

Daniel Mack, Andreas Bortfeldt

Eine Heuristik für das
mehrdimensionale Bin Packing
Problem

**Wirtschafts-
wissenschaft**

Diskussionsbeiträge der Fakultät für Wirtschaftswissenschaft
Diskussionsbeitrag Nr. 423
April 2008



FernUniversität in Hagen

Eine Heuristik für das mehrdimensionale Bin Packing Problem

Daniel Mack, Andreas Bortfeldt

Diskussionsbeitrag Nr. 423
April 2008

Diskussionsbeiträge der Fakultät für Wirtschaftswissenschaft
der FernUniversität in Hagen

Herausgegeben von der Dekanin der Fakultät

Alle Rechte liegen bei den Autoren

Eine Heuristik für das mehrdimensionale Bin Packing Problem

Daniel Mack, Andreas Bortfeldt

Abstract:

The more-dimensional bin packing problem (BPP) considered here requires packing a set of rectangular shaped items into a minimum number of identical rectangular shaped bins. All items may be rotated and the guillotine constraint has to be respected. A straightforward heuristic is presented that is based on a method for the container loading problem following a wall-building approach and on a procedure for the one-dimensional BPP. 1800 new benchmark instances are introduced for the two-dimensional and for the three-dimensional BPP. The instances include more than 1500 items on average. Applied to these very large instances, the heuristic generates solutions of acceptable quality in short computation times. Moreover, the influence of different instance parameters on the solution quality is investigated by an extended computational study.

Keywords:

Bin packing, Single Bin-Size Bin Packing Problem, SBSBPP, Single Stock-Size Cutting Stock Problem, SSSCSP, heuristic.

Fakultät für Wirtschaftswissenschaft, FernUniversität in Hagen

Profilstr. 8, D-58084 Hagen, BRD

Tel.: 02331/987-4433

Fax: 02331/987-4447

E-Mail: daniel.mack@fernuni-hagen.de

Eine Heuristik für das mehrdimensionale Bin Packing Problem

Daniel Mack, Andreas Bortfeldt

1 Einleitung

Gegenstand dieses Beitrags ist das mehrdimensionale Bin Packing Problem (BPP). Das dreidimensionale BPP sei wie folgt definiert: Gegeben ist eine Menge J von quaderförmigen Packstücken (kurz: Kisten) sowie eine unbegrenzte Anzahl von identischen quaderförmigen Containern (Bins). Jede Kiste ist durch ihre Maße gegeben, nämlich ihre Länge l_j , Breite w_j und Höhe h_j ($j = 1, \dots, n$, $n = |J|$). Jeder Container besitzt die Länge L , die Breite W und die Höhe H . Gesucht ist eine zulässige Anordnung aller Kisten in einer minimalen Anzahl von Containern. Das zweidimensionale BPP (2D-BPP) ergibt sich als Spezialfall des dreidimensionalen BPP (3D-BPP), wenn $H = 1$ sowie $h_j = 1$, $j = 1, \dots, n$, gesetzt wird. Eine Instanz des mehrdimensionalen BPP kann daher stets als Quadrupel (L, W, H, J) notiert werden.

Eine Anordnung, auch als Packplan bezeichnet, gilt als zulässig, falls jede Kiste vollständig in einem Container liegt und parallel zu dessen Seitenflächen angeordnet ist. Je zwei in einem Container liegende Kisten müssen ferner überschneidungsfrei platziert sein.

Wie bei Cutting- und Packing-Problemen (C&P) generell üblich, wird zwischen einer schwach und einer stark heterogenen Kistenmenge unterschieden. Im ersteren Fall existieren nur wenige Kistentypen und relativ viele Exemplare pro Typ, während im letzteren Fall viele Kistentypen und nur wenige Exemplare pro Typ vorhanden sind. Zwei Kisten sind von demselben Typ, wenn sie kongruent sind. Hier werden der schwach und der stark heterogene Fall behandelt. Gemäß der Typologie der C&P-Probleme von Wäscher et al. (2007) geht es also in diesem Beitrag um die Problemtypen Single Stock-Size Cutting Stock Problem (SSSCSP) sowie Single Bin-Size Bin Packing Problem (SBSBPP).

In der Literatur werden vor allem zwei Restriktionen in die Problemstellung einbezogen:

(C1) Orientierungs-Restriktion

Während 90°-Drehungen von Kisten im Allgemeinen zulässig sind, legt die Orientierungs-Restriktion ggf. die Orientierung aller Kisten fest und verbietet ihre Rotation.

(C2) Guillotineschnitt-Restriktion

Diese Restriktion fordert, dass alle platzierten Kisten durch eine Serie von Guillotineschnitten reproduzierbar sind. Ein Guillotineschnitt zerlegt einen Quader in zwei kleinere Quader, wobei die Schnittfläche parallel zu einer Seitenfläche des durchgeschnittenen Quaders liegt.

Mit Bezug auf diese Restriktionen unterscheiden Lodi et al. (1999a) vier Subtypen des BPP:

- RF: Kisten sind um 90° rotierbar (R), Guillotineschnitt-Restriktion ist nicht gefordert (F);
- RG: Kisten sind um 90° rotierbar (R), Guillotineschnitt-Restriktion ist gefordert (G);
- OF: Orientierung aller Kisten ist fix (O), Guillotineschnitt-Restriktion ist nicht gefordert (F);
- OG: Orientierung aller Kisten ist fix (O), Guillotineschnitt-Restriktion ist gefordert (G).

Eine zulässige Lösung bezüglich des Subtyps OG ist auch zulässig hinsichtlich der anderen drei Subtypen. Eine bezüglich des Subtyps OF oder RG zulässige Lösung ist auch zulässig in Bezug auf den Subtyp RF.

Als Packproblem kommt das 3D-BPP bei der kompletten Verstaung einer Menge quaderförmiger Stückgüter in Standardcontainern, Lastkraftwagen oder Eisenbahnwagons vor. Als Zuschnittproblem tritt es beispielsweise auf, wenn geordnete Stahlblöcke mit gewünschten Abmessungen aus größeren Blöcken einheitlichen Formats gesägt werden müssen. Zahlreiche praktische Anwendungen des 2D-BPP erwähnen Lodi et al. (1999a). Sowohl die Orientierungs-Restriktion als auch die Guillotineschnitt-Restriktion sind häufig bei dem Zuschnitt von Material

einzuhalten und sind beispielsweise bedingt durch die Beschaffenheit der Materialoberfläche (etwa infolge Walzens) bzw. durch die eingesetzte Schneidetechnologie.

In diesem Beitrag wird eine Heuristik für das mehrdimensionale BPP und den Subtyp RG vorgestellt, die auf große Probleminstanzen mit mehr als 1000 Kisten anwendbar ist. Der folgende Abschnitt gibt einen Überblick der Literatur zum mehrdimensionalen Bin Packing Problem. Abschnitt 3 beschreibt die Heuristik. In Abschnitt 4 werden neue große Benchmarkinstanzen für das 3D-BPP und das 2D-BPP eingeführt und die Ergebnisse eines umfangreichen numerischen Tests ausgewertet. Abschnitt 5 fasst den Beitrag zusammen.

2 Literaturüberblick

Das Bin Packing Problem ist (unabhängig von der Dimensionalität) NP-hart. Daher werden in der Literatur hauptsächlich heuristische Verfahren zu seiner Lösung vorgeschlagen. Dabei handelt es sich zum Teil um Konstruktionsheuristiken (CH) oder Verbesserungsheuristiken (IH), in wachsendem Maße jedoch um Metaheuristiken. Dazu gehören vor allem Verfahren der Nachbarschaftssuche wie Simulated Annealing (SA), Tabu Search (TS) oder Guided Local Search (GLS), aber auch Constrained Programming (CP). Für eine Einführung in metaheuristische Suchstrategien sei auf Glover und Kochenberger (2003) verwiesen. Branch and Bound (B&B) und alternative Ansätze der Baumsuche (Tree Search, TRS; Branch and Cut, B&C) werden in exakten und teilweise auch in heuristischen Verfahren angewendet. Tabelle 1 stellt eine Auswahl der für mehrdimensionale Bin Packing Probleme verschiedener Varianten entwickelten Verfahren vor.

Tab. 1. Verfahren für das mehrdimensionale Bin Packing Problem mit rechteckigen Stücken.

