

Anforderungs-Dokumentation - Hochregallager

Inhaltsverzeichnis

1	Vorwort	1
1.1	Zielgruppe	1
1.2	Autoren	1
1.3	Dokument-Versionen	2
2	Einleitung	2
3	Benutzeranforderungen	2
3.1	Funktionale Anforderungen	2
3.2	Nichtfunktionale Anforderungen	3
3.2.1	Domainspezifische Anforderungen	3
4	System-Architektur	3
5	Systemanforderungen	4
5.1	Funktionale Anforderungen	4
5.2	Nichtfunktionale Anforderungen	4
6	Testing	4
7	Mögliche Szenarien	4
8	Schwerpunkte	5
9	Ziele	5
9.1	Angestrebte Pflichtziele	5
9.2	Wunschziele	5
10	Möglicher Ablauf der Arbeitsschritte	5
11	Sonstiges	6

1 Vorwort

1.1 Zielgruppe

Dieses Dokument beschreibt die Anforderungen an eine Software-Lösung für die Simulation und Optimierung von Hochregallagern im Detail. Es enthält die Aspekte der gesuchten Lösung mit dem Fokus, welcher auf die technische Seite gerichtet ist. Der Leser sollte ein grundlegendes Verständnis von Logistik und Lagerungssystem haben und mit der in diesem Bereich verwendeten Terminologie vertraut sein.

1.2 Autoren

Die Autoren diesen Dokument sind:

- Marc Schärer scham36@bfh.ch

- Arthur van Ommen `vanoa1@bfh.ch`
- Fabian Affolter `affof1@bfh.ch`

1.3 Dokument-Versionen

Version	Autor	Bemerkungen	Datum
0	Team	Skelett	20.09.2013
0.1	Team	erste Version, Dokumentation der Anforderungen	27.09.2013
0.2	Team	überarbeitete Version, Prioritätenliste, Schwerpunkte, u. a.	01.10.2013
0.3	Team	überarbeitete Version nach Besprechung	08.10.2013
0.4	Team	überarbeitete Version nach Besprechung	08.10.2013

2 Einleitung

Ein Hochregallager (HRL) beschreibt ein Lagersystem mit Plätzen in sogenannten Regalen. Hochregallager gibt es in den unterschiedlichsten Ausprägungen. Die grössten Ausführungen besitzen Höhen bis etwa 50 m und können mehreren hunderttausend Plätze besitzen. Oftmals werden direkt Euro-Paletten als Träger für das Lagergut verwendet, ist das Lagergut zu klein, werden häufig spezielle Kunststoff-Behälter benutzt.

Grobgesagt besteht ein Hochregallager aus einer bestimmten Anzahl von Gassen. Eine Gasse wiederum hat links und rechts Lagerplätze und im Freiraum bewegt sich ein Bediengerät. In einem manuellen Hochregallager ist dieser Raum so gross, dass mit einem Gabelstapler zwischen den Regalwänden manövriert werden kann. Bei automatischen Lagern fährt ein Bediengerät, welches von einem Lagerverwaltungssystem seine Befehle bekommt, ohne manuelle Interventionen in der Gasse und liefert das Lagergut zur Entnahmestelle.

Die Hochregallager haben eine hohe Raumnutzung und bei der Erstellung sind hohe Investitionen nötig, da bei kleineren Ausführungen eine Halle um das Hochregallager gebaut werden muss. Bei grossen Varianten wird das Hochregal als Tragstruktur für das Gebäude mitbenutzt.

3 Benutzeranforderungen

3.1 Funktionale Anforderungen

- Definition des Szenarios (statische Parameter)
 - Grundkonfigurationen (Beispiel für ein Lager mit 10000 Fächer)
 - * Anzahl der Lagergassen
 - * Definition der einzelnen Lagergassen (Lagerfächer ein-/beidseitig, Höhe/Breite/Tiefe der Lagergassen und Lagerfächer)
 - Anzahl der Lagergüter (ergibt die benötigte Fächer-Anzahl)
 - Geometrische Bedingungen (maximale Gebäude-Abmessungen oder ähnlich)
- Eingeben der Simulationsparameter (dynamische Parameter)
 - Maximale Masse der Lagergüter

- Physikalische Eigenschaften der Lagergüter
- Geschwindigkeit der beweglichen Elemente (RBG)
- Beschleunigungs- und Verzögerungsverhalten der beweglichen Elemente (RBG)
- Simulationssteuerung
 - Diverse Modi (zeitliche Intervalle, so schnell wie möglich)
 - Definition von Zeiträumen, die betrachtet werden können.
- Szenarienmanagement
 - Laden von vordefinierten Szenarien (gemäss Auflistung in Abschnitt 7 auf Seite 4)
 - Laden von eigenen Szenarien
 - Speichern von erstellten Szenarien

3.2 Nichtfunktionale Anforderungen

- Keine unsinnig grossen (langlaufende) Simulationen
 - Plausibilitätsprüfung vor Beginn
 - Prognose der Berechnungsdauer (ev. anhand von gewonnenen Daten während der Entwicklung)
 - Abbruch, resp. Warnung, bei Überschreitung einer definierten Grenze
- Grafische Darstellung während der Simulation (informativ)
- Sprache der Applikation ist in Englisch
- Ausgabe / Export der Ergebnisse auf Drucker oder als Dokument (z.B. .txt, .csv, .xml, usw.)

