Parallel greedy algorithms for packing unequal spheres into a cuboidal strip or a cuboid

Timo Kubach, Andreas Bortfeldt,
Thomas Tilli and Hermann Gehring

Diskussionsbeitrag Nr. 440

August 2009

Diskussionsbeiträge der Fakultät für Wirtschaftswissenschaft
der FernUniversität in Hagen

Herausgegeben vom Dekan der Fakultät

Parallel greedy algorithms for packing unequal spheres into a cuboidal strip or a cuboid

Timo Kubach, Andreas Bortfeldt, Thomas Tilli and Hermann Gehring

Abstract:
Given a finite set of spheres of different sizes we study the three-dimensional Strip Packing Problem (3D-SPP) as well as the three-dimensional Knapsack Problem (3D-KP). The 3D-SPP asks for a placement of all spheres within a cuboidal strip of fixed width and height so that the variable length of the cuboidal strip is minimized. The 3D-KP requires packing of a subset of the spheres in a given cuboid so that the wasted space is minimized. To solve these problems some greedy algorithms were developed which adapt the algorithms proposed by Huang et al. (2005) to the 3D case with some important enhancements. Furthermore, the new greedy algorithms were parallelized using a master slave approach. The resulting parallel methods were tested using the instances introduced by Stoyan et al. (2003). Additionally, two sets of 288 instances each for the 3D-SPP and for the 3D-KP were generated and results for these new instances are also reported.

Keywords:
Packing, spheres, strip packing problem, knapsack problem, greedy method, parallelization.

Fakultät für Wirtschaftswissenschaft, FernUniversität in Hagen
Profilstr. 8, 58084 Hagen, Deutschland
Tel.: +49 (0)2331 987-4431
Fax: +49 (0)2331 987-4447
E-Mail: andreas.bortfeldt@fernuni-hagen.de

Parallel greedy algorithms for packing unequal spheres into a cuboidal strip or a cuboid

Timo Kubach, Andreas Bortfeldt, Thomas Tilli and Hermann Gehring

1 Introduction

This paper deals with the three-dimensional (3D) Strip Packing Problem (SPP) and the 3D Knapsack Problem (KP) where unequal spheres are the small objects to be packed. Given a finite set of spheres, the SPP asks for a non-overlapping placement of all spheres within a cuboidal strip of fixed width and height so that the variable length of the strip is minimized. The KP requires packing a subset of a given set of spheres in a cuboid of fixed dimensions without overlap so that the wasted space is minimized. The 2D-SPP and the 2D-KP each with unequal circles are analogously defined. All problems described are known to be NP-hard (cf. Lenstra and Rinnooy Kan, 1979).

Up to the present, only a few solution methods are known that address 3D packing problems with differently sized spheres. Visscher and Bolsterli (1972) consider a problem related to the 3D-SPP. By means of a Monte Carlo method, their algorithm simulates the falling of spheres in a cuboid following gravitation. Wang (1999) deals with a special 3D packing problem for the planning of radiosurgical treatment using a dynamic programming algorithm. Sutou and Dai (2002) address the generalized 3D-KP of packing spheres into a 3D polyhedron. Their algorithms are based on mathematical optimization methods (e.g., LP relaxation). The maximum instance size is five spheres. Stoyan et al. (2003) propose a mathematical optimization method for the 3D-SPP with unequal spheres. The calculated instances range between 25 and 60 spheres.

For the 2D-SPP and the 2D-KP each with unequal circles meta-heuristics (cf. Glover and Kochenberger 2003) as well as analytical methods were proposed. George et al. (1995) put forward a genetic algorithm (GA) for a problem from industrial practice that is related to the 2D-KP. Hifi and M'Hallah (2004) present a construction heuristic and a GA for solving the cutting problem dual to the 2D-KP while Hifi et al. (2004) propose a Simulated Annealing algorithm for the same problem. An analytical method for the 2D-SPP goes back to Stoyan and Yaskov (2004).