Autoren, Quelle	Dimension	#Cont.-Typen	Subtyp	Verfahrenstyp	Max. Größe n
Bengtsson (1982)	2	1	RF	IH	200
Berkey und Wang (1987)	2	1	OF	CH	100
Ivancic et al. (1989)	3	M	RF	CH	180
El Bouri et al. (1994)	2	1	RF	TRS	200
Bischoff und Ratcliff (1995)	3	1	RF	CH	180
Martello und Vigo (1998)	2	1	OF	B&B, exakt	100
Lodi et al. (1998)	2	1	RG	TS	164
Lodi et al. (1999a)	2	1	alle	TS	100
Lodi et al. (1999b)	2	1	OG	TS	100
Martello et al. (2000)	3	1	OF	B&B, exakt	90
Terno et al. (2000)	3	1	RF	B&B	-
Bortfeldt (2000)	3	M	RF	CH, IH	180
Dell'Amico et al. (2002)	2	1	RF	B&B, exakt	100
Faroe et al. (2003)	3	1	OF	GLS	100
Eley (2003)	3	M	RF	TRS	180
de Castro Silva et al. (2003)	3	1	OF	CH	90
Pisinger und Sigurd (2007)	2	1	OF,OG	CP, B&C exakt	100

Pro Verfahren wird neben der Quelle die behandelte Variante des BPP durch die Dimension, die Anzahl der Containertypen (es erscheint nur 1 oder M , M steht für den Fall mehrerer Containertypen) und den Subtyp im oben definierten Sinn charakterisiert. Man beachte, dass bei mehreren Containertypen anstelle der Binzahl meist die gesamten Containerkosten zu minimieren sind. Ferner werden der Verfahrenstyp, ggf. ein Hinweis auf ein exaktes Verfahren und die maximale berechnete Problemgröße, definiert durch die Kistenanzahl n , angegeben. Für eine umfassende Bibliographie zum Bin Packing Problem sei auf Coffman et al. (2004) verwiesen.

In der Literatur wird das mehrdimensionale BPP bisher vergleichsweise selten behandelt. Insbesondere wurden für das 3D-BPP bisher nur wenige Lösungsverfahren vorgestellt. Nicht nur die exakten, sondern auch die heuristischen Verfahren wurden fast ausschließlich anhand kleinerer und mittlerer Instanzen mit nicht mehr als 200 Stücken getestet. Auffällig ist weiterhin, dass in den verwendeten Benchmarkinstanzen die Kisten im Vergleich zu den Containern in der Regel relativ groß sind. So passen etwa bei den Instanzen von Ivancic et al. (1989) nur ca. 10 Kisten in einen Container. Dies dürfte etlichen Anwendungen nicht entsprechen, man denke etwa an das Verpacken von Haushaltsgeräten in Standardcontainern. Ferner wurden die Verfahren meist entweder nur anhand schwach heterogener oder nur mittels stark heterogener Instanzen getestet.

Daher erscheint es sinnvoll, eine Heuristik zu entwickeln, die in der Lage ist, große Instanzen (mit tausend oder mehr Kisten) des mehrdimensionalen BPP zu berechnen. Bei neuen Benchmarkinstanzen entsprechenden Umfangs sollten auch Instanzparameter wie z. B. die mittlere relative Kistengröße (bezogen auf die Containergröße) variiert werden, um den Einfluss solcher Parameter auf die Lösungsqualität empirisch untersuchen zu können.

3 Die Heuristik

Die Heuristik wird nachfolgend für den 3D-Fall beschrieben. Sie beinhaltet folgende fünf Module bzw. Teilverfahren (vgl. Abb. 1):

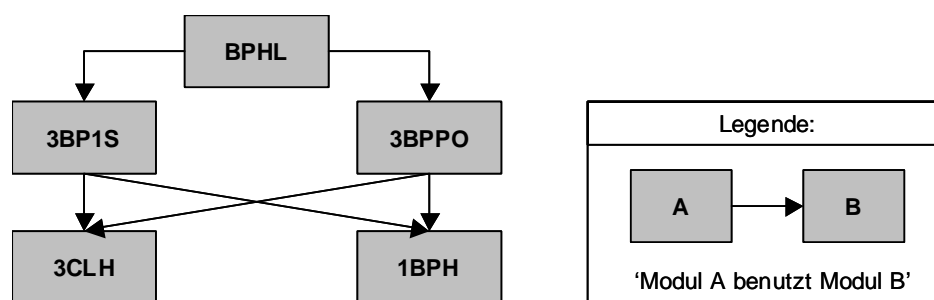


Abb. 1. Modularer Aufbau der 3D-BPP-Heuristik.

- **3CLH**: ein Verfahren zur Lösung des dreidimensionalen Container Loading Problems (CLP). Das CLP kann wie folgt definiert werden: Gegeben seien eine Kistenmenge J sowie ein einzelner Container mit der Länge L , Breite W und Höhe H . Es gelten die gleichen Zulässigkeitsbedingungen für Kistenanordnungen wie bei dem mehrdimensionalen BPP. Gesucht wird eine zulässige Anordnung einer Teilmenge der Kisten von J , welche das verpackte Volumen maximiert. Eine CLP-Instanz I_{CL} kann als Quadrupel (L, W, H, J) notiert werden. Als CLP-Verfahren kommt die Heuristik von Pisinger (2002) zum Einsatz.
- **1BPH**: ein Verfahren zur Lösung des 1D-BPP. Genutzt wird die Heuristik von Bortfeldt (2005) (vgl. auch Bortfeldt 2007).
- **3BPIS**: eine Heuristik zur Erzeugung einer einzelnen Lösung für das 3D-BPP unter Verwendung einer gewissen Parametrisierung.
- **3BPPO**: eine Heuristik zur Nachoptimierung einer mit **3BPIS** erzeugten Lösung.
- **BPHL**: das Hauptmodul der 3D-BPP-Heuristik (H wie Heuristik, L wie layer).

Im Anschluss werden die Teilverfahren im Einzelnen betrachtet.

3.1 Das CLP-Verfahren (Modul 3CLH)

Die CLP-Heuristik von Pisinger (2002) folgt dem schichtbildenden Ansatz. Jeder erzeugte Packplan für einen Container besteht i. Allg. aus mehreren Schichten (vgl. Abb. 2). Diese folgen in Längsrichtung des Containers lückenlos aufeinander. Jede einzelne Schicht besteht aus einem

Quader $l_i \times W \times H$, in dem eine oder mehrere Kisten platziert sind; das Maß l_i wird als Schichtdicke bezeichnet. Die Schichtauslastung ist der Quotient aus dem verpackten Volumen und dem Schichtvolumen, analog ist die Containerauslastung definiert. Eine Schicht besteht im Allgemeinen aus mehreren sogenannten horizontalen und/oder vertikalen Streifen. Jeder horizontale Streifen besteht dabei aus einzelnen Kisten, die lückenlos nebeneinander in der Schicht angeordnet sind. Bei einem vertikalen Streifen sind die einzelnen Kisten übereinander angeordnet (siehe Abb. 2).

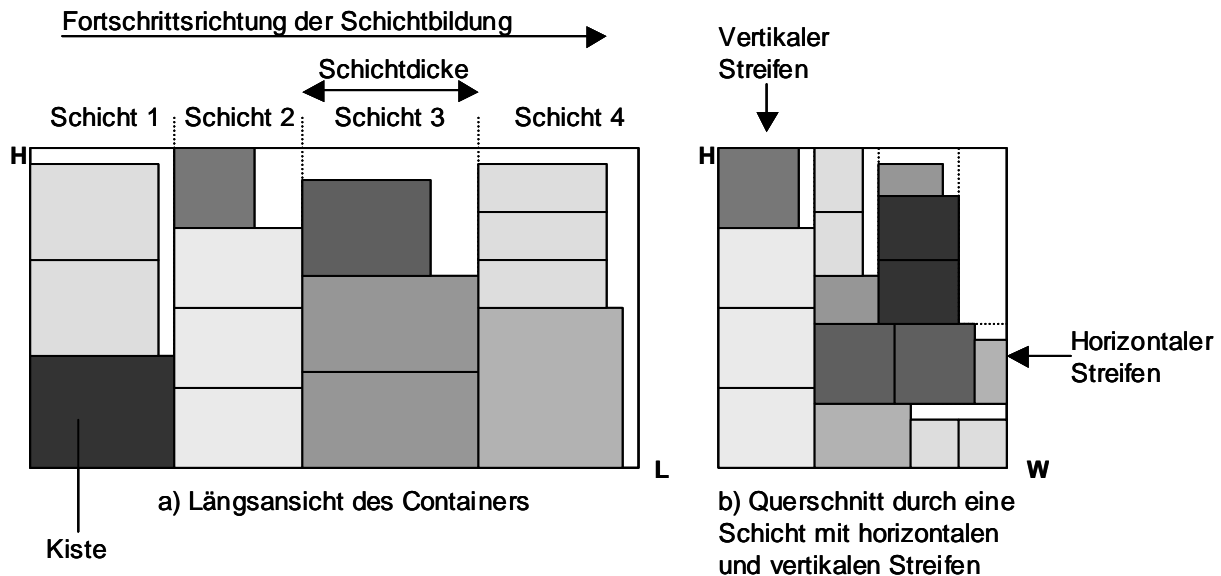


Abb. 2. Struktur einer mit 3CLH erzeugten Lösung.

