansätze verfolgt werden: Entweder muss die Qualität der Prognosen sehr hoch sein, sodass nur mit wenigen oder kleinen Abweichungen zu rechnen ist, oder die Optimierungsstrategie muss in der Lage sein, flexibel auf Abweichungen zu reagieren. Das kann auch dadurch erreicht werden, dass die Strategie sich generell möglichst unabhängig von den Zukunftsprognosen macht.

Für die Erstellung der Zukunftsprognosen gibt es verschiedene Methoden, die in der einschlägigen Fachliteratur ausführlich diskutiert werden. Siehe hierzu beispielsweise [Gla08, S. 58ff.]. Darauf wird an dieser Stelle aber nicht näher eingegangen. Stattdessen werden im Folgenden die in Kapitel 3.2 entwickelten Strategien auf ihre Einsetzbarkeit in einem dynamischen Umfeld untersucht.

Dynamische Zonierung

Wie der Name der Strategie vermuten lässt, ist diese Strategie bereits speziell auf den Einsatz in einem dynamischen Umfeld ausgelegt und muss dafür nicht weiter angepasst werden.

Für die Einordnung einer Lagereinheit in die Verweildauerverteilung des jeweils aktuellen Bestands zum Zeitpunkt ihrer Einlagerung sind keine Informationen über die fernere Zukunft nötig. Wie viel des theoretischen Optimierungspotenzials realisiert werden kann, hängt jedoch stark von der Qualität der Verweildauerprognosen ab. Glass widmet diesem Thema deshalb ein eigenes Kapitel [Gla08, S. 95], in dem er auch einen Vorschlag zur praktischen Realisierung vorstellt. [Gla08, S. 100]

Job-Scheduling-Strategie

Da die Job-Scheduling-Strategie bei ihren Entscheidungen von Anfang an die Einlagerungs- und Zugriffsinformationen über den gesamten Betrachtungszeitraum hinweg berücksichtigt, ist sie stark von der Verlässlichkeit dieser Prognosen abhängig. Im Gegensatz zur dynamischen Zonierung ist sie dementsprechend relativ unflexibel gegenüber Abweichungen von diesen Prognosen. Die Verwendung eines Referenzzeitraums zur Erstellung der benötigten Prognosen würde daher vermutlich häufig zu ungenaue Werte liefern.

Treten Abweichungen auf, die das Beibehalten einer einmal erstellten Planung unmöglich machen, muss unter Fixierung der bereits gelagerten Lagereinheiten gegebenenfalls eine neue Lösung berechnet werden, die unter Umständen wieder nur von kurzer Gültigkeit wäre. Es ist daher fraglich, wie viel des theoretischen Optimierungspotenzials dieser Strategie praktisch realisiert werden kann.

4. Implementierung

Die Implementierung der in Kapitel 3 entwickelten Strategien erfolgt in Form eines Smalltalk-Prototyps, der die verschiedenen Algorithmen für die Anwendung auf einer gegebenen Datenquelle zur Verfügung stellt. Dieser Prototyp ist unabhängig von anderen Systemen und so konzipiert, dass er leicht mit jeder beliebigen Datenquelle kombiniert werden kann.

4.1. Design

Das Design des Prototyps ist angelehnt an Alistair Cockburns „Ports and Adapters“-Architektur [Coc05] (zu deutsch etwa: „Anschlüsse und Adapter“), einer Weiterentwicklung des Adapter-Patterns der Gang of Four [GHJV94, S. 139]. Diese Architektur basiert auf der Idee, dass die interne Logik einer Anwendung unabhängig sein sollte von der Technologie, die beispielsweise zur Darstellung oder zur Belieferung mit Daten eingesetzt wird. Für jede solche Schnittstelle wird deshalb ein Interface entwickelt, das die Art und Weise der Kommunikation mit der Anwendung definiert - die sogenannten Ports. Für jede eingesetzte Technologie (zum Beispiel eine Datenbank) wird nun ein Adapter geschrieben, der die bei der Anwendung eingehenden Ereignisse und Nachrichten in für sie verständliche Methodenaufrufe und Objekte übersetzt. Genauso übersetzt derselbe Adapter von der Anwendung ausgesandte Ereignisse und Nachrichten für die jeweilige Technologie, für die er geschrieben wurde.

Durch diesen Ansatz wird die entwickelte Anwendung nicht nur extrem flexibel in Bezug auf die Technologien, mit denen sie kombinierbar ist, sondern sie ist außerdem unabhängig von anderen Systemen und damit sehr leicht testbar, auch in automatisierten Testumgebungen.

Abbildung 4.1 zeigt die Architektur des Prototyps im Überblick. Der Daten-Port für den Prototyp wird dabei durch das Interface `DatenQuelle` repräsentiert. Jede Datenquelle, die mit dem Prototyp kombiniert werden soll, muss dieses Interface implementieren oder über einen Adapter daran angeschlossen werden. Intern arbeitet ein `Algorithmus` mit der Abstraktion einer `DatenQuelle` - welche Technologie tatsächlich dahinter steckt, ist für den `Algorithmus` nicht relevant.

Aus der Datenquelle wird das Lager erstellt, das das Herz des Prototypen darstellt. Die Klasse `Lager` modelliert ein Lager mit allen typischen Funktionen, wie dem Ein- und Auslagern von Lagereinheiten oder ganzen Bestellungen, und verwaltet zu jeder Zeit den aktuellen Zustand des Lagers. Sie berechnet außerdem die Kosten der im Lager zurückgelegten Wege.

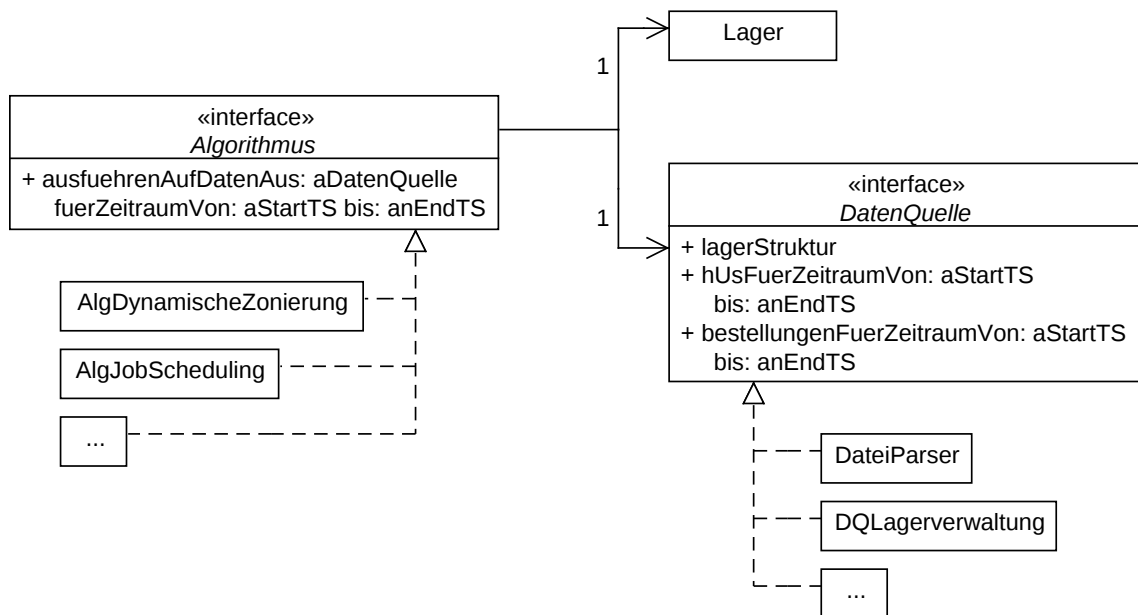


Abbildung 4.1.: Design des Prototyps

Die Methoden dieser Klasse sind auf ein Lager vom Typ 1a abgestimmt (vgl. Abbildung 3.2). Da die entwickelten Optimierungsstrategien, die in den Algorithmen implementiert sind, ebenfalls ein solches Lager voraussetzen, ist der Typ des Lagers nicht austauschbar. Da das Lager jedoch auf Basis der Informationen aus der Datenquelle aufgebaut wird, können die Algorithmen trotzdem problemlos auf verschiedenen Lagern desselben Typs ausgeführt werden. Details hierzu finden sich in Kapitel 4.2 und in den Kommentaren im Quellcode.