Huang et al. (2005) are concerned with the two-dimensional decision problem related to the circular 2D-KP: if a rectangular container of fixed dimensions and a set of generally unequal circles is given, is there a feasible packing plan placing all circles in the container? They propose two greedy algorithms that reached promising results.

Thus, it is the intention of the paper at hand to adapt the greedy methods of Huang et al. to the 3D case in such a way that both 3D optimization problems defined above are directly addressed. Moreover, the methods of Huang et al. should be enhanced and merged in order to create a single and effective greedy method for the SPP and the KP with unequal spheres, respectively. To improve their efficiency, the new algorithms are then parallelized. A comparison to the 3D-SPP method from Stoyan et al. (2003) is carried out. In order to stimulate more detailed comparisons of different methods dealing with the problems studied here, two sets of 288 instances each for the SPP and for the KP are additionally generated and first results gained for these instances are reported and analyzed.

The remainder of the paper is organized as follows. In Section 2, the greedy methods of Huang et al. are summarized and their main concepts are adapted to the 3D case. Then modified methods for the 3D-KP and the 3D-SPP, respectively, are described. The new benchmark instances for both problems are defined in Section 3. In Section 4, the experimental results are presented. Finally, the paper is summarized in Section 5.

2 Greedy algorithms for the KP and the SPP with unequal spheres

2.1 The methods B1.0 and B1.5 proposed by Huang et al.

The methods of Huang et al. (2005) for the 2D decision problem defined above are based on three important concepts: corner placement, hole degree and maximum hole degree (MHD) rule. A placement $p = (i, x_i, y_i)$ of a given circle i with the midpoint coordinates x_i and y_i is regarded as feasible if the circle lies completely within the container and does not overlap with any other circle. A feasible placement is called a *corner placement* if the circle touches (at least) two items, i.e., another circle or a side of the rectangle.

Let P be a feasible packing plan (consisting of feasible placements exclusively) and $p = (i, x_i, y_i)$ a corner placement belonging to P. Moreover, let u and v be the two items (circle or side) touching circle i. The *hole degree* λ of the corner placement p is defined as

$$\lambda(p) = \lambda((i, x_i, y_i)) = 1 - \frac{d_{\min}}{r_i}. \tag{1}$$

In equation (1) r_i is the radius of circle i while $d_{\min}$ is the minimal distance from circle i to other circles of P and to the rectangle sides with the exception of items u and v. The distance between two circles ($d_{i,j}$) and the distance between a circle and a rectangle side ($d_{i,sj}$) are illustrated in figure 1.

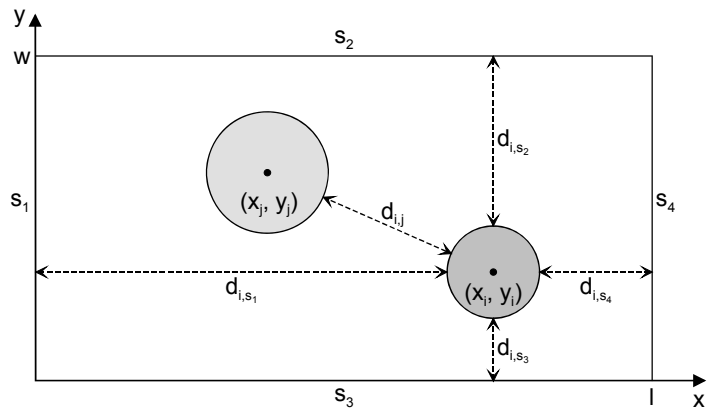

Figure 1. Distances between circles and between a circle and a rectangle side.

The hole degree of a corner placement indicates how close to other circles of a plan (and to the rectangle sides) a given circle is accommodated. The higher the mean hole degree of the placements of a packing plan the higher the density of the plan. Therefore, the *maximum hole degree rule* says that given a set of possible (additional) corner placements, the placement with the maximum hole degree should be selected as the next one.