Die CLP-Heuristik ist als Baumsuche-Verfahren ausgelegt, wobei Verzweigungen auf Schichtebene und Streifenebene verschachtelt durchgeführt werden. Bei jeder Verzweigung auf Schichtebene ist zunächst eine geeignete Schichttiefe zu ermitteln. Hierbei werden besonders große und besonders häufig vorkommende Kistenmaße bevorzugt als Schichtdicke verwendet. Um die Suche nach geeigneten Lösungen nicht ausufern zu lassen, wird die Anzahl der probierten Schichttiefen der jeweils nächsten Schicht mit dem Parameter *cut3* beschränkt. Auf Streifenebene geht man ähnlich vor: für jeden neuen Streifen wird die Anzahl der zu prüfenden Streifenbreiten mit Hilfe des Parameters *cut2* beschränkt. Für eine gegebene Schichtdicke und Streifenbreite kann das CLP so auf ein Knapsack Problem (1D-CLP) reduziert werden, welches mit einem geeigneten integrierten Verfahren effizient gelöst wird.

3.2 Das 1D-BPP-Verfahren (Modul 1BPH)

Das 1D-BPP-Verfahren sei nur knapp charakterisiert. Eine gegebene 1D-BPP-Instanz wird zunächst mittels der bekannten Heuristik "First Fit Decreasing" (FFD) sowie mehreren auf FFD basierenden, aber leistungsfähigeren Konstruktions- sowie Verbesserungsheuristiken bearbeitet. Das Verfahren bricht ab, sobald eine Optimallösung erzielt wurde, die ggf. anhand verschiedener der Literatur entnommener bounds verifiziert wird. Wurde zuvor keine Optimallösung erkannt, so wird abschließend ein B&B-Verfahren eingesetzt, welches eine gegebene Teillösung binweise expandiert. Charakteristisch für das B&B-Verfahren ist, dass für verschiedene Instanztypen bzgl. der mittleren relativen Itemgröße verschiedene Expansionsstrategien verwendet werden; so gibt es z.B. für sogenannte Triplet-Instanzen (vgl. Bortfeldt 2005) eine gesonderte Fortsetzungsstrategie.

3.3 Erzeugung einzelner Lösungen für das 3D-BPP (Modul 3BP1S)

Eine Pseudocodedarstellung der Heuristik *3BP1S* wird in Abb. 3 gegeben. Die Heuristik *3BP1S* folgt anfangs einem bereits von Bischoff und Ratcliff (1995) realisierten simplen Ansatz: eine einzelne Lösung der gegebenen 3D-BPP-Instanz wird ermittelt, indem solange weitere Bins mit dem CLP-Algorithmus *3CLH* gefüllt werden, bis keine Kisten aus *J* mehr übrig sind. Der Inputparameter *nbbest* bezeichnet einen bereits vorher ermittelten Bestwert für die Instanz, *cut3* und *cut2* die oben beschriebenen Parameter des CLP-Verfahrens. *3BP1S* bricht ab, falls feststeht, dass der bisherige Bestwert *nbbest* nicht unterboten werden kann. In diesem Fall wird keine (neue) Lösung zurückgegeben. Man beachte, dass die bei einmaliger Ausführung von *3BP1S* erzeugte Lösung durch die Nutzung der Parameter *cut3* und *cut2* gekennzeichnet ist.

```

procedure 3BP1S (in: Problem Daten  $L, W, H, J$ ; Parameter cut3, cut2, bisheriger Bestwert nbbest;
out: 3D-BPP-Lösung sb)
{initialisiere}
sb :=  $\emptyset$ ; {3D-BPP-Lösung}
nb := 0; {Zähler für Bins}
R := J; {Restvorrat an Kisten}
VR := gesamtes Volumen des Restvorrates R;
{ermittle 3D-BPP-Lösung durch wiederholtes Beladen eines Containers}
while  $R \neq \emptyset$  do
    if  $nb + \lceil VR / (L \cdot W \cdot H) \rceil \geq nbbest$  then
        setze sb :=  $\emptyset$ ; break; {es kann kein neuer Bestwert erreicht werden}
    endif;
    berechne Lösung s für CLP-Instanz  $I_{CL} := (L, W, H, R)$  mittels 3CLH und Parametern cut3, cut2;
    sb := sb  $\cup$  s; {erweitere 3D-BPP-Lösung um ein gefülltes Bin}
    nb := nb + 1;
    aktualisiere Restvorrat R und Volumen VR;
endwhile;
{verbessere 3D-BPP Lösung durch Umverteilung der Schichten}
if  $sb \neq \emptyset$  and  $R = \emptyset$  then {es wurde eine komplette zulässige Lösung sb ermittelt}
    sei  $\{l_i\}$  die Menge der Schichtdicken in der Lösung sb;
    if  $\sum_i l_i \leq (nb - 1)L$  then {Verbesserung durch Umverteilung eventuell möglich}
        berechne Lösung s1 für die 1D-BPP-Instanz gegeben durch L und  $\{l_i\}$  mittels 1BPH;
        if Anzahl der Bins in Lösung s1 < nb then
            führe eine Umverteilung der Schichten in sb gemäss s1 durch;
        endif;
    endif;
endif;
end.

```

Abb. 3. Algorithmus der Heuristik *3BP1S*.

Nach Ermittlung einer neuen Lösung *sb* wird sichergestellt, dass deren Schichten optimal auf die Bins verteilt sind. Es kann in der Tat vorkommen, dass durch eine Umverteilung der Schichten mindestens ein weiteres Bin eingespart werden kann. Dies folgt aus dem Greedy-Ansatz des lokal optimierenden Beladens der einzelnen Bins mit dem CLP-Verfahren, der zu einem suboptimalen Resultat für die 3D-BPP-Instanz führen kann. Sei $\{l_i\}$ die Menge der Schichtdicken in der Lösung *sb*. Falls Chancen auf eine weitere Bineinsparung durch Umverteilung der Schichten bestehen, so wird die 1D-BPP-Instanz $(L, \{l_i\})$ mittels des Verfahrens *1BPH* gelöst. Wird eine 1D-BPP-Lösung *s1* mit einer Binanzahl gefunden, die die Binanzahl *nb* von *sb* unterbietet, so werden die Schichten in den 3D-Bins von *sb* entsprechend umverteilt, um auf diese Weise eine Bineinsparung auch für die 3D-Lösung zu erreichen.

3.4 Hauptmodul der 3D-BPP-Heuristik (Modul BPHL)

Abb. 4 beschreibt das Hauptmodul der 3D-BPP-Heuristik. Es wird eine Vielzahl von Lösungen für die gegebene 3D-BPP-Instanz erzeugt, indem das Parameterpaar $(cut3, cut2)$ des integrierten 3D-CLP-Verfahrens systematisch variiert wird, für jedes Wertepaar mittels der Prozedur *3BPIS* zwei weitere Lösungen generiert werden und die beste Lösung *sbest* ggf. aktualisiert wird. Bei dem ersten Aufruf von *3BPIS* für ein gewisses Wertepaar $(cut3, cut2)$ ist die Fortschrittsrichtung der Schichten pro Bin die Containerbreite W , d.h. die Schichten folgen (anders als ursprünglich für das CLP-Verfahren beschrieben, vgl. 3.1) längs der Containerbreite W aufeinander. Bei dem jeweils zweiten *3BPIS*-Aufruf ist die Fortschrittsrichtung die Containerlänge L . Die Wahl der Containerhöhe als dritte denkbare Fortschrittsrichtung wäre redundant, da innerhalb des CLP-Verfahrens bereits horizontale und vertikale Streifen gleichermaßen berücksichtigt werden. Der Wechsel der Fortschrittsrichtung wird durch die Vertauschung von Containerlänge und -breite beim Aufruf der Heuristik *3BPIS* erzwungen. Der an *3BPIS* übergebene bisherige Bestwert *nbbest* ermöglicht ggf. einen vorzeitigen Abbruch (vgl. 3.3).

```

procedure BPHL (in: Problem Daten  $L, W, H, J$ , Parameter:  $cut3max, cut2max$ ;
                out: beste 3D-BPP-Lösung sbest)
{erzeuge Lösungen für verschiedene Parametersätze}
nbbest :=  $|J|$ ; {Initialwert}
for  $cut3 := 1$  to  $cut3max$  by 1 do                                {Anzahl der Verzweigungen auf Schichtebene}
    for  $cut2 := cut3^2$  to  $cut2max$  by  $cut3^2$  do                {Anzahl der Verzweigungen auf Streifenebene}
        for  $sdir := 1$  to 2 by 1 do                                {Fortschrittsrichtung der Schichten}
            if  $sdir = 1$  then
                setze  $LL := W; WW := L$ ; {Fortschrittsrichtung ist Containerbreite  $W$ }
            else
                setze  $LL := L; WW := W$ ; {Fortschrittsrichtung ist Containerlänge  $L$ }
            endif;
            rufe 3BPIS( $LL, WW, H, J, cut3, cut2, nbbest, sb$ );
            if  $sb \neq \emptyset$  and  $Binanzahl(sb) < nbbest$  then
                 $sbest := sb; nbbest := Binanzahl$  von sbest;
                 $sdirbest := sdir$ ;
            endif;
        endfor  $sdir$ ;
    endfor  $cut2$ ;
endfor  $cut3$ ;

{führe Nachoptimierung der bisher besten Lösung durch}
rufe 3BPPO( $L, W, H, J, cut3max, cut2max, sdirbest, sbest$ );
end.