3.2.1 Domainspezifische Anforderungen

- Gefahrgut / Brandschutz
- Konformität
- Arbeitssicherheit

4 System-Architektur

- Die Anwendung soll eine reine Clientanwendung sein.
- Trennung von Simulation, Auswertung und Visualisierung: die Simulation soll unabhängig einer gewählten Visualisierung ablaufen. D.h. keine für die Simulation benötigte Logik oder Komponenten innerhalb der Visualisierung. Ebenfalls soll die Simulation unabhängig der gewünschten Auswertungsart ablaufen können.

5 Systemanforderungen

5.1 Funktionale Anforderungen

- Szenario laden
- Szenario simulieren / berechnen
- Simuliertes Szenario auswerten / ausgeben

5.2 Nichtfunktionale Anforderungen

- Lauffähig auf Standard-Hardware
- Nur Standard-Software (JRE, Bibliotheken, etc.)

6 Testing

- Unit tests

7 Mögliche Szenarien

Dieser Abschnitt beschreibt mögliche Szenarien, welche in Simulationen betrachtet werden könnten.

- Maschinenbaufirma im 1-Schichtbetrieb mit Fertigung / Montage / Service – kurze Zugriffszeiten Tagsüber, freie Ressourcen während der Nacht
- Versandhandel im 3-Schichtbetrieb mit Bereitstellung / Konvektionierung – hoher Lagerdurchsatz, 24h-Zugriff für Ein-/Auslagerung
- Gleichzeitiges Ein-/Auslagern, Queue
- Mehrere Ein-/Ausgabeplätze pro Gasse (auf z-Achse)
- Mehrere Regalbediengeräte pro Gasse (auf x-Achse, bei mehreren Ein-/Ausgabeplätzen auch auf z-Achse)
- Mehrere Ladearme pro Regalbediengerät (mehrere vertikal, horizontal [ohne / mit Durchreichemöglichkeit], radial)
- Vorgezogenes Auslagern (Bereitstellung noch im Lager)
- Ausfall einer Gasse, Mehrplatzeinlagerung gleicher Teile in unterschiedlichen Gassen
- Fixe Lagerplatzzuordnung (Reservation, defekte Lagerplätze, abgeschottete Lagerplätze für Gefahrgut)
- Befüllung (initial von leerem Lager / Nachbefüllung) (zufällig chaotisch, zeitoptimiert chaotisch, zugeordnet, positionsoptimiert [ABC])
- Optimierung von bereits belegtem Lager aufgrund Zugriff-History

8 Schwerpunkte

- Simulation / Simulationsauswertung / Simulationsvisualisierung
- Optimierung

9 Ziele

9.1 Angestrebte Pflichtziele

1. Laden eines einfachen Szenarios mit einer Gasse und einigen wenigen Lagerplätzen. Darin soll das Ein- und Auslagern von einigen wenigen Gütern simuliert werden. Die Güter sollen dabei auf fix vorgegebenen Lagerplätzen eingelagert werden.
2. Die Simulation soll in einer einfachen Grafik (2D) visualisiert werden können. Dabei soll die Geschwindigkeit der Simulation an die Visualisierung angepasst werden können.
3. Die Simulation soll angepasst werden, so dass die Güter anhand einer Zugriffshistorie automatisch auf dem jeweils günstigsten Lagerplatz eingelagert werden. Dabei ist die Zugriffszeit entscheidend.
4. Die Simulation soll so angepasst werden, dass auch die physikalischen Eigenschaften der Güter berücksichtigt werden. Dabei sind das Gewicht, die Abmessungen, usw. entscheidend. Diese Eigenschaften beeinflussen die Geschwindigkeit des Laderoboters und ändern somit auch die Entscheidungsfindung zum optimalen Lagerplatz.
5. Das Szenario soll komplexer werden können, z.B. verschieden grosse Paletten oder Gefahrenzonen.
6. Möglichkeit zum Laden einer vorgegebenen Liste von Einlagerungen, jeweils versehen mit einem Zeitstempel. Dadurch können dann mehrere Tage simuliert werden.

9.2 Wunschziele

1. Möglichkeit zum Laden von Operationsplänen zum nächtlichen Vorbereiten der Güter für einen schnellen Zugriff am Morgen beim Auslagern.
2. Das Szenario erweitern, so dass der Laderoboter mehrere Güter auf einmal transportieren und einlagern kann. Ausserdem soll der Laderoboter die Güter untereinander tauschen können. Dies kann bis zu einem Paternoster-Laderoboter führen.
3. Erweiterung der Simulation und der Visualisierung auf die vorgesezte Ladezone.
4. Visualisierung in 3D.

10 Möglicher Ablauf der Arbeitsschritte

1. Laden eines einfachen Szenarios (mit fixen Einstellungen)
2. Berechnung der Simulations-Eckdaten
3. Visuelle Darstellung des Szenario (2D) – Überprüfung des Klassen-Diagramm, Abschätzung der Performance

4. Steigerung der Komplexität der Szenarien (einfache Input-Funktion, ASCII- oder Spreadsheet-Datei) – Szenarien-Management
5. Simulierte Szenarien auswerten / einfache Daten-Ausgabe
6. Optimierung von einzelnen Szenarien (nach einer vorgegebenen Auswertung, nach vorgegebener Strategie, Änderung der Hochregallager-Parameter)
7. Erweiterung der Export-Funktion (Spreadsheet oder ähnlich, für grafische Auswertungen)
8. Erweiterung der visuellen Repräsentation (3D)
9. Ausgabe für die Dimensionierung/Auslegung von Hochregallagern
10. Einbezug der Vorzone in Simulation/Optimierung
11. Multifunktionale Benutzeroberfläche für die Eingabe/Simulation/Auswertung/Auslegung

11 Sonstiges

- Repository: <https://github.com/fabaff/high-rack-warehouse>
- Dokumentation: /docs
- Code: /src (voraussichtlich)