Die ebenfalls abstrakte Klasse *Algorithmus*, die das Interface eines Algorithmus definiert, stellt einen Teil der Schnittstelle zum Benutzer dar. Die Algorithmen holen sich alle benötigten Informationen aus der Datenquelle, erstellen sich daraus ein Lager, auf dem sie ihre Berechnungen ausführen, und geben den optimierten Belegungsverlauf zurück. Sind außerdem die Kommissionierkosten interessant, die bei diesem Belegungsverlauf für eine gegebene Menge von Bestellungen entstehen, so kann diese Information von einem Lager aus derselben Datenquelle abgefragt werden.

4.2. Umsetzung und Annahmen

Die Datenquelle stellt alle Informationen bereit, die zur Ausführung der Algorithmen benötigt werden. Das sind:

- Informationen zur Struktur des Lagers.
- Die im gegebenen Zeitraum in diesem Lager gelagerten Lagereinheiten.

- Die im gegebenen Zeitraum aus diesem Lager bedienten Bestellungen.

Die Details zur Form, in der diese Informationen bereitzustellen sind, sind in den Kommentaren im Quelltext dokumentiert.

Aus den Informationen zur Lagerstruktur erstellt sich der Algorithmus eine Instanz der Klasse Lager. Dabei wird ein Lager erwartet, das aus einer oder mehreren Gassen mit je einem oder zwei Regalen besteht. Die Regale sind wiederum in Ebenen (horizontal) und Säulen (vertikal) unterteilt. Dadurch ist jeder Lagerplatz im Regal über seine Koordinaten eindeutig identifizierbar.

		Säule			
		1	2	...	n
Ebene	m	7			
	...	5	...		
	2	3	11		
	1	1	9		

Abbildung 4.2.: Frontalansicht eines Lagerregals

Bezüglich des Lagers und der zugehörigen Lagereinheiten geht der Algorithmus davon aus, dass jede Lagereinheit auf jedem Lagerplatz innerhalb des von der Datenquelle bereitgestellten Lagers platziert werden kann. Sollten hier Einschränkungen bestehen, zum Beispiel aufgrund von Gewichts- oder Größenbeschränkungen, sind diese bei Erstellung der Datenquelle zu berücksichtigen. Gegebenenfalls ist eine entsprechende Aufteilung eines Lagers in kleinere Teillager vorzunehmen, sodass diese Voraussetzung für jedes Teillager und die darin unterzubringenden Lagereinheiten erfüllt ist. Für jedes dieser Teillager muss dann eine eigene Datenquelle mit den relevanten Daten instanziiert und der Algorithmus separat darauf ausgeführt werden.

Jedem Lagerplatz ist außerdem eine Wegnummer zugeordnet. Diese Wegnummer ist für jeden Lagerplatz innerhalb eines Lagers eindeutig und bestimmt die Reihenfolge, in der die Plätze innerhalb einer Gasse belegt werden. Eine eventuell zuvor vorgenommene Aufteilung eines Lagers in kleinere Teillager muss hier entsprechend berücksichtigt werden. Die Abbildungen 4.3 und 4.4 zeigen beispielhaft schematische Darstellungen der aus den historischen Daten übernommenen Reihenfolge.

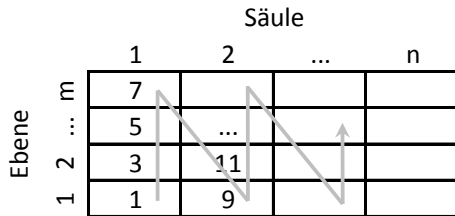
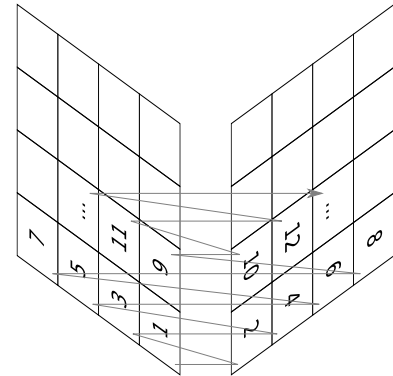


Abbildung 4.3.: Reihenfolge Einlagerung im Regal



Intern verwendet das Lager einen Cache, um teure Berechnungen nicht mehrfach ausführen zu müssen.

4.3. Algorithmen

Der Input für die Algorithmen besteht aus der Datenquelle und dem zu berücksichtigenden Betrachtungszeitraum. Rückgabewert ist der optimierte Belegungsverlauf für diesen Zeitraum, d.h. die Zuordnung jeder im Betrachtungszeitraum in diesem Lager gelagerten Lagereinheit zu einem Lagerplatz.

Im Folgenden sind die komplexeren der eingesetzten Algorithmen kurz theoretisch beschrieben.

4.3.1. Schwerste stabile Menge im Intervallgraphen

Dieser Algorithmus wird für die Job-Scheduling-Strategie verwendet. Um auch Lagereinheiten mit Teilentnahmen angemessen berücksichtigen zu können (siehe hierzu auch Kapitel 3.2.5), wird hier die gewichtete Variante des Algorithmus implementiert.

Gegeben sei eine Menge $\mathcal{L} = \{L_1, L_2, \dots, L_n\}$ von Intervallen, wobei jedes Intervall $L_i = (s_i, e_i, w_i)$ charakterisiert ist durch einen Startzeitpunkt s_i , einen Endzeitpunkt e_i sowie ein Gewicht w_i .

Ziel ist es, eine Indexmenge $S \subseteq \{1 \dots n\}$ zu finden, sodass sich keine zwei der entsprechenden Intervalle überschneiden und gleichzeitig das Gesamtgewicht der Intervalle mit Indizes aus S maximal wird. Für alle $i, j \in S$ soll also gelten:

$$L_i \cap L_j = \emptyset \quad (4.1)$$

und

$$\sum_{i \in S} w_i \text{ ist maximal.} \quad (4.2)$$

Der Algorithmus betrachtet alle Intervallgrenzen (Start- und Endzeitpunkte) in chronologischer Reihenfolge. Bei n Intervallen sind das maximal $2n$ viele.

Seien die Intervallgrenzen $r_1, r_2, \dots, r_k$. Für jede Intervallgrenze r_i wird in einem Array $\text{maxWeight}[1 \dots k]$ jeweils in $\text{maxWeight}[i]$ das maximale Gewicht einer stabilen Menge verwaltet, deren spätestes Intervall in oder vor r_i endet. $\text{maxWeight}[1]$ wird dementsprechend mit dem Gewicht des schwersten Intervalls initialisiert, das in r_1 endet.

Zur Bestimmung von $\text{maxWeight}[i]$ werden alle Intervalle betrachtet, die in r_i enden. Sei $L = (s, e, w)$ eines dieser Intervalle. Falls L in der schwersten stabilen Menge enthalten ist, müssen die übrigen Intervalle aus dieser Menge eine stabile Menge bilden, deren

spätestes Intervall in oder vor s endet. Damit erhalten wir für die stabile Menge mit L ein Gesamtgewicht von

$$w_{sum} = \text{maxWeight}[f] + w \quad (4.3)$$

wobei $r_f = s$. Folglich gilt

$$\text{maxWeight}[i] = \max(\text{maxWeight}[i-1], \text{maxWeight}[f] + w) \quad (4.4)$$

über alle in r_i endenden Intervalle $L = (s, e, w)$.

Dieser Algorithmus läuft in $O(n \log n)$ wegen der chronologischen Sortierung der Intervallgrenzen.

4.3.2. Färbe-Algorithmus für Intervallgraphen

Dieser Algorithmus wird für die Strategie zur Minimierung der Anzahl verwendeter Lagerplätze eingesetzt.

Die chromatische Zahl ist minimale Anzahl von Farben, die benötigt wird, um einen Graphen zu färben. Dabei dürfen keine zwei durch eine Kante verbundenen Knoten die gleiche Farbe bekommen. Im Intervallgraphen entspricht die chromatische Zahl der Größe der größten Clique im Graphen, also der größten Menge von Knoten, in der jedes beliebige Knotenpaar durch eine Kante verbunden ist. Im vorliegenden Fall repräsentiert diese Clique die größte Menge von Lagereinheiten, die irgendwann im betrachteten Zeitraum gleichzeitig auf Lager ist, und von denen daher keine zwei gleichzeitig auf demselben Platz gelagert werden können.