```

Abb. 4. Algorithmus des Hauptmoduls *BPHL*.

BPHL ist auf große Probleminstanzen mit Kistenvorräten von 1000 oder mehr Kisten zugeschnitten. Um auch für eine große 3D-BPP-Instanz möglichst schnell eine Ausgangslösung zu ermitteln, erhalten die Parameter anfangs minimale Werte ($cut3 = 1, cut2 = 1$). Bei den folgenden Aufrufen werden beide Parameter dann nach und nach inkrementiert, wodurch der Suchaufwand steigt. Der Parameter $cut3$ wird jeweils um 1 erhöht, während $cut2$ in Schritten von $cut3^2$ gesteigert wird. Bei Erreichen einer vorgegebenen Zeitschranke $t1$ bricht die Erzeugung von Lösungen für verschiedene Parametersätze ab. Danach wird die beste gefundene Lösung *sbest* noch einer Nachoptimierung mittels der Heuristik *3BPPO* unterzogen. Der Aufwand der Nachoptimierung wird durch eine weitere Zeitschranke $t2$ beschränkt.

3.5 Nachoptimierung einer Lösung für das 3D-BPP (Modul 3BPPO)

Bei jeder mittels der Heuristik *3BPIS* erzeugten Lösung existiert eine einheitliche Fortschrittsrichtung der Schichten, diese folgen entweder längs der Containerlänge L oder längs der Containerbreite W aufeinander. Die Nachoptimierungs-Heuristik *3BPPO* zielt darauf ab, beide Fortschrittsrichtungen in ein und derselben Lösung zu kombinieren. Ausgehend von einer mittels *3BPIS* erzeugten Lösung werden dazu mehrere Schichten durch neue Schichten ersetzt, welche in der jeweils alternativen Fortschrittsrichtung aufeinander folgen.

Abb. 5 veranschaulicht die einzelnen Schritte der Nachoptimierung anhand eines Beispiels. Hierbei werden zunächst die vier Schichten der übergebenen Lösung mit den geringsten Volumenauslastungen ermittelt und aufgelöst; diese liegen in lauter verschiedenen Bins. In Schritt 2 werden dann die verbleibenden Schichten auf möglichst wenige Bins verteilt. Die entsprechende eindimensionale BPP-Instanz wird wiederum mit dem Verfahren *1BPB* gelöst. In Schritt 3 wird geprüft, ob die in der Fortschrittsrichtung der Ausgangslösung gelegenen Restdistanzen in den belegten Bins ausreichen, um weitere Kisten zu verstauen. Falls möglich, werden die Resträume dieser Bins mit dem CLP-Algorithmus *3CLH* gefüllt. Dabei wird die Fortschrittsrichtung für die neuen Schichten alternativ zur Fortschrittsrichtung der Ausgangslösung gewählt.

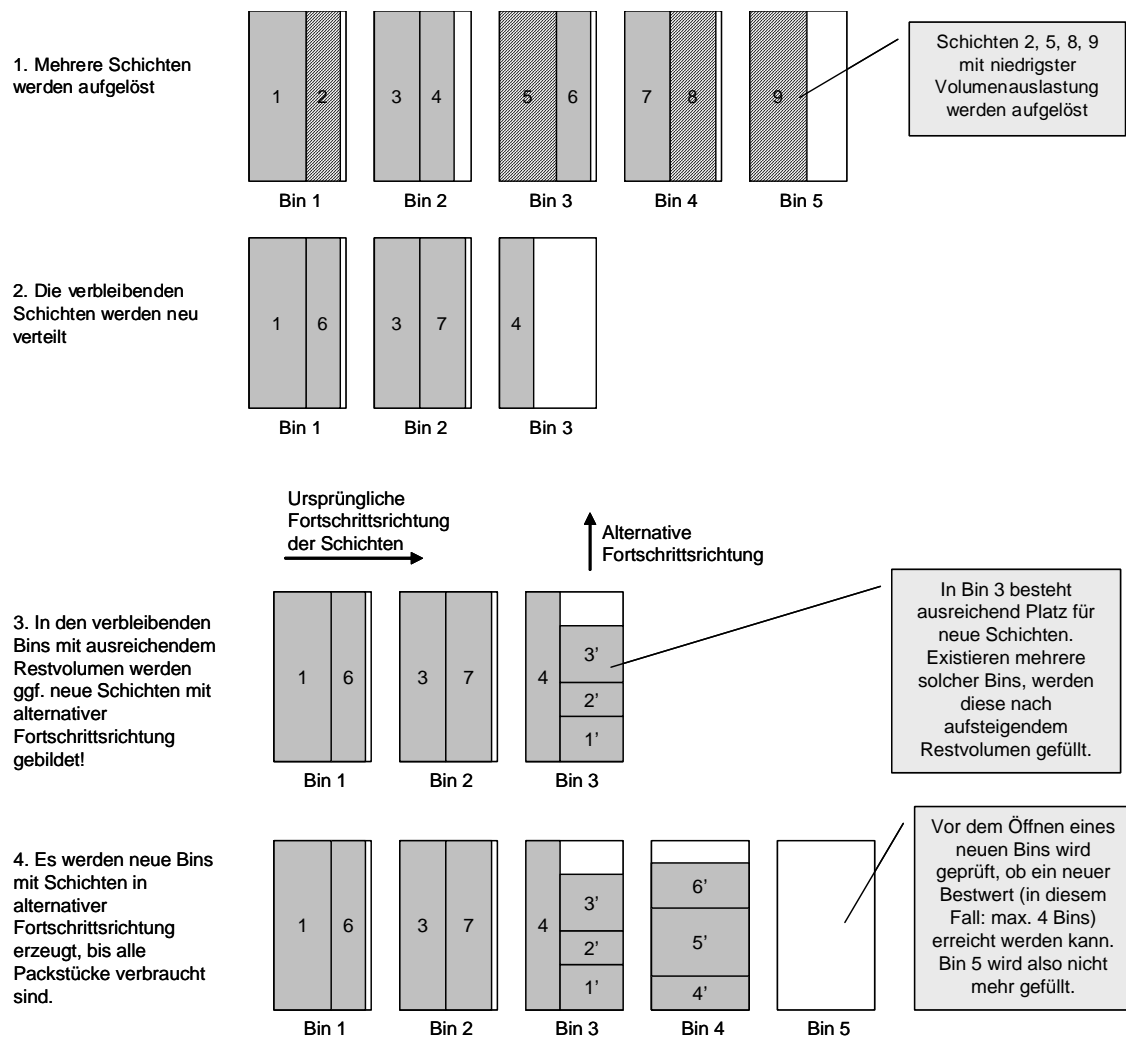


Abb. 5. Vier Schritte der Nachoptimierung bei vorgegebener Anzahl aufzulösender Schichten.

Im vierten und letzten Schritt werden gegebenenfalls weitere Bins mit dem CLP-Verfahren ebenfalls gemäß der alternativen Fortschrittsrichtung beladen, bis der gesamte Kistenvorrat verbraucht ist.

Abb. 6 beschreibt die Nachoptimierung im Detail. In der Hauptschleife wird die Anzahl k der aufzulösenden Schichten von 1 bis $nl-1$ inkrementiert, wobei nl für die Anzahl der Schichten der Lösung sb steht. Für eine gegebene Anzahl k aufzulösender Schichten werden jeweils die vier Schritte aus Abb. 5 durchgeführt. Der Suchaufwand ist anfänglich gering, da die Anzahl der neu zu packenden Kisten relativ klein ist, und steigt dann kontinuierlich an. Die Nachoptimierung bricht bei Erreichen einer vorgegebenen Zeitschranke $t2$ ab.

Die Heuristik *3BPPO* erfordert erheblichen Aufwand hauptsächlich wegen der umfänglichen Variation der Anzahl aufgelöster Schichten. Sie wird daher nur einmalig zur Nachoptimierung der vorher erreichten Bestlösung eingesetzt.