The first greedy algorithm – called B1.0 – consists of a core procedure and a frame procedure:

- The core procedure (B1.0C) takes an incomplete packing plan *P*' as input and provides a complete packing plan *P*''. For this further corner placements are implemented one by one according to the MHD rule. A plan is complete if either all given circles are placed (success) or no further corner placements are available for the remaining circles (failure). A plan including all circles is also called a successful plan.
- The frame procedure generates step by step so-called initial plans. An initial plan consists of two circle placements. The first circle is placed at the bottom left corner of the rectangle, and the second circle touches the first one and a side of the rectangle or it touches two rectangle sides. For each possible initial plan, the core procedure B1.0C is called once. If B1.0C returns with success, the frame procedure (i.e. the algorithm) stops immediately. Otherwise, the algorithm ends if there is no further untried initial plan.

The second greedy algorithm – denoted by B1.5 – has a similar structure to B1.0 and the frame procedure of B1.0 is transferred to B1.5. Again, the core procedure (B1.5C) transforms a packing plan *P*' into a complete packing plan *P*'' by corner placements carried out one after another. The next corner placement p^* for an interim plan P_1, however, is now determined as follows:

- Each possible corner placement p is tentatively implemented and the extended plan $P_1 \cup \{p\}$ is then completed by means of the core procedure B1.0C. This results in a complete packing plan $P_2(p)$ with a density $d_P(P_2)$.
- At the end, the corner placement p^* is selected and finally implemented for which the density $d_P(P_2)$ of the associated complete plan $P_2(p^*)$ is maximal.

Obviously, the core procedure B1.5C employs a forward-looking strategy to make the selection of corner placements more sophisticated. While in algorithm B1.0 the decision about the next corner placement only depends on placements implemented earlier, in algorithm B1.5, the circles not yet packed are also taken into account.

Huang et al. (2005) have shown that the B1.0's time complexity is $O(n^6)$ and B1.5's is $O(n^{10})$ (n stands for the number of circles).

2.2 Adaptation of basic concepts to the 3D case

The three basic concepts explained above can easily be adapted to the 3D case:

- A feasible placement $p = (i, x_i, y_i, z_i)$ of a sphere i with midpoint coordinates x_i, y_i, z_i is called a corner placement if the sphere touches (at least) three objects (placed spheres, sides of cuboid).
- The hole degree $\lambda(p)$ of a corner placement $p = (i, x_i, y_i, z_i)$ is calculated as follows. Suppose sphere i with radius r_i touches the objects j, k and l. Let further d_{min} denote the minimal Euclidean distance from sphere i to all other objects of a feasible packing plan (including cuboid sides) except objects j, k and l. Then the hole degree of corner placement p is again given by $\lambda(p) = 1 - d_{min}/r_i$.
- The MHD rule requires that from a set of possible placements of spheres the placement with the maximum hole degree is always selected as the next placement.

2.3 Algorithm B1.6_3DKP for the 3D Knapsack Problem

As the algorithm developed for the 3D-KP is an advancement and a 3D adaptation of algorithm B1.5, it is called B1.6_3DKP. The frame procedure of algorithm B1.6_3DKP is shown in figure 2.

The frame procedure generates step by step different initial plans. Every initial plan is then extended to a complete packing plan by means of the core procedure B1.6_3DKP_C and the best packing plan in terms of density is updated whenever necessary and returned at last.

```
Procedure B1.6_3DKP(in: instance data I, parameter τ, out: packing plan P_best)
P_best := ø;
for (every initial plan P_init) do
        P_res := B1.6_3DKP_C(I, τ, P_init);
        if (d_p(P_res) > d_p(P_best)) then
                P_best := P_res;
                // stop immediately if a global optimal solution is reached
                if (P_best includes all given spheres) then return P_best; endif;
        endif;
endfor;
return P_best;
end.
```

Figure 2. Frame procedure of algorithm B1.6_3DKP.

An initial plan consists of two sphere placements. Seven types of initial plans are differentiated. It is assumed that the container is embedded into a 3D coordinates system (3D-CS) as shown in figure 3. The larger sphere of an initial plan is always placed in the origin of the 3D-CS. A type A initial plan is given if the smaller sphere is situated in the corner vis-à-vis to the origin while in initial plans of types B to G the smaller sphere lies in one of the other six corners of the cuboid as illustrated in figure 3. Note that this mode of construction does not rule out that a second sphere is placed in a corner that is already occupied since such a placement can be implemented as the third, fourth etc. one.