Um zudem eine möglichst hohe Frequentierung von günstigen Plätzen zu erreichen, ist in diesem Fall ein weiteres Ziel, die Färbung des Intervallgraphen so vorzunehmen, dass immer möglichst viele Knoten die gleiche Farbe bekommen. Anschließend werden die Farben nach der Häufigkeit ihrer Nutzung sortiert und je einem Lagerplatz zugeordnet: Die am häufigsten genutzte Farbe wird dem günstigsten Lagerplatz zugeordnet, die am nächsthäufigsten genutzte Farbe dem nächstgünstigsten Lagerplatz, usw. Als Nutzung zählt dabei jeder Zugriff auf eine Lagereinheit mit dieser Farbe.

Gegeben sei eine Menge $\mathcal{L} = \{L_1, L_2, \dots, L_n\}$ von Intervallen, wobei jedes Intervall $L_i = (s_i, e_i, w_i)$ charakterisiert ist durch einen Startzeitpunkt s_i , einen Endzeitpunkt e_i sowie ein Gewicht w_i . Gesucht ist die optimale Färbung des Intervallgraphen.

Die Liste $\text{color}[1 \dots c]$ verwaltet in jedem Element $\text{color}[i]$ die Liste von Intervallen, die dieser Farbe zugeordnet sind. Die Menge $F \subset \{1 \dots c\}$ enthält zu jedem Zeitpunkt die aktuell verfügbaren Farben.

Der Algorithmus betrachtet alle Intervallgrenzen (Start- und Endzeitpunkte) in chronologischer Reihenfolge. Bei n Intervallen sind das maximal $2n$ viele.

Seien die Intervallgrenzen $r_1, r_2, \dots, r_k$. Für jede Intervallgrenze r_i werden alle Intervalle betrachtet, die in r_i enden. Die ihnen zugeordneten Farben werden in die Menge F der

verfügbaren Farben aufgenommen. Daraufhin bekommt jedes Intervall, das in r_i beginnt, die freie Farbe aus F zugewiesen, die bisher schon am häufigsten genutzt wurde.

Dieser Algorithmus läuft in $O(n \log n)$ wegen der chronologischen Sortierung der Intervallgrenzen.

5. Evaluation

5.1. Empirische Untersuchungen

Um die Vergleichbarkeit der Ergebnisse zu gewährleisten, wurde für alle empirischen Untersuchungen dasselbe Lager mit denselben Daten zu Lagereinheiten und Bestellungen verwendet. Auch die durch die Wegnummern definierte Einlagerungsreihenfolge (vgl. Kapitel 4.2) wurde beibehalten.

Das Lager besteht aus fünf einseitig begehbaren Gassen à zwei Regalen. Jedes der Regale hat sechs Ebenen mit je bis zu 26 Plätzen. Insgesamt umfasst das Lager 1.514 Plätze. In Abbildung 5.1 ist das Lager in der Draufsicht dargestellt. Kommissionierplätze sind darin mit einem ‚K‘ gekennzeichnet.

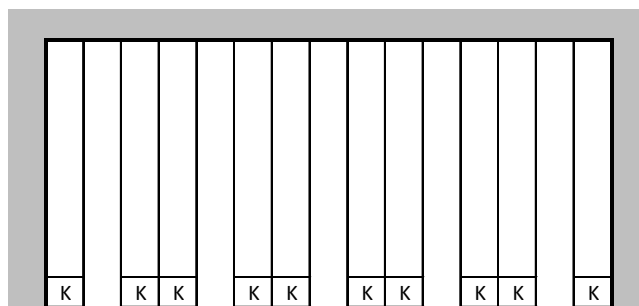


Abbildung 5.1.: Lager für die empirischen Untersuchungen

Bezüglich der Wege im Lager wurden für die Berechnungen folgende Annahmen getroffen:

1. Der Weg von einem Kommissionierplatz zur ersten Säule einer Gasse beträgt 1. Wege zwischen den Kommissionierplätzen werden nicht berücksichtigt.
2. Der Weg zwischen zwei benachbarten Säulen eines Regals beträgt 1.
3. Der Weg zwischen zwei benachbarten Ebenen eines Regals beträgt 1.
4. Diagonale Bewegungen sind nicht erlaubt.
5. Der Weg von einem Regal zum anderen innerhalb derselben Gasse ist vernachlässigbar.

Bei diesen Annahmen wurde bewusst auf Einheiten verzichtet, um deutlich zu machen, dass diese Größen nicht die tatsächlichen Maße des Lagers darstellen, sondern nur das Verhältnis der Wege untereinander definieren.

Da die Lagerplätze auf den verschiedenen Ebenen nicht alle die gleichen Attribute haben, wurde für die empirischen Untersuchungen nur das Teillager bestehend aus den Ebenen 3, 4 und 5 betrachtet, dessen Lagerplätze äquivalent sind. Dieses Teillager umfasst 770 Plätze.

Als zu platzierende Lagereinheiten wurden für jeden Algorithmus die historischen Lagereinheiten aus dem angegebenen Zeitraum verwendet. Damit eine Lagereinheit von den Algorithmen platziert werden kann, muss insbesondere der Auslagerungszeitpunkt der Lagereinheit bekannt sein. Daher wurden hier nur solche Lagereinheiten berücksichtigt, für die die entsprechenden Zugriffsdaten vorlagen. Waren Zugriffe vermerkt, aber kein Auslagerungszeitpunkt angegeben, wurde der letzte Zugriff als Auslagerung interpretiert.

Die folgenden Kapitel liefern einen Überblick über die Ergebnisse der empirischen Untersuchungen und interpretieren diese Ergebnisse. Die konkreten Ergebnisse aller durchgeführten Testreihen sind in Anhang A.2 zu finden.

5.1.1. Im Lager ohne Nachlieferungen

Da die hierzu untersuchten Strategien anhand der Zugehörigkeit der Lagereinheit zu einer Bestellung einlagern, wurden bei der Einlagerung keine Lagereinheiten berücksichtigt, auf die während des Betrachtungszeitraums kein Zugriff stattfindet. Diese Lagereinheiten würden bei Berücksichtigung auf freien Plätzen im hinteren Teil des Lagers eingelagert, würden also die Platzierung der im Betrachtungszeitraum kommissionierten Lagereinheiten nicht beeinflussen. Dadurch spielen sie auch bei der Berechnung der Kommissionierkosten für die übrigen Lagereinheiten keine Rolle.

Weiter wurde die Menge der Bestellungen aus dem Betrachtungszeitraum soweit eingeschränkt, dass keine zwei Bestellungen auf die selbe Lagereinheit zugreifen. Andernfalls würde die gemeinsame Platzierung aller einer Bestellung zugeordneter Lagereinheiten schnell unmöglich.

Trotz dieser Einschränkungen geht aus dem in Abbildung 5.2 dargestellten Ergebnis des empirischen Vergleichs deutlich hervor, dass sich die beiden Sortier-Strategien im Ergebnis kaum unterscheiden.

Für diese Abbildung wurde das Ergebnis der Sortier-Strategie von hinten als Vergleichsmaßstab angesetzt und die Ergebnisse der Sortier-Strategie von vorne dazu ins Verhältnis gesetzt. Ein Vergleich zur historischen Situation macht hier keinen Sinn, da die historische Platzierung der relevanten Lagereinheiten auch von der Platzierung anderer Lagereinheiten abhängig war, die aufgrund der oben genannten Einschränkungen bei den betrachteten Bestellungen von den Sortier-Strategien aber nicht berücksichtigt werden.