Abschließend sei auf die Einhaltung der Guillotineschnitt-Bedingung eingegangen. Diese wird auf der Ebene einer einzelnen Schicht durch die integrierte CLP-Heuristik aufgrund der Streifenstruktur eingehalten (vgl. Abb. 2). Ihre Einhaltung auf der Ebene kompletter Lösungen resultiert aus der Schichtstruktur. Die Verwendung verschiedener Fortschrittsrichtungen in einer Lösung führt nicht zu einer Verletzung der Guillotinierbarkeit (vgl. Abb. 5).

```

procedure 3BPPO ( in: Problem Daten  $L, W, H, J$ , Param.  $cut3, cut2$ , Fortschrittsrichtung  $sdir$ ; inout: Lösung  $sb$ )
if  $sdir = 1$  then   $LL := W; WW := L$ ; {Fortschrittsrichtung von  $sb$  ist Containerbreite  $W$ }
else   $LL := L; WW := W$ ; {Fortschrittsrichtung von  $sb$  ist Containerlänge  $L$ }
endif;
 $sbold := sb$ ;
sei  $(l_1, l_2, \dots, l_{nl})$  die aufsteigend nach Schichtauslastung sortierte Folge der Dicken der Schichten in  $sbold$ ;
for  $k := 1$  to  $nl-1$  do
    {Schritt 1}
    löse die ersten  $k$  Schichten von  $sbold$  auf, setze  $R :=$  Menge der Kisten der Schichten  $1, \dots, k$  von  $sbold$ ;
    {Schritt 2}
    löse die 1D-BPP-Instanz  $(LL, \{l_j | j = k+1, \dots, nl\})$  mittels 1BPH und bestimme die Lösung  $s1$  mit  $nb_0$  Bins;
    erzeuge die 3D-BPP-Rumpflösung  $sbnew$  durch Kombination
        der Schichten  $k+1, \dots, nl$  von  $sbold$  in  $nb_0$  3D-Bins gemaess  $s1$ ;
    sortiere die Bins in  $sbnew$  aufsteigend nach der freien Restdistanz in Fortschrittsrichtung  $sdir$ ;
    {Schritt 3}
    for each Bin  $ib$  in  $sbnew$  do
        if freie Restdistanz  $lfree(ib) \geq$  minimales Kistenmass in  $R$  then
            {bilde Schichten mit zu  $sdir$  alternativer Fortschrittsrichtung im Bin  $ib$ }
            berechne Lösung  $s$  für CLP-Instanz  $I_{CL} = (WW, lfree(ib), H, R)$  mittels
                3CLH mit Parametern  $cut3, cut2$ ;
            ergänze Anordnung im Bin  $ib$  durch  $s$ ;
            entferne aus  $R$  die in  $s$  platzierten Kisten;
        endif;
    endfor;
    {Schritt 4: Fülle zusätzliche Bins mit Schichten in zu  $sdir$  alternativer Fortschrittsrichtung}
    rufe 3BPIS( $WW, LL, H, R, cut3, cut2$ ,  $Binanzahl(sb) - nb_0, sbr$ );
    if Lösung  $sbr \neq \emptyset$  then
         $sbnew := sbnew \cup sbr$ ;
        if ( $Binanzahl(sbnew) < Binanzahl(sb)$ ) then  $sb := sbnew$ ; endif;
    endif;
endfor;
end.

```

Abb. 6. Algorithmus der Heuristik *3BPPO*.

4 Numerischer Test

Die Heuristik wurde in C programmiert und mit Hilfe eines Standard-PC (1,86 GHz Intel mit 500 MB RAM) getestet. Im Anschluss werden zunächst neue Benchmarkinstanzen eingeführt, bevor die Testergebnisse präsentiert und analysiert werden.

4.1 Probleminstanzen

Ausgehend von den CLP-Benchmarkinstanzen von Bischoff und Ratcliff (1995) bzw. Davies und Bischoff (1998) (kurz: BR-Instanzen) werden insgesamt 1800 Bin Packing-Instanzen abgeleitet. Die BR-Instanzen gliedern sich in 15 Testfälle mit je 100 Probleminstanzen. Pro Testfall ist die Anzahl der Kistentypen für alle Instanzen identisch. Die Kistenvorräte der Testfälle variieren von schwach zu stark heterogen. Die Containermaße entsprechen durchweg den Innenmaßen eines 20-Fuss-Standard-Containers.

Die neuen Benchmarkinstanzen für das BPP sollten folgende Anforderungen erfüllen:

- Es sollen sowohl 3D- als auch 2D-Probleminstanzen bereitgestellt werden.
- Nur große Instanzen sind gefragt, die in der Regel mindestens 1000 Kisten umfassen sollten.
- Es sollen verschiedene Größenverhältnisse der Kisten im Vergleich zum Container getestet werden.
- Die Kistenmengen der Instanzen sollen verschiedene Heterogenitätsgrade besitzen.
- Um die Vergleichbarkeit der Resultate zu erleichtern, soll der continuous lower bound LO der Binanzahl (auch: material lower bound) für große Instanzgruppen einen einheitlichen Wert besitzen. Der bound ist gemäß $LO = \lceil \text{gesamtes Kistenvolumen} / \text{Containervolumen} \rceil$ definiert.

Die Instanzerzeugung unter Beachtung der genannten Anforderungen wird zunächst für den 3D-Fall beschrieben:

Containergröße: Drei Containergrößen werden betrachtet, wobei das Containervolumen 100%, 25% und 15% des Containervolumens der BR-Instanzen beträgt. Die drei Containergrößen werden nachfolgend auch mit "100%", "25%" und "15%" bezeichnet. Bei den Varianten 25% und 15% werden die Innenmaße des 20-Fuss-Standard-Containers jeweils proportional angepasst. Die Kistengrößen werden unverändert von den BR-Instanzen übernommen. Daher ergeben sich verschiedene Verhältnisse des mittleren Kistenvolumens zum Containervolumen.

Kistentypen: Bei der Bereitstellung von Kistentypen werden zwei Modi unterschieden. In Modus 1 werden für eine 3D-BPP-Instanz alle Kistentypen einer BR-Instanz übernommen. Wie bei den BR-Instanzen kommen daher Kistenvorräte mit 3 bis 100 Kistentypen vor. In Modus 2 werden die Kistentypen einer 3D-BPP-Instanz durch Zusammenfassung der Kistentypen von 10 konsekutiven BR-Instanzen eines Testfalls gewonnen (Instanzen 1-10, 11-20, usw.). Die Anzahl an Kistentypen wird daher im Vergleich zu den BR-Instanzen verzehnfacht. Die Instanzen des Modus 2 haben folglich zwischen 30 und 1000 Kistentypen und besitzen im Vergleich zu den Instanzen des Modus 1 eine viel höhere Heterogenität.

Kistenanzahlen: Zuletzt sind pro Instanz die Kistenzahlen aller vorkommenden Kistentypen zu bestimmen. Die Kistenanzahlen werden derart ermittelt, dass der lower bound LO abhängig von der Containervariante folgende Werte annimmt: $LO = 10$ für Containervariante 100%, $LO = 40$ für Containervariante 25% und $LO = 40$ für Containervariante 10%. Die Kistenanzahlen für die einzelnen Kistentypen werden zufällig bestimmt; ausnahmsweise kann es vorkommen, dass für einen Kistentyp keine Kiste ermittelt wird, dieser also entfällt.

Pro Modus der Kistentypauswahl (1 oder 2) und Containergröße werden für jeden der 15 BR-Testfälle jeweils 10 3D-BPP-Instanzen generiert. Für Modus 1 werden jeweils nur die ersten 10 Testinstanzen jedes BR-Testfalls verwendet, während Modus 2 sämtliche 100 Instanzen für die Erzeugung von 10 neuen Testinstanzen benötigt. Insgesamt werden daher $2 \cdot 3 \cdot 15 \cdot 10 = 900$ 3D-BPP-Instanzen erzeugt. Die Generierung von 2D-BPP-Instanzen verläuft analog wie im 3D-Fall. Anders ist lediglich, dass pro Instanz die Höhe des ursprünglichen 20-Fuss-Containers sowie pro Kistentyp jeweils eine zufällig bestimmte Kistendimension auf 1 gesetzt wird. Wie im 3D-Fall werden 900 2D-BPP-Instanzen generiert. Tabelle 2 stellt die

Systematik und die Bezeichnungen der 12 neuen Testfälle mit je 150 Instanzen vor. Anders als bei den BR-Instanzen für das CLP werden hier alle Instanzen einer Dimension, eines Kistentyp-Modus und einer Containergröße zu einem Testfall zusammengefasst. Bezüglich der Instanzgrößen gelten folgende Angaben: bis auf 10 Instanzen beinhaltet jede Instanz mehr als 1000 Packstücke. Die mittlere Packstückanzahl über alle 1800 Instanzen beträgt 1566, im 2D-Fall gehören im Mittel 1646 Rechtecke und im 3D-Fall 1486 Kisten zu einer Instanz. Die kleinste Instanz beinhaltet 792 und die größte 4232 Packstücke. Die insgesamt 1800 BPP-Instanzen werden demnächst unter www.fernuni-hagen.de/WINF zum Download bereitgestellt.