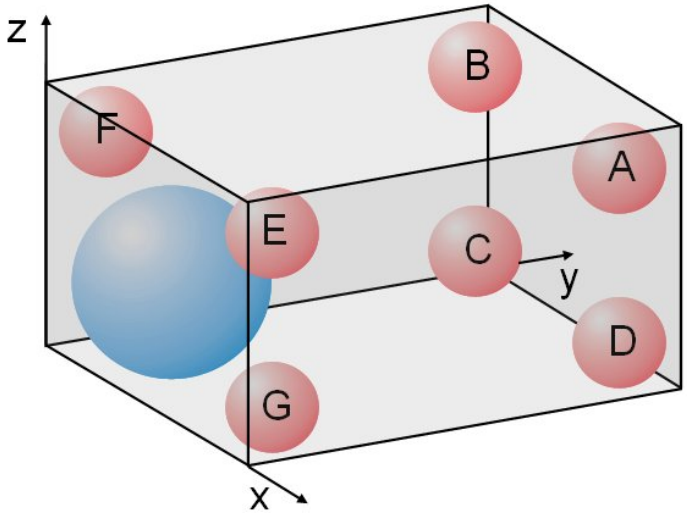

Figure 3. Initial plans of types A to G.

To limit the search effort, only a subset of possible initial plans is investigated. Let n_{dif} be the number of sphere types of a given instance (where a sphere type is given by a radius) and let the sphere types be sorted by the radius in descending order. For each pair (i, j) of sphere types ($1 \leq i \leq j \leq n_{dif}$), at least the initial plan of type A is probed. If

there are no more than 1000 initial plans of type A, then all possible initial plans of the types B to G are also tried.

Although the frame procedure just described is similar to those of the methods B1.0 and B1.5, there are some important differences: Having been adapted to the Knapsack Problem, method B1.6_3DKP searches for packing plans of maximum density instead of focusing only on plans including all given spheres. Further, B1.6_3DKP is able to deal efficiently with strictly heterogeneous instances (where any two spheres are unequal) as well as with non-strictly heterogeneous instances, too. In the latter case, redundant calculations are consistently avoided only by B1.6_3DKP and this applies, in particular, to the generation of initial plans. Finally, the generation of initial plans is governed by different rules compared to B1.0 and B1.5.

In figure 4 the core procedure B1.6_3DKP_C of algorithm B1.6_3DKP is listed.

```
Procedure B1.6_3DKP_C(in: instance data I, parameter τ, inout: packing plan P)
determine list L of possible corner placements p = (i, x_i, y_i, z_i)
regarding (incomplete) packing plan P and calculate the hole degrees λ(p);
while (there are corner placements in L) do
    select placement p* with the maximum hole degree λ_max from L;
    if (λ_max > τ) then
        P := P ∪ {p*}; // implement placement p*
        update list L;
    else
        best_density := 0;
        for (each corner placement p in L) do
            let P' be a copy of P and L' be a copy of L;
            P' := P' ∪ {p}; // implement placement p tentatively
            update list L';
            P'' := B1.6_3DKP_C2(I, L', P'); // complete plan P'
            if (P'' includes all given spheres) then P := P''; return P; endif;
            if (d_p(P'') > best_density or
               d_p(P'') = best_density and λ(p) > λ(p*)) then
                    p* = p; best_density := d_p(P'');
            endif;
        endfor;
        P := P ∪ {p*}; // implement placement p* finally
        update list L;
    endif;
endwhile;
return P;
end.
```

Figure 4. Core procedure B1.6_3DKP_C.