5.1.2. Im Lager mit Nachlieferungen

Anders als im Lager ohne Nachlieferungen wurden bei den Untersuchungen zum Lager mit Nachlieferungen auch Lagereinheiten mit einbezogen, auf die während des Betrachtungszeitraums kein Zugriff stattfindet. Da die hier untersuchten Strategien außerdem

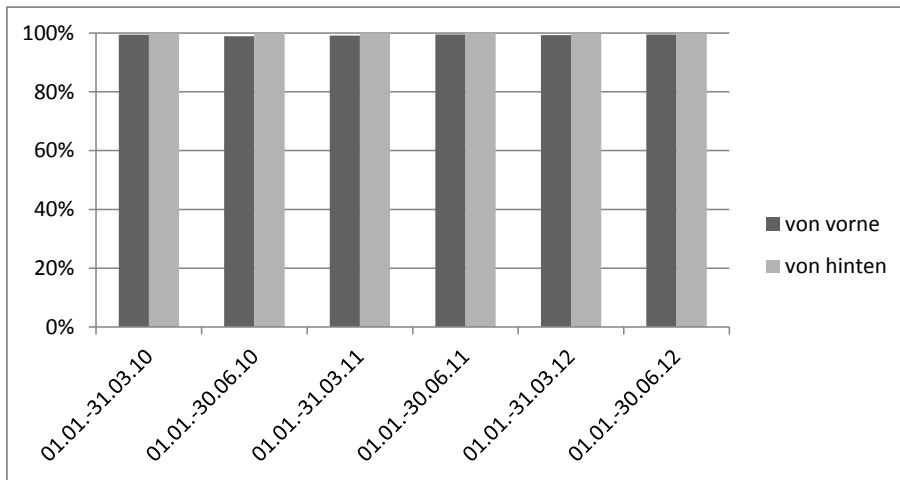


Abbildung 5.2.: Vergleich der Sortier-Strategien im Lager ohne Nachlieferungen

unproblematisch auf die Berücksichtigung von Teilentnahmen angepasst werden konnten (siehe hierzu auch Kapitel 3.2.5), mussten auch bei den Bestellungen keine Einschränkungen vorgenommen werden.

Die Abbildungen 5.3 und 5.4 zeigen die Ergebnisse des empirischen Vergleichs getrennt nach Kommissionierstrategie. In beiden Fällen wurden die historischen Kosten als Vergleichsmaßstab verwendet und alle anderen Ergebnisse dazu in Relation gesetzt.

Das Verhältnis der Kommissionierkosten unter Einsatz der jeweiligen Optimierungsstrategie im Vergleich zur historischen Situation ist bei beiden Kommissioniermethoden sehr ähnlich, weil die Kommissionierstrategie auch auf die absoluten Kosten nur wenig Einfluss hat. Diese hohe Vergleichbarkeit zwischen den beiden Kommissionierstrategien resultiert vor allem aus dem überwiegenden Anteil kleiner Bestellungen.

In jedem der untersuchten Zeiträume umfassen rund 70% der Bestellungen nur eine Lagereinheit. Insgesamt liegen etwa 90% der Bestellungen bei einer Größe von maximal 3 Lagereinheiten. Obwohl in manchen der untersuchten Zeiträume bezüglich der absoluten Anzahl von Bestellungen ein Vielfaches an dem kommissioniert wurde, was in anderen Zeiträumen anfiel, bleibt diese Zusammensetzung über alle Betrachtungszeiträume hinweg fast konstant. Deshalb haben die Synergieeffekte der Kommissionierung pro Bestellung keine erkennbaren Auswirkungen.

Während die Ergebnisse sowohl bei Zuweisung des günstigsten freien Platzes als auch unter Einsatz der Strategie zur Minimierung der Anzahl genutzter Lagerplätze eine große Bandbreite abdecken, erzielen sowohl die dynamische Zonierung als auch die Job-Scheduling-Strategie relativ gleichbleibende Resultate. Die Job-Scheduling-Strategie liegt im Schnitt sehr konstant leicht unter 40% der historischen Kosten, während die dynamische Zonierung eine Senkung der Kommissionierkosten auf durchschnittlich etwa 80% des historischen Niveaus erreicht.

5. Evaluation

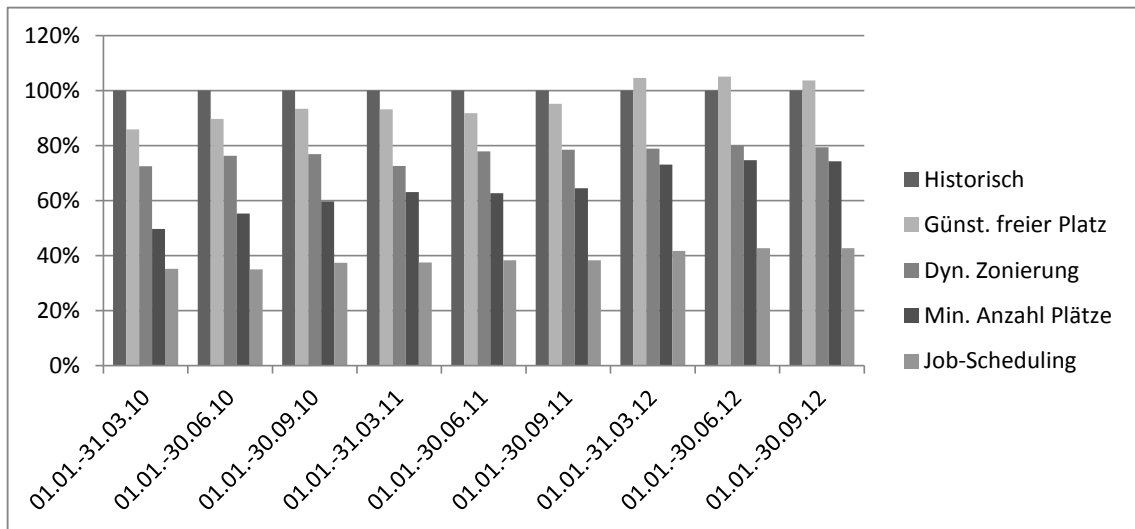


Abbildung 5.3.: Kommissionierung pro Lagereinheit im Lager mit Nachlieferungen

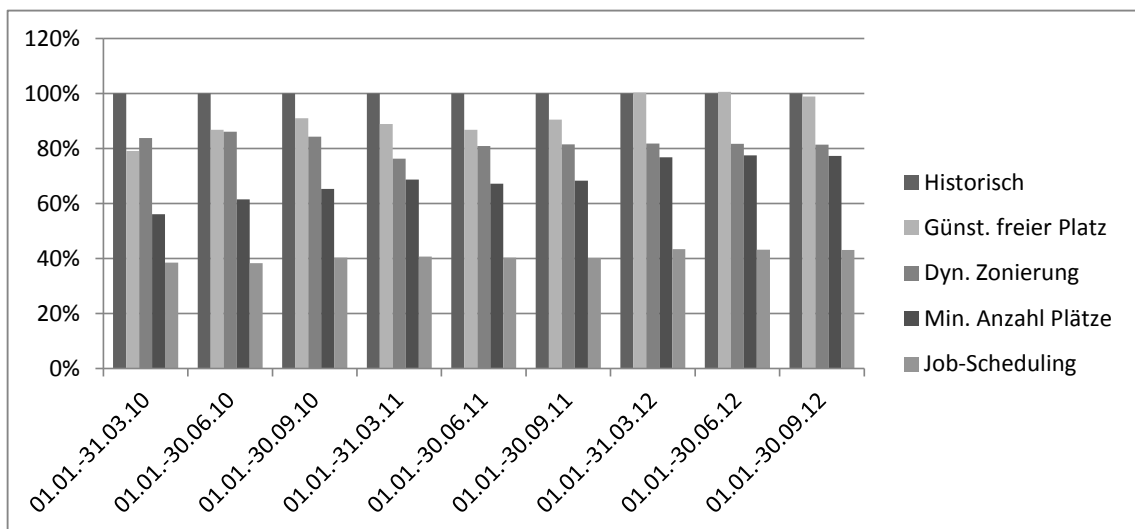


Abbildung 5.4.: Kommissionierung pro Bestellung im Lager mit Nachlieferungen

Die praktische Realisierung der Reduktionen, die die Job-Scheduling-Strategie hier erzielt, ist jedoch unrealistisch, da sie das Bekanntsein der Ein- und Auslagerungszeitpunkte aller Lagereinheiten bereits zu Beginn des Betrachtungszeitraums voraussetzt. Wie groß der realisierbare Anteil der möglichen Einsparungen tatsächlich ist, bliebe daher noch zu untersuchen. Siehe hierzu auch Kapitel 6.