Tab. 2. Systematik und Bezeichnungen der 12 BPP-Testfälle.

Testfälle (je 150 Instanzen)	Container- volumen	100%	25%	15%
	L0	10	40	100
3D	Modus 1	MB01	MB03	MB05
	Modus 2	MB02	MB04	MB06
2D	Modus 1	MB07	MB09	MB11
	Modus 2	MB08	MB10	MB12

4.2 Testresultate

Sämtliche Testergebnisse wurden mit folgenden Parameterwerten berechnet, welche sich in Vortests als geeignet erwiesen haben: $t1 = 60s$, $t2 = 60s$, $cut3max = 4$, $cut2max = 8$. Tabelle 3 enthält die Ergebnisse der Heuristik für alle 1800 Probleminstanzen und sei wie folgt erläutert:

- Jede Zeile repräsentiert die Mittelwerte über alle 150 Testinstanzen eines Testfalls (vgl. 4.1).
- Die ersten drei Spalten enthalten die Dimensionalität und Bezeichnung des jeweiligen Testfalls sowie den zugehörigen (konstanten) Wert des continuous lower bound $L0$.
- "*BPHL*" steht für die Vollversion der Heuristik inklusive Nachoptimierung, "*BPHL-*" steht für die reduzierte Version ohne Nachoptimierung.
- Pro Version und Testfall werden die mittlere Anzahl an benötigten Bins (nb), die relative Abweichung $gap = (nb-L0)/L0$ (in %) vom bound $L0$ sowie die durchschnittliche Volumenauslastung $vutil$ (in %) in allen Bins angegeben. Für die Vollversion *BPHL* werden außerdem Anzahl und Prozentsatz der Instanzen (n_{opt}) angegeben, in denen der Optimalwert $L0$ erreicht wurde, sowie die mittlere Rechenzeit pro Instanz in (1.86 GHz-)Sekunden.
- In den Spalten "Wirkung von 3BPPO" werden die Ergebnisdifferenzen der Versionen *BPHL* und *BPHL-* zusammengefasst. Speziell gibt n_{i1} die Anzahl der Testinstanzen pro Testfall an, für welche die Nachoptimierung (*3BPPO*) zu einer Bineinsparung führt.
- In den Spalten "Wirkung von 1BPH" werden einerseits die Rechenzeiten des Teilverfahrens *1BPH* absolut und prozentual bzgl. der Gesamtrechenzeit angegeben. Ferner gibt n_{i2} die Anzahl der Instanzen pro Testfall an, für die *1BPH* zu einer Binreduzierung führt.

Die Ergebnisse des Tests werden nachfolgend unter drei Aspekten analysiert. Erstens wird die Performance der Heuristik *BPHL* beurteilt. Zweitens wird die Wirksamkeit einzelner Komponenten der Heuristik eingeschätzt. Drittens wird der Zusammenhang zwischen gewissen Instanzparametern einerseits und der erreichten Lösungsgüte andererseits betrachtet.

Tab. 3. Testresultate von *BPHL* für 1800 Probleminstanzen.

Dim	Testfall	LO	BPHL						BPHL-			Wirkung von 3BPPO				Wirkung von 1BPH		
			nb (Bins)	gap (%)	vutil (%)	n_opt	n_opt (%)	t_tot (s)	nb (Bins)	gap (%)	vutil (%)	nb (Bins)	gap (%)	vutil (%)	n_i1	t_1BPH (s)	t_1BPH (%)	n_i2
3D	MB01	10	11.0	10.40	86.33	11	7.3	125.5	11.1	10.87	85.97	-0.05	-0.47	0.36	7	0.9	0.7	3
	MB02		10.7	6.67	88.99	50	33.3	99.1	10.7	7.13	88.60	-0.05	-0.47	0.39	7	1.2	1.2	0
	MB03	40	46.9	17.17	84.29	0	0.0	101.2	47.8	19.52	82.63	-0.94	-2.35	1.66	97	2.2	2.2	13
	MB04		44.0	10.10	89.71	0	0.0	97.1	45.1	12.68	87.65	-1.03	-2.58	2.06	137	1.9	1.9	0
	MB05	100	130.4	30.41	76.33	0	0.0	123.2	133.9	33.86	74.37	-3.45	-3.45	1.97	132	3.2	2.6	12
	MB06		122.1	22.11	81.51	0	0.0	122.6	124.4	24.38	80.02	-2.27	-2.27	1.49	147	2.6	2.1	6
Mittel-3D			-	16.14	84.53	10.2	6.8	111.5	-	18.07	83.21	-1.30	-1.93	1.32	87.8	2.0	1.8	5.7
2D	MB07	10	10.2	2.00	93.28	120	80.0	26.7	10.2	2.00	93.28	0.00	0.00	0.00	0	0.1	0.5	2
	MB08		10.2	1.53	93.40	127	84.7	22.5	10.2	1.73	93.21	-0.02	-0.20	0.18	3	0.1	0.6	0
	MB09	40	41.4	3.53	95.38	4	2.7	118.1	41.5	3.87	95.07	-0.13	-0.33	0.31	16	2.3	1.9	21
	MB10		40.9	2.17	96.61	27	18.0	106.5	41.0	2.42	96.38	-0.10	-0.25	0.24	15	4.2	3.9	11
	MB11	100	111.1	11.07	89.57	0	0.0	122.5	111.9	11.87	88.92	-0.80	-0.80	0.64	78	1.4	1.1	23
	MB12		104.7	4.71	95.03	0	0.0	122.9	105.4	5.38	94.43	-0.67	-0.67	0.60	83	1.2	1.0	10
Mittel-2D			-	4.17	93.88	46.3	30.9	86.5	-	4.54	93.55	-0.29	-0.37	0.33	32.5	1.5	1.5	11.2
Mittel			-	10.16	89.20	28.3	18.8	99.0	-	11.31	88.38	-0.79	-1.15	0.82	60.2	1.8	1.7	8.4

Ein direkter Vergleich von *BPHL* mit anderen Bin Packing-Verfahren ist derzeit nicht möglich, weil für die neuen Instanzen noch keine Vergleichswerte vorliegen. Auch ein Test von *BPHL* anhand bekannter Benchmarkinstanzen für das 2D-BPP oder 3D-BPP scheitert aus, weil diese eine Orientierungsrestriktion aufweisen, welche *BPHL* aufgrund des integrierten CLP-Verfahrens nicht berücksichtigen kann. Jedoch kann die mit *BPHL* erreichte Lösungsgüte indirekt bewertet werden.

Die von *BPHL* im 3D-Fall erreichte mittlere Auslastung von 84.5% liegt nicht allzu weit entfernt von Auslastungswerten, die für das CLP mit einem Container von aktuellen Heuristiken erreicht werden. So erzielte das Baumsuche-Verfahren von Eley (2002) für die ersten sieben Testfälle von Bischoff und Ratcliff eine mittlere Auslastung von 88.8% und das GRASP-Verfahren von Moura und Oliveira (2005) erreichte für alle 15 Testfälle von Bischoff und Ratcliff eine mittlere Auslastung von 86.7%. Im 2D-Fall gilt eine analoge Aussage.

Für einen weiteren indirekten Vergleich der Lösungsgüte sei die L_0 -Abweichung (*gap*) betrachtet. *BPHL* erreicht für die 3D-Instanzen eine mittlere L_0 -Abweichung von 16.1% und für die 2D-Instanzen eine mittlere L_0 -Abweichung von 4.2%. Faroe et al. (2003) verwenden zur Auswertung der Ergebnisse ihrer leistungsfähigen Metaheuristik den lower bound L_2 , der stärker als L_0 ist. Für die berechneten 3D-Instanzen wird eine mittlere L_2 -Abweichung von 6.8% und für die beiden 2D-Testsets eine mittlere L_2 -Abweichung von 7.2% bzw. 5.5% erzielt. Nun besitzen die von Faroe et al. berechneten Instanzen anders als die hier getesteten Instanzen eine Orientierungsrestriktion. Dennoch spricht auch dieser Ergebnisvergleich dafür, dass *BPHL* in der Regel eine befriedigende Lösungsqualität erreicht. Gute Ergebnisse liegen insbesondere für die 2D-Testfälle mit den L_0 -Werten 10 und 40 sowie für die 3D-Testfälle mit $L_0 = 10$ vor. Bei den letztgenannten 3D-Testfällen (MB01, MB02) wäre eine mittlere Binauslastung von 95.1% über die zugehörigen 300 Instanzen erforderlich, um den L_0 -Wert bei jeder Instanz zu realisieren! Bei dieser Sachlage ist eine mittlere L_0 -Abweichung von ca. 8.5% für die beiden Testfälle MB01 und MB02, wonach der lower bound L_0 im Mittel um weniger als ein Bin verfehlt wird, offensichtlich ein guter Wert. Auch die erreichten Prozentsätze optimaler Lösungen (n_{opt}) mit Bezug auf L_0 sprechen für eine teils gute bis sehr gute Lösungsqualität.

Auch zur Evaluierung der verbrauchten Rechenzeit sei ein Vergleich mit dem Verfahren von Faroe et al. angestellt. Diese Metaheuristik wurde auf einer Workstation mit einer Taktfrequenz von 500 MHz getestet. Als Zeitschranke wurden 1000 Sekunden für 3D-Instanzen bzw. 100 Sekunden für 2D-Instanzen verwendet (das Zeitlimit wurde stets ausgeschöpft). Nun wurde zwar *BPHL* auf einem schnelleren Prozessor (1.86 GHz) getestet. Jedoch muss beachtet werden, dass die berechneten Instanzgrößen stark differieren. Faroe et al. berechneten 3D-Instanzen mit 50 bis maximal 200 Kisten und 2D-Instanzen mit 20 bis 100 Rechtecken. Dagegen werden hier Instanzen mit durchschnittlich mehr als 1500 Packstücken berechnet (vgl. 4.1). Die mittlere von *BPHL* konsumierte Rechenzeit von ca. 100 Sekunden ist daher als recht gering einzustufen. Zusammenfassend kann eingeschätzt werden, dass *BPHL* offenbar in der Lage ist, für extrem große zwei- und dreidimensionale Bin Packing-Instanzen in knapp bemessener Rechenzeit Lösungen mit befriedigender Qualität zu ermitteln.

Betrachtet sei nun die Wirksamkeit einzelner Verfahrenskomponenten. Der Modul *IBPH* zur Lösung des eindimensionalen BPP zielt auf eine Bineinsparung durch Umverteilung von Schichten ab. In 101 der 1800 Instanzen (5.6% der Instanzen) ist diese Maßnahme mindestens einmal im Suchverlauf erfolgreich. Der Rechenaufwand ist sehr gering und beträgt im Mittel 1.8 s pro Instanz; diese Zeit entspricht einem durchschnittlichen Anteil von 1.8% der mittleren Rechenzeit pro Instanz.

Die Nachoptimierungsprozedur *3BPPO* bewirkt eine mittlere Reduzierung der L_0 -Abweichung um 1.15 %-Punkte und eine mittlere Steigerung der Volumenauslastung um 0.82 %-Punkte bezogen auf alle 1800 Instanzen. Für etwa 40% der 1800 Instanzen führt *3BPPO* zu einer

Ergebnisverbesserung. Weil für die Heuristik ohne Nachoptimierung (*BPHL*–) sowie für die Nachoptimierung (*3BPPO*) im Test jeweils 60 maximal Sekunden Rechenzeit verfügbar waren ($t_1 = 60s$, $t_2 = 60s$), werden keine näheren Angaben zum Zeitbedarf von *BPHL*– und *3BPPO* gemacht. Durch einen Verzicht auf die Nachoptimierung kann bei der gegebenen Parametrisierung die Rechenzeit um fast die Hälfte gesenkt werden, wobei eine nur mäßig schlechtere Lösungsqualität die Folge ist (vgl. Tab. 3). Dies unterstreicht nochmals die Fähigkeit der Heuristik *BPHL*, in kurzer Zeit extrem große Instanzen befriedigend zu lösen.

Abb. 7 verdeutlicht anhand der Instanz MB06_015 den Suchverlauf von *BPHL*. Bereits nach wenigen Sekunden ist eine Ausgangslösung mit 133 Bins ermittelt, die sogleich mit einer Umverteilung der Schichten auf 132 verbessert wird (Modul *IBPH* ist erfolgreich). In den ersten 60 Sekunden der Suche werden dann die Suchparameter (*cut2*, *cut3*) sukzessive gesteigert, und eine Lösung mit 128 Bins wird gefunden. Anschließend werden im Zuge der Nachoptimierung mittels *3BPPO* immer mehr Schichten der bisherigen Bestlösung aufgelöst und durch Schichten in alternativer Fortschrittsrichtung ersetzt, wodurch eine weitere Binreduzierung auf schließlich 122 Bins erfolgt.

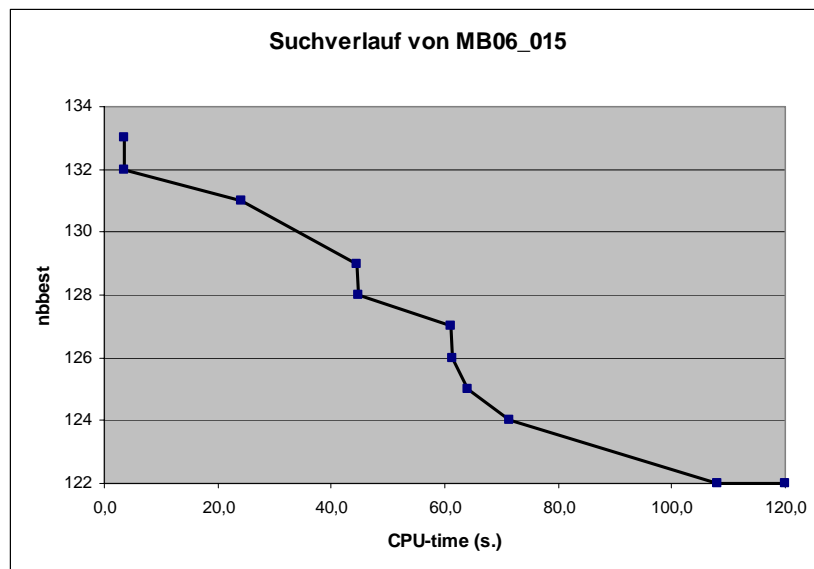


Abb. 7. Suchverlauf von *BPHL* am Beispiel der Instanz MB06_015.

Abschließend sei der Frage möglicher Zusammenhänge zwischen den Parametern der Testinstanzen einerseits und der erreichten Lösungsgüte andererseits nachgegangen.

Für 2D-Instanzen werden im Vergleich zu 3D-Instanzen im Mittel deutlich höhere Auslastungen und zugleich geringere Abweichungen von dem lower bound LO erreicht. Dies korrespondiert mit der Tatsache, dass in der Literatur für das 2D-Knapsack-Problem bzw. 2D-Strip-Packing-Problem allgemein höhere Füllraten als für die analogen 3D-Probleme berichtet werden; so erreichen gute Verfahren im 2D-Fall Füllraten von 98% oder mehr, während derart hohe Auslastungen im 3D-Fall selten vorkommen. Generell erscheint es plausibel, dass das Erreichen einer gewissen Füllrate in einem Behälter mit wachsender Anzahl von Dimensionen schwieriger wird, weil die Vermeidung von Lücken in größerem Umfang die simultane näherungsweise Einhaltung gewisser Maßbeziehungen (z.B. Längenmass der Kiste 1 + Längenmaß der Kiste 2 = Containerlänge) erfordert.

Ein recht klarer Zusammenhang lässt sich zwischen den Werten des lower bound LO einerseits und den Auslastungen bzw. LO -Abweichungen andererseits beobachten. Im Allgemeinen gilt, dass mit wachsendem LO -Wert die erreichten Auslastungen sinken und zugleich die LO -Abweichungen zunehmen; eine Ausnahme bilden lediglich die 2D-Testfälle MB09 und MB10. Auch dieser Trend

erscheint plausibel, weil mit wachsenden *LO*-Werten die Größe der Packstücke relativ zu den Containermaßen wächst und die Packstücke "sperriger" werden. Die Module *IBPH* (Schichtumverteilung) bzw. *3BPPO* (Nachoptimierung) erweisen sich für die Instanzen mit höheren *LO*-Werten (40, 100) bzw. mit relativ großen Packstücken als besonders erfolgreich (s. Tab. 3).