Bei der dynamischen Zonierung hingegen handelt es sich um eine fertig entwickelte Strategie, die prinzipiell direkt praktisch einsetzbar ist. Auch hier sind zwar Zukunftsprognosen

anzustellen, diese beschränken sich jedoch auf die Prognose der Verweildauer der jeweils einzulagernden Lagereinheit zu deren Einlagerungszeitpunkt sowie der Verweildauern der zu diesem Zeitpunkt gelagerten Lagereinheiten. Diese Prognosen können gegebenenfalls im Verlauf angepasst werden, weil zum Zeitpunkt jeder Einlagerung eine neue Berechnung stattfindet.

5.2. Schlussfolgerungen

Die Ergebnisse der empirischen Untersuchungen zeigen ein signifikantes Optimierungspotenzial gegenüber der historischen Situation. Dabei erzielt besonders die Job-Scheduling-Strategie hervorragende Ergebnisse. Sollte beim Einsatz dieser Strategie aufgrund von Prognosefehlern selbst nur die Hälfte der möglichen Kostenersparnis realisiert werden können, wäre das Ergebnis immer noch besser als unter Einsatz der dynamische Zonierung, die von Glass als die „Strategie der Wahl“ [Gla08, S. 116] unter den Schnellläuferstrategien bezeichnet wird – selbst wenn man für diese Strategie von fehlerfreien Prognosen ausgeht.

Damit zeigen die Ergebnisse dieser Untersuchungen insbesondere, dass selbst bei als sehr gut bewerteten Strategien noch immenses Optimierungspotenzial besteht, das bisher ungenutzt bleibt.

6. Zusammenfassung und Ausblick

Aufgabe der Arbeit war die Formulierung des Ziels der langfristigen Minimierung von Kommissionierkosten als diskretes Optimierungsproblem; die Entwicklung, Implementierung und Bewertung von Algorithmen zu dessen Lösung sowie die empirische Untersuchung anhand historischer Daten. Dafür wurden für verschiedene Szenarien neue Algorithmen entwickelt und implementiert und bezüglich ihrer Leistungsfähigkeit mit bekannten Algorithmen verglichen.

6.1. Fazit

Die Ähnlichkeit der Ergebnisse der empirischen Untersuchungen über alle betrachteten Zeiträume und Kommissioniermethoden hinweg zeigt deutlich, dass die entwickelten Strategien nicht nur in eingeschränkten Szenarien zu deutlichen Verbesserungen führen. Das lässt die Reproduzierbarkeit dieser Ergebnisse auch für andere Zeiträume und andere Lager desselben Typs erwarten.

Wie groß das Einsparungspotenzial gegenüber der aktuellen Situation im konkreten Fall ist, hängt dabei natürlich immer von der bisher eingesetzten Einlagerungsstrategie ab. Anhand historischer Daten kann dieses Potenzial für den Einzelfall leicht ermittelt werden.

Da die Qualität der erzielten Ergebnisse bei jeder der entwickelten Optimierungsstrategien von der Qualität der Prognosen abhängt, wird in der Praxis wohl nie das gesamte theoretische Optimierungspotenzial realisiert werden können. In Anbetracht der hohen Betriebskosten eines Lagers könnten jedoch schon einige Prozent Einsparung einen Wettbewerbsvorteil bedeuten. Hier scheinen weitere Investitionen in eine praktisch anwendbare Version der Job-Scheduling-Strategie besonders lohnenswert, da diese im Vergleich mit den anderen Strategien die mit Abstand größten Einsparungen verspricht.

6.2. Anknüpfungspunkte

Die Untersuchungen zeigen weiterhin, dass das Optimum dessen, was erreicht werden kann, noch weit entfernt ist von dem, was bisherige Schnellläuferstrategien leisten können. Daher ergeben sich aus den Ergebnissen der vorliegenden Arbeit zahlreiche Anknüpfungspunkte für weitere Untersuchungen.

Dazu gehört sicherlich die Weiterentwicklung der Job-Scheduling-Strategie als vielversprechender Ansatz für eine Schnelldläuferstrategie, aber auch die weitere Verfolgung anderer Ansätze in diesem Bereich. Die Job-Scheduling-Strategie ist in der aktuell vorliegenden Form nicht ohne Anpassungen in einem dynamischen Umfeld einsetzbar. Hier müssten weitere Untersuchungen angestellt werden, um diejenige Form der Anpassung zu finden, die den größten Anteil des theoretischen Einsparpotenzials realisieren kann.

Interessant wären auch differenziertere empirische Untersuchungen zu den hier diskutierten Strategien, wie beispielsweise ein Vergleich der Strategien bei unterschiedlichen Lagerfüllgraden oder großen Ungleichverteilungen der Zugriffswahrscheinlichkeiten im Sortiment.

Aus Kundenperspektive könnte außerdem ein Vorgehen attraktiv sein, bei dem keine initiale Reorganisation des Lagers erfolgt, sondern nur neu eingehende Lagereinheiten entsprechend der gewählten Strategie platziert werden. Wie viel Optimierungspotenzial durch eine solche „passive Reorganisation“ verloren ginge und wie lange es dauern würde, bis die Unterschiede zur „aktiven Reorganisation“ mit initialer Umlagerung ausgeglichen wären, bliebe ebenfalls noch zu untersuchen.

A. Anhang

A.1. Beweise

A.1.1. Optimalität der Sortier-Strategie im linearen Lager

Jeder Tausch zweier Bestellungen B_i und B_j in der Reihenfolge mit $i < j$ würde die Summe der Kommissionierkosten für die getauschten Bestellungen um ihre Größendifferenz

$$d_{i,j} = |B_j| - |B_i| \quad (\text{A.1})$$

ändern. Zudem würden auch alle Lagereinheiten der dazwischen liegenden Bestellungen $B_{i+1}, \dots, B_{j-1}$ um $d_{i,j}$ verschoben. Die Änderung der Gesamtkosten läge in der Folge bei

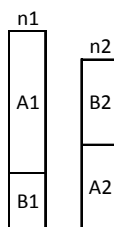
$$d = d_{i,j} * (j - i) \quad (\text{A.2})$$

Da aber aufgrund der Sortierung jede Bestellung mindestens so groß ist wie ihre Vorgänger, also immer $d_{i,j} \geq 0$ ist, kann kein Tausch zweier Bestellungen zu einem besseren Ergebnis führen. ■

A.1.2. Ein Split ist besser als zwei Splits

Gegeben seien zwei voll belegte Gassen mit n_1 bzw. n_2 Plätzen sowie die Menge $\mathcal{B} = \{A, B\}$ der darin unterzubringenden Bestellungen. A_1 und A_2 bzw. B_1 und B_2 bezeichnen die beiden Teile der Bestellung A bzw. B im Fall eines Splits.

Splittet man sowohl A als auch B , ergibt sich folgende Konstellation:

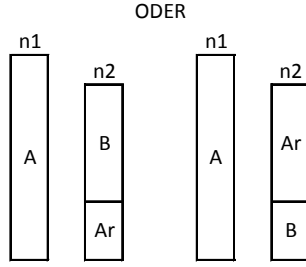


$$c_{2S}(\mathcal{B}) = n_1 + n_2 + |A_2| + |B_1|$$

Abbildung A.1.: Fall mit 2 Splits

Trennt man nur eine der beiden Bestellungen, muss zwischen zwei Fällen unterschieden werden.

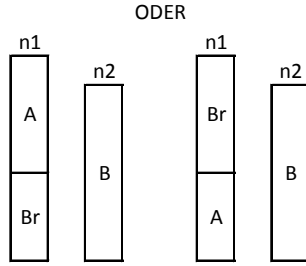
Fall 1: $A_2 > |B_1|$



$$\begin{aligned}
 c(\mathcal{B}) &= n_1 + n_2 + \min(|A_r|, |B|) \\
 &= n_1 + n_2 + \min(|A_2| - |B_1|, |B|) \quad (\text{A.3}) \\
 &< c_{2S}(\mathcal{B})
 \end{aligned}$$

Abbildung A.2.: Fall $|A_2| > |B_1|$

Fall 2: $A_2 \leq |B_1|$



$$\begin{aligned}
 c(\mathcal{B}) &= n_1 + n_2 + \min(|B_r|, |A|) \\
 &= n_1 + n_2 + \min(|B_1| - |A_2|, |A|) \quad (\text{A.4}) \\
 &< c_{2S}(\mathcal{B})
 \end{aligned}$$

Abbildung A.3.: Fall $|A_2| \leq |B_1|$

In beiden Fällen sind also die Gesamtkosten echt kleiner als beim Split beider Bestellungen. ■

A.1.3. Keine Reorganisation wegen Split bei Einlagerung von hinten im realistischen Lager

Bestellung B_i sei zu platzieren. Alle bereits platzierten Bestellungen haben eine höhere Kardinalität als B_i und damit auch eine höhere Kardinalität als jede Teilmenge von B_i . Es kann sich daher keine Situation ergeben, in der einer der durch den Split entstehenden Teile von B_i vor einer der bereits platzierten Bestellungen hätte platziert werden müssen, um der durch die Strategie vorgegebenen Reihenfolge zu entsprechen.

Für die beiden Teile von B_i sind zwei Fälle zu unterscheiden:

Fall 1 $|B_{i1}| \geq |B_{i2}|$: Der platzierte Teil ist gleich groß oder größer als der verbleibende.

In diesem Fall würde B_{i2} zu genau demselben Zeitpunkt platziert als hätte man die beiden Teile von B_i bereits von Anfang an getrennt behandelt. Da im Reorganisationsfall keine neuen Splits akzeptiert werden, würde B_{i2} also auch zu den gleichen Kosten platziert.

Fall 2 $|B_{i1}| < |B_{i2}|$: Der platzierte Teil ist kleiner als der verbleibende.

Bei einer Reorganisation würde B_{i2} vor B_{i1} platziert. Wegen $|B_{i1}| < |B_{i2}|$ könnte B_{i2} aber nicht den aktuellen Platz von B_{i1} einnehmen. Dies könnte nur bei einer Bestellung der gleichen Kardinalität passieren, und diese Veränderung würde an den Gesamtkosten der Kommissionierung nichts ändern. ■

A.1.4. Job-Scheduling-Strategie erzielt optimale Frequentierung

Sei n die Anzahl der Plätze im Lager. Sei S_p die Menge der Lagereinheiten, die auf die Plätze $\{p, \dots, n\}$ zu verteilen sind, und sei $\mathcal{P}$ die Menge der dem Lagerplatz p zugewiesenen Lagereinheiten.

Nimmt man an, dass der Algorithmus nicht das optimale Ergebnis liefert, dann muss es einen Platz p geben, dem nicht die maximale Menge von Lagereinheiten aus S_p zugewiesen wurde. Daraus folgt, dass es eine Menge $\mathcal{M}$ von Lagereinheiten aus S_p geben muss, deren Lagerzeiten sich nicht überschneiden, und für die gilt: $|\mathcal{M}| > |\mathcal{P}|$.

Dies wiederum impliziert, dass es eine Lagereinheit $x \in \mathcal{M}$ mit $x \notin \mathcal{P}$ gibt, sodass x die Lagereinheit mit dem frühesten Kommissionierzeitpunkt aus $\mathcal{M} \triangle \mathcal{P}$ ist. Das aber wäre ein Widerspruch zur Definition des Algorithmus, da $(\mathcal{M} \triangle \mathcal{P}) \subseteq S_p$. ■

A.1.5. Optimale Lösung besteht aus maximalen stabilen Mengen

Sei n die Anzahl der Plätze im Lager, und sei $F(p_i)$ die Anzahl der Kommissioniervorgänge auf Platz p_i im betrachteten Zeitraum. Dann sind die Kosten für die Kommissionierung aller in diesem Zeitraum gelagerten Lagereinheiten

$$c = \sum_{i=1}^n c(p_i) \cdot F(p_i) \quad (\text{A.5})$$

Angenommen, eine optimale Lösung bestünde nicht aus einer Kombination von maximalen stabilen Mengen. Dann gäbe es einen Platz p_i , auf dem in dieser Lösung keine maximale stabile Menge liegt. Auf einem teureren Platz p_j läge also eine Lagereinheit, die ohne Konflikte auch auf p_i gelagert werden könnte.

Durch die Umlagerung dieser Lagereinheit von p_j nach p_i kann sich auf den anderen Plätzen nichts verschlechtern. Jede der dort gelagerten Mengen ist immer noch stabil. Gleichzeitig würden sich die Gesamtkosten für die Kommissionierung um

$$d = c(p_i) - c(p_j) < 0 \quad (\text{A.6})$$

verändern. Wegen $d < 0$ kann die untersuchte Lösung also nicht optimal gewesen sein. ■

A.2. Anlagen zu den durchgeführten Testreihen

A.2.1. Verteilung der Bestellgrößen pro Betrachtungszeitraum

Die folgenden Tabellen zeigen die Verteilung der Bestellgrößen für jeden untersuchten Betrachtungszeitraum. Für jede Bestellgröße ist sowohl die absolute Anzahl als auch der prozentuale Anteil der Bestellungen in dieser Größe angegeben.

Verwendete Abkürzungen:

LE Lagereinheit

Anz. LEs	01.01.-31.03.10		01.01.-30.06.10		01.01.-30.09.10	
1	412	70%	1076	70%	1704	71%
2	86	15%	259	17%	406	17%
3	42	7%	78	5%	124	5%
4	9	2%	28	2%	49	2%
5	7	1%	18	1%	30	1%
6	7	1%	14	1%	25	1%
7	7	1%	10	1%	12	0%
8	3	1%	18	1%	21	1%
9	3	1%	7	0%	9	0%
10	3	1%	4	0%	5	0%
11	3	1%	9	1%	10	0%
12	1	0%	3	0%	4	0%
13	1	0%	3	0%	3	0%
14	0	0%	0	0%	1	0%
15	1	0%	2	0%	2	0%
16	1	0%	1	0%	2	0%
17	0	0%	1	0%	1	0%
18	0	0%	0	0%	1	0%
19	0	0%	0	0%	0	0%
20	0	0%	0	0%	0	0%
21	0	0%	1	0%	1	0%
	586	100%	1532	100%	2410	100%

Tabelle A.1.: Bestellgrößen Betrachtungszeiträume 2010

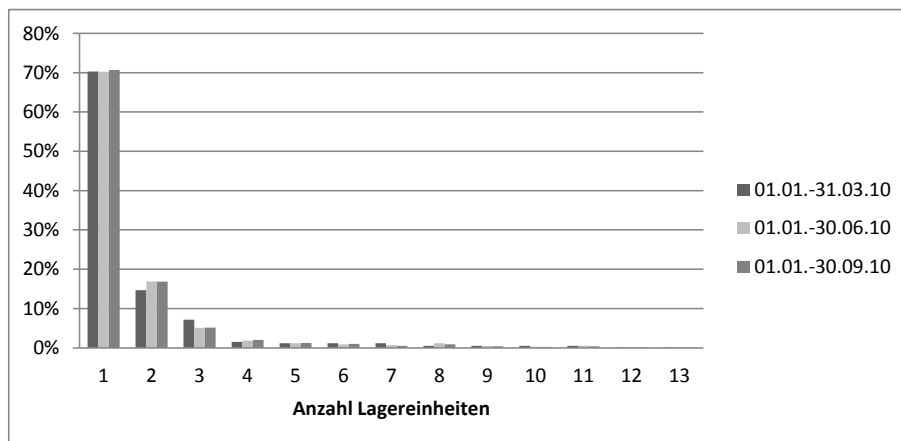


Abbildung A.4.: Verteilung der Bestellgrößen 2010

Anz. LEs	01.01.-31.03.11		01.01.-30.06.11		01.01.-30.09.11	
1	645	65%	1285	66%	2075	68%
2	177	18%	343	18%	513	17%
3	66	7%	137	7%	203	7%
4	38	4%	64	3%	110	4%
5	25	3%	47	2%	64	2%
6	13	1%	22	1%	31	1%
7	5	1%	14	1%	18	1%
8	7	1%	13	1%	17	1%
9	4	0%	6	0%	13	0%
10	2	0%	3	0%	5	0%
11	3	0%	7	0%	8	0%
12	2	0%	2	0%	2	0%
13	0	0%	1	0%	2	0%
	987	100%	1944	100%	3061	100%