Schließlich wird für die Testfälle, deren Kistenmengen gemäß Modus 2 erzeugt wurden, generell eine bessere Lösungsqualität erreicht als für die korrespondierenden Testfälle des Modus 1 (vgl. 4.1). Die nach Modus 2 erzeugten Kistenmengen besitzen eine im Allgemeinen viel höhere Heterogenität. Weil bei dem 3D-CLP in der Vergangenheit umgekehrt für Testfälle mit Kistenmengen von geringer Heterogenität besonders hohe Auslastungen erreicht wurden, wirkt der hier festgestellte Trend zunächst etwas überraschend. Nun kann sich die Heterogenität einer Kistenmenge zumindest in zweierlei Hinsicht auf die erreichbare Auslastung auswirken. Einerseits ermöglicht die Existenz einer gewissen Anzahl von Kisten pro Kistentyp die Bildung zusammenhängender Teilanordnungen (Blöcke) aus Kisten gleichen Typs und in gleicher räumlicher Orientierung, die keine Innenverluste aufweisen. Insofern kann sich eine relativ geringe Heterogenität positiv auf eine hohe Gesamtauslastung auswirken. Andererseits wächst mit zunehmender Heterogenität der Kistenmenge der Reichtum an Kombinationsvarianten für Maße, speziell für die Containerabmessungen, was ebenfalls eine hohe Gesamtauslastung begünstigen kann. Eine mögliche Erklärung für den beobachteten Trend könnte nun darin bestehen, dass bei dem hohen mittleren Niveau der Heterogenität der hier eingeführten Instanzen der zuletzt genannte Einfluss überwiegt. Auch bei der großen Mehrzahl der nach Modus 1 erzeugten Instanzen sind meist zu wenige Exemplare pro Kistentyp vorhanden, um größere Blöcke bilden zu können; daher erweist sich vermutlich der Vorteil der nach Modus 2 erzeugten Instanzen bezüglich der Kombinierbarkeit von Maßen als ausschlaggebend. Jedoch muss diese Erklärung im Lichte weiterer Testergebnisse für die eingeführten Instanzen überprüft werden.

5 Zusammenfassung

Der Beitrag stellt eine Heuristik für das mehrdimensionale Bin Packing Problem (BPP) mit einem Containertyp und rechteckigen Packstücken vor. Die Heuristik basiert auf einem Verfahren für das eindimensionale BPP sowie einem Verfahren für das 3D-Container Loading Problem (CLP), welches dem schichtbildenden Ansatz folgt. Als 3D-CLP-Verfahren wird hier das Baumsuche-Verfahren von Pisinger (2002) gewählt. Ihrem Typ nach ist die Heuristik ein auf recht einfachen Verfahrensideen basierendes Verbesserungsverfahren. Um ihre Eignung für sehr große Probleminstanzen nachzuweisen, wurden 1800 neue Instanzen mit durchschnittlich mehr als 1500 Stücken eingeführt. Neben der Dimensionalität werden auch die relative Packstückgröße sowie die Heterogenität der Packstückmenge systematisch variiert. Der Test ergab, dass die Heuristik in der Lage ist, für die extrem großen Instanzen in kurzen Rechenzeiten von ein bis zwei Minuten Lösungen zu berechnen, die zumindest in der Regel eine befriedigende Volumenauslastung aufweisen. Nach dem besten Wissen der Autoren wird erstmals eine Heuristik für das mehrdimensionale BPP anhand von Instanzen der angegebenen Größenordnung getestet und bewertet. Empirisch untersucht und analysiert wurde auch die Abhängigkeit der erzielten Lösungsqualität von der mittleren relativen Packstückgröße und der Heterogenität der Kistenmenge einer Instanz. Die Heuristik beachtet bisher nur die Guillotineschnitt-Bedingung. Weitere Restriktionen, die etwa die Orientierung der Packstücke sowie die Stabilität der Beladung betreffen, können durch die Wahl eines alternativen 3D-CLP-Verfahrens (mit schichtbildendem Ansatz) relativ einfach berücksichtigt werden. Dies bleibt ein Gegenstand weiterer Forschungstätigkeit.

Danksagung: Die Autoren bedanken sich bei Herrn Jens Hochmuth für seine wertvolle Mithilfe bei der technischen Umsetzung des Verfahrens.

Literaturverzeichnis

- BENGTSOON, B.-E. (1982). Packing Rectangular Pieces – A Heuristic Approach. *The Computer Journal*, Vol. 25, pp. 353–357.
- BERKEY, J.O.; WANG, P.Y. (1987). Two dimensional finite bin packing algorithms. *Journal of the Operational Research Society*, Vol. 38, pp. 423–429.
- BISCHOFF, E. E.; RATCLIFF, M. S. W. (1995). Issues in the development of approaches to container loading. *Omega*, Vol. 23, pp. 377–390.
- BORTFELDT, A. (2000). Eine Heuristik für Multiple Containerladeprobleme. *OR Spectrum*, Vol. 22, pp. 239–261.
- BORTFELDT, A. (2005). A hybrid procedure for the one-dimensional bin packing problem. Presentation on the 2th ESICUP-Meeting, Southampton 2005. Siehe <http://paginas.fe.up.pt/~esicup/>.
- BORTFELDT, A. (2007). Verfahren für das eindimensionale Bin Packing Problem. Vortrag in der Forschungswerkstatt der Fakultät Wirtschaftswiss. der FernUniversität in Hagen, März 2007. S. www.fernuni-hagen.de/WINF.
- COFFMAN JR., E.G.; CSIRIK, J.; JOHNSON, D.S.; WOEGINGER, G.J. (2004). An Introduction to Bin Packing. Bibliographie. Siehe www.inf.u-szeged.hu/~csirik.
- DAVIES, A.P.; BISCHOFF, E.E. (1998). Weight Distribution Considerations in Container Loading. European Business Management School, University of Wales, Swansea, Technical Report.
- DE CASTRO SILVA, J.L.; SOMA, N.Y.; MACULAN, N. (2003). A greedy search for the tree-dimensional bin packing problem: the packing stability case. *Internat. Transactions in Operational Research*, Vol. 10, pp. 141–153.
- DELL'AMICO, M.; MARTELLO, S.; VIGO, D. (2002). A lower bound for the non-oriented two-dimensional bin packing problem. *Discrete Applied Mathematics*, Vol. 118, pp. 13–24.
- EL BOURI, A.; POPPLEWELL, N.; BALAKRISHNAN, S.; ALFA, A. (1994). A search based heuristic for the two-dimensional bin-packing problem. *INFOR*, Vol. 32, pp. 265–274.
- ELEY, M. (2002). Solving Container Loading Problems by Block Arrangements. *European Journal of Operational Research*, Vol. 141, pp. 393–409.
- ELEY, M. (2003). A bottleneck assignment approach to the multiple container loading problem. *OR Spectrum*, Vol. 25, pp. 45–60.
- FAROE, O.; PISINGER, D.; ZACHARIASEN, M. (2003). Guided local Search for the Three-Dimensional Bin-Packing Problem. *INFORMS Journal on Computing*, Vol. 15, pp. 267–283.
- GLOVER, F.; KOCHENBERGER, G. (2003)(Eds.): *Handbook of Metaheuristics*. Kluwer Academic Publishers, Dordrecht.
- IVANCIC, N.; MATHUR, K.; MOHANTY, B. B. (1989). An Integer-Programming Based Heuristic Approach to the Three-Dimensional Packing Problem. *Journal Manufacturing and Operations Management*, Vol. 2, pp. 268–298.
- LODI, A.; MARTELLO, S.; VIGO, D. (1998). Neighborhood search algorithm for the guillotine non-oriented two-dimensional bin packing problem. In: S. Voss, S. Martello, I.H. Osman, C. Roucairol (Eds.), *Meta-Heuristics: Advances and Trends in Local Search Paradigms for Optimization*, Kluwer Academic Publishers, Boston, pp. 125–139.
- LODI, A.; MARTELLO, S.; VIGO, D. (1999a). Heuristic and metaheuristic approaches for a class of two-dimensional bin packing problems. *INFORMS Journal on Computing*, Vol. 11, pp. 345–357.
- LODI, A.; MARTELLO, S.; VIGO, D. (1999b). Approximation algorithms for the oriented two-dimensional bin packing problem. *European Journal of Operational Research*, Vol. 112, pp. 158–166.
- MARTELLO, S.; PISINGER, D.; VIGO, D. (2000). The three-dimensional bin packing problem. *Operations Research*, Vol. 48, pp. 256–267.
- MARTELLO, S.; VIGO, D. (1998). Exact solution of the two-dimensional finite bin packing problem. *Management Science*, Vol. 44, pp. 388–399.
- MOURA, A.; OLIVEIRA, J. F. (2005). A GRASP approach to the Container-Loading Problem. *IEEE Intelligent Systems* Vol. 20, pp. 50–57.
- PISINGER, D. (2002). Heuristics for the Container Loading Problem. *European Journal of Operational Research*, Vol. 141, pp. 143–153.
- PISINGER, D.; SIGURD, M. (2007). Using Decomposition Techniques and Constraint Programming for Solving the Two-Dimensional Bin-Packing Problem. *INFORMS Journal on Computing*, Vol. 19, pp. 36–51.
- TERNO, J.; SCHEITHAUER, G.; SOMMERWEIß, U.; RIEHME, J. (2000). An efficient approach for the multi-pallet loading problem. *European Journal of Operational Research*, Vol. 123, pp. 372–381.
- WÄSCHER, G.; HAUBNER, H.; SCHUMANN, H. (2007). An improved typology of cutting and packing problems. *European Journal of Operational Research*, Vol. 183, pp. 1109–1130.