Tabelle A.2.: Bestellgrößen Betrachtungszeiträume 2011

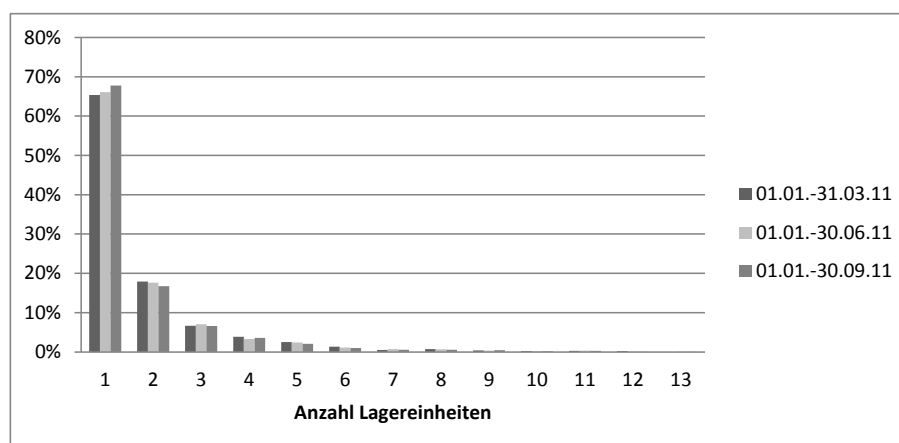


Abbildung A.5.: Verteilung der Bestellgrößen 2011

A.2. Anlagen zu den durchgeführten Testreihen

Anz. LEs	01.01.-31.03.12		01.01.-30.06.12		01.01.-30.09.12	
1	934	66%	1833	67%	2721	67%
2	256	18%	467	17%	698	17%
3	85	6%	160	6%	248	6%
4	43	3%	82	3%	128	3%
5	38	3%	58	2%	84	2%
6	19	1%	38	1%	59	1%
7	10	1%	24	1%	35	1%
8	11	1%	21	1%	29	1%
9	4	0%	14	1%	17	0%
10	3	0%	7	0%	8	0%
11	2	0%	5	0%	7	0%
12	1	0%	5	0%	8	0%
13	1	0%	4	0%	4	0%
14	1	0%	2	0%	3	0%
15	2	0%	3	0%	3	0%
16	0	0%	0	0%	4	0%
17	0	0%	0	0%	1	0%
18	0	0%	0	0%	0	0%
19	0	0%	0	0%	2	0%
20	0	0%	0	0%	1	0%
	1410	100%	2723	100%	4060	100%

Tabelle A.3.: Bestellgrößen Betrachtungszeiträume 2012

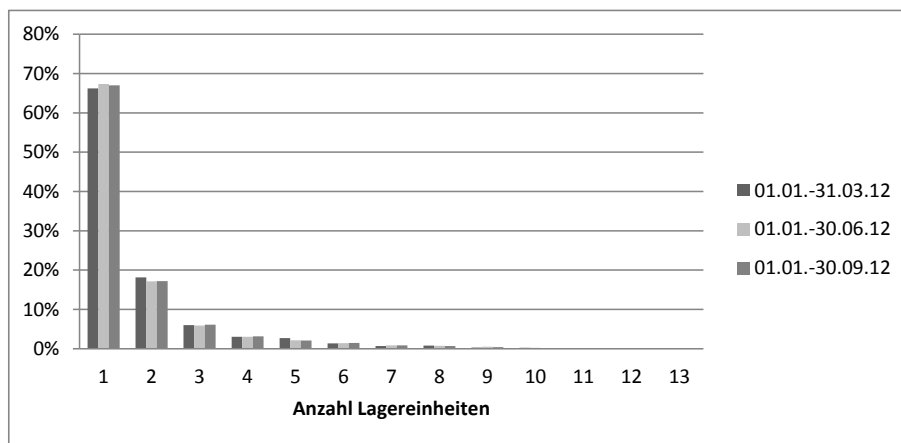


Abbildung A.6.: Verteilung der Bestellgrößen 2012

A.2.2. Anzahl disjunkter Bestellungen pro Betrachtungszeitraum

Zeitraum	Anz. Best.	disjunkt
01.01.-31.03.10	586	137
01.01.-30.06.10	1.532	345
01.01.-30.09.10	2.410	526

Tabelle A.4.: Disjunkte Bestellungen 2010

Zeitraum	Anz. Best.	disjunkt
01.01.-31.03.11	987	222
01.01.-30.06.11	1.944	396
01.01.-30.09.11	3.061	584

Tabelle A.5.: Disjunkte Bestellungen 2011

Zeitraum	Anz. Best.	disjunkt
01.01.-31.03.12	1.410	278
01.01.-30.06.12	2.723	496
01.01.-30.09.12	4.060	660

Tabelle A.6.: Disjunkte Bestellungen 2012

A.2.3. Ergebnisse im Lager ohne Nachlieferungen

Die folgenden Tabellen geben die Distanzen an, die unter Anwendung der jeweiligen Strategie bei Kommissionierung der disjunkten Bestellungen des Betrachtungszeitraums im Lager zurückzulegen sind. Die prozentualen Angaben zeigen das Verhältnis zu den Kosten unter Anwendung der Sortier-Strategie von hinten.

Für die Annahmen zu den Distanzen im Lager siehe Kapitel 5.1.

Zeitraum	Anz. Best.	Sortier-Strategie			
		von vorne		von hinten	
01.01.-31.03.10	137	1.844	99%	1.854	100%
01.01.-30.06.10	345	7.086	99%	7.164	100%

Tabelle A.7.: Ergebnisse im Lager ohne Nachlieferungen 2010

A.2. Anlagen zu den durchgeführten Testreihen

Zeitraum	Anz. Best.	Sortier-Strategie			
		von vorne		von hinten	
01.01.-31.03.11	222	3.526	99%	3.556	100%
01.01.-30.06.11	396	8.612	99%	8.656	100%

Tabelle A.8.: Ergebnisse im Lager ohne Nachlieferungen 2011

Zeitraum	Anz. Best.	Sortier-Strategie			
		von vorne		von hinten	
01.01.-31.03.12	278	4.934	99%	4.970	100%
01.01.-30.06.12	496	12.668	99%	12.732	100%

Tabelle A.9.: Ergebnisse im Lager ohne Nachlieferungen 2012

A.2.4. Ergebnisse im Lager mit Nachlieferungen

Die folgenden Tabellen geben die Distanzen an, die unter Anwendung der jeweiligen Strategie bei Kommissionierung der Bestellungen des Betrachtungszeitraums im Lager zurückzulegen sind. Die prozentualen Angaben zeigen das Verhältnis zu den historischen Kosten.

Für die Annahmen zu den Distanzen im Lager siehe Kapitel 5.1.

Verwendete Abkürzungen:

GFP	Günstigster Freier Platz
Dyn. Zon.	Dynamische Zonierung
Min. Plätze	Minimale Anzahl benutzter Plätze
JS	Job-Scheduling

Kommissionierung pro Lagereinheit

Zeitraum	Historisch		GFP		Dyn. Zon.		Min. Plätze		JS	
01.01.-31.03.10	34.262	100%	29.428	86%	24.822	72%	17.038	50%	12.060	35%
01.01.-30.06.10	89.660	100%	80.390	90%	68.406	76%	49.588	55%	31.432	35%
01.01.-30.09.10	133.590	100%	124.790	93%	102.662	77%	79.752	60%	49.980	37%

Tabelle A.10.: Ergebnisse bei Kommissionierung pro Lagereinheit 2010

A. Anhang

Zeitraum	Historisch		GFP		Dyn. Zon.		Min. Plätze		JS	
01.01.-31.03.11	56.008	100%	52.168	93%	40.628	73%	35.328	63%	21.014	38%
01.01.-30.06.11	112.600	100%	103.386	92%	87.676	78%	70.640	63%	43.102	38%
01.01.-30.09.11	171.240	100%	162.996	95%	134.454	79%	110.370	64%	65.642	38%

Tabelle A.11.: Ergebnisse bei Kommissionierung pro Lagereinheit 2011

Zeitraum	Historisch		GFP		Dyn. Zon.		Min. Plätze		JS	
01.01.-31.03.12	79.252	100%	82.848	105%	62.502	79%	57.934	73%	33.092	42%
01.01.-30.06.12	156.478	100%	164.334	105%	125.084	80%	116.894	75%	66.902	43%
01.01.-30.09.12	234.614	100%	243.284	104%	186.320	79%	174.290	74%	100.258	43%

Tabelle A.12.: Ergebnisse bei Kommissionierung pro Lagereinheit 2012

Kommissionierung pro Bestellung

Zeitraum	Historisch		GFP		Dyn. Zon.		Min. Plätze		JS	
01.01.-31.03.10	25.918	100%	20.506	79%	21.718	84%	14.522	56%	9.978	38%
01.01.-30.06.10	67.728	100%	58.754	87%	58.304	86%	41.636	61%	25.944	38%
01.01.-30.09.10	104.694	100%	95.210	91%	88.212	84%	68.306	65%	42.188	40%

Tabelle A.13.: Ergebnisse bei Kommissionierung pro Lagereinheit 2010

Zeitraum	Historisch		GFP		Dyn. Zon.		Min. Plätze		JS	
01.01.-31.03.11	45.228	100%	40.192	89%	34.514	76%	31.082	69%	18.376	41%
01.01.-30.06.11	92.424	100%	80.170	87%	74.744	81%	62.066	67%	37.226	40%
01.01.-30.09.11	142.586	100%	128.978	90%	116.122	81%	97.292	68%	57.352	40%

Tabelle A.14.: Ergebnisse bei Kommissionierung pro Lagereinheit 2011

Zeitraum	Historisch		GFP		Dyn. Zon.		Min. Plätze		JS	
01.01.-31.03.12	65.624	100%	65.820	100%	53.666	82%	50.372	77%	28.466	43%
01.01.-30.06.12	130.228	100%	130.962	101%	106.336	82%	100.876	77%	56.256	43%
01.01.-30.09.12	193.752	100%	191.562	99%	157.714	81%	149.730	77%	83.532	43%

Tabelle A.15.: Ergebnisse bei Kommissionierung pro Lagereinheit 2012

B. Glossar

Artikel Angebotene Versionen der Produkte, die sich unter Umständen auch nur durch ihre Verpackung unterscheiden. Beispiele sind 500g Himbeerjoghurt im Glas, 500g Himbeerjoghurt im Becher, ...

Ausfassung Siehe *Kommissionierung*.

Auslagerung Siehe *Kommissionierung*.

Einzelspiel Kommissionierung einer einzelnen Lagereinheit.

Handling Unit Siehe *Lagereinheit*.

Kardinalität einer Bestellung Die Anzahl der Lagereinheiten, die dieser Bestellung zugeordnet sind. In einem Lager mit einer Lagereinheit pro Lagerplatz entspricht dies auch der Anzahl von Lagerplätzen, die bei der Kommissionierung der Bestellung zu besuchen sind.

Kommissionierplatz Der Platz im Lager, an dem Kommissioniervorgänge beginnen und enden.

Kommissionierung (Synonyme: *Ausfassung*, *Auslagerung*). Der Vorgang der Entnahme einer Lagereinheit aus dem Lager.

Lager Überbegriff für eine Menge von Lagerplätzen. Ein Lager im Sinne eines Lagerhauses kann aufgrund verschieden organisierter Bereiche oder verschiedener Attribute der in ihm enthaltenen Lagerplätze in mehrere kleinere Lager untergliedert werden. Beispiele dafür sind ein Hochregallager, ein Schüttlager, oder auch Teile davon, wie zum Beispiel eine Menge von Lagerplätzen im Hochregallager mit gleichen Abmessungen.

Lagereinheit (Synonyme: *Handling Unit*). Eine Menge von Einheiten eines Artikels, die im Lager als Einheit bewegt werden. Beispielsweise ein Karton oder eine Palette mit einer oder mehreren Einheiten eines Artikels.

Lagerplatz Ein Platz im Lager, auf dem Lagereinheiten gelagert werden.

Langsamdreher Eine Lagereinheit oder ein Artikel mit langer Verweildauer.

Lineares Lager Ein Lager mit nur einer Gasse.

Mehrfachspiel Kommissionierung mehrerer Lagereinheiten auf einmal, oder eine Kombination von Ein- und Auslagerungen in einer Tour.

Misch-Lagereinheit Eine nicht artikelreine Lagereinheit (siehe hierzu auch Kapitel 3.2.5).

Nachlieferung Eine Lieferung von Lagereinheiten, die erst nach Beginn des Betrachtungszeitraums eintrifft.

Perfekt balanciertes Lager Ein Lager, in dem in jeder beliebig gewählten Periode T für jede ausgehende Lagereinheit eine Lagereinheit mit gleicher Verweildauer eintrifft. [GR90, S. 1122]

Produkt Die Grundlage eines Artikels. Beispielsweise Naturjoghurt, Himbeerjoghurt, ...

Schnelldreher Siehe *Schnellläufer*.

Schnellläufer (Synonyme: *Schnelldreher*). Eine Lagereinheit oder ein Artikel mit kurzer Verweildauer.

Sortiment Die Menge aller angebotenen Produkte.

Teilentnahme Die Entnahme eines Teilbestands aus einer Lagereinheit (siehe hierzu auch Kapitel 3.2.5).

Verbrauchsfolgeverfahren Ein Verfahren, das die Verbrauchsreihenfolge für gelagerte Lagereinheiten vorgibt (zum Beispiel FIFO, LIFO, ...).

Verweildauer Die Aufenthaltsdauer im Lager.

artikelbezogen Die durchschnittliche Verweildauer einer Lagereinheit von Artikel A .

lagereinheitsbezogen Die Verweildauer einer individuellen Lagereinheit.

Zone Abgegrenzter Bereich innerhalb eines Lagers.

Literaturverzeichnis

- [Coc05] A. Cockburn. Alistair.Cockburn.us | Hexagonal architecture, 2005. URL <http://alistair.cockburn.us/Hexagonal+architecture>. (Zitiert auf Seite 27)
- [CS91] K. I. Choe, G. Sharp. Small Parts Order Picking: Design and Operation. Technischer Bericht, Georgia Institute of Technology, Atlanta, Georgia, USA, 1991. URL <http://www2.isye.gatech.edu/~mgoetsch/cali/Logistics%20Tutorial/order/article.htm>. (Zitiert auf Seite 3)
- [GHJV94] E. Gamma, R. Helm, R. Johnson, J. Vlissides. *Design Patterns: Elements of Reusable Object-Oriented Software*. Addison-Wesley Professional, Boston, Mass., 1. Auflage, 1994. (Zitiert auf Seite 27)
- [Gla08] M. Glass. *Schnellläuferstrategien in Lagern und Dynamische Zonierung*. Dissertation, Technische Universität Dresden, Dresden, 2008. URL <http://www.qucosa.de/fileadmin/data/qucosa/documents/2523/SchnelllaeuerstrategienInLagern.pdf>. (Zitiert auf den Seiten 1, 2, 3, 4, 5, 17, 18, 25 und 39)
- [GR90] M. Goetschalckx, H. D. Ratliff. Shared Storage Policies Based on the Duration Stay of Unit Loads. *Management Science*, 36(9):1120 – 1132, 1990. URL <http://www.redi-bw.de/db/ebsco.php/search.ebscohost.com/login.aspx%3fdirect%3dtrue%26db%3dbuh%26AN%3d6685411%26site%3dehost-live>. (Zitiert auf den Seiten 4, 5, 7 und 54)
- [MN10] M. Muhsal, C. Nettesheim. LVS Markt-Überblick 2010. Technischer Bericht, IWL AG, 2010. URL <http://www.wmsfinder.com/Downloads/LVS%20Markt-%C3%9Cberblick%202010%20v1.0.pdf>. (Zitiert auf Seite 1)
- [Ski08] S. S. Skiena. *The Algorithm Design Manual*. Springer-Verlag London, 2. Auflage, 2008. (Zitiert auf Seite 18)

Alle URLs wurden zuletzt am 27.01.2013 geprüft.

Erklärung

Ich versichere, diese Arbeit selbstständig verfasst zu haben.

Ich habe keine anderen als die angegebenen Quellen benutzt und alle wörtlich oder sinngemäß aus anderen Werken übernommenen Aussagen als solche gekennzeichnet.

Weder diese Arbeit noch wesentliche Teile daraus waren bisher Gegenstand eines anderen Prüfungsverfahrens.

Ich habe diese Arbeit bisher weder teilweise noch vollständig veröffentlicht.

Das elektronische Exemplar stimmt mit allen eingereichten Exemplaren überein.

(Stefanie Götz)