A main feature of the core procedure is the control mechanism introduced with the continuous threshold parameter τ. Let p^* be a possible corner placement with maximum hole degree λ_{max} at a given time, i.e., for a certain cycle of the while-loop. If λ_{max} exceeds the value of the threshold parameter, the sphere corresponding to p^* is packed as in procedure B1.0C. Otherwise, the forward-looking strategy of procedure B1.5C is used to decide which sphere will be placed next. Hence, the core procedure B1.6_3DKP_C combines the core procedures of the algorithms B1.0 and B1.5. This combination shows two aspects:

- From a formal point of view two algorithms are replaced and generalized by one. If threshold τ is set to 1, procedure B1.6_3DKP_C behaves exactly like B1.5C; if τ is set to a sufficiently small value B1.6_3DKP_C proceeds as B1.0C.
- More importantly, the threshold parameter allows the trade-off between solution quality and runtime effort to be controlled. The higher the value of τ the higher the packing density that may be expected and the lower τ the faster the search will be finished.

The forward-looking strategy adopted from procedure B1.5C was enhanced by a second evaluation criterion for placements: The hole degree is taken as a tie breaker if two completed packing plans have the same density. Procedure B1.6_3DKP_C2 is called repetitively by B1.6_3DKP_C and it coincides with procedure B1.0C already explained, cf. figure 5.

```
Procedure B1.6_3DKP_C2 (in: instance data I, list L of corner placements, inout: packing plan P)

while (there are corner placements in L) do
        select placement p* with the maximum hole degree λmax from L;
        P := P ∪ {p*}; // implement placement p*
        update list L;
endwhile;
return P;
end.
```

Figure 5. Core procedure B1.6_3DKP_C2.

The worst case time complexity of the core procedure B1.6_3DKP_C can be determined analogously as for B1.5C and it is of the order $O(n^{18})$. As the number of initial plans is bounded by $O(n^2)$ and B1.6_3DKP_C is called once per plan, the time complexity of B1.6_3DKP is $O(n^{20})$.

2.4 Algorithm B1.6_3DSPP for the Strip Packing Problem

The algorithm proposed for the 3D-SPP is termed B1.6_3DSPP. It consists of a frame procedure and two core procedures that are similar to those of algorithm B1.6_3DKP. Therefore, only the main differences between the corresponding procedures are outlined in what follows.

Again, the frame procedure of B1.6_3DSPP generates initial plans and a core procedure called B1.6_3DSPP_C tries to extend each plan to a complete packing plan. Now a complete plan should include all given spheres and the search is for a complete plan of minimum length. Hence, initialization and update of the best plan are changed accordingly and the (current) best plan is stored together with its length l_{best}. As in the KP method, only a subset of initial plans is investigated and an analogous rule is applied to select initial plans. However, alternative types of initial plans are tried to some extent in which the placed spheres mostly touch one another.

In the core procedure B1.6_3DSPP_C, a list *L* of possible corner placements is supplied first, as in B1.6_3DKP_C, but as there is no length given for the SPP, no cuboid side *s4* (cf. figure 1) can be considered either as a possibly touched item of a corner placement or in the computation of hole degree values. Moreover, no corner placement touching or overlapping the line $x = l_{best}$ is accepted for list *L*, since after such a placement the current best solution cannot be improved anymore. The features explained above not only affect the generation of list *L* but also its later updates. Finally, the placements of the last two spheres of each generated packing plan are determined by full enumeration, as the computational effort is negligible.

It is easy to see that B1.6_3DSPP doesn't differ from B1.6_3DKP in terms of worst case time complexity. So B1.6_3DSPP's time complexity is also $O(n^{20})$.

2.5 Parallelization of the algorithms

B1.6_3DKP and B1.6_3DSPP sequentially check in most cases a large number of initial plans. Initial plans are explored independently of each other. Hence, it is possible to check them in parallel, leading to a runtime reduction while the solution quality is not affected. The parallelization applied here can be described in detail as follows:

- A shared memory master-slave approach was chosen where the communication between processes is kept to a minimum and only a little control effort accounts for the master process. Therefore, the master is also involved in checking initial plans.
- The shared variables allocated by the master process are primarily used to store the input data (filename of instance, threshold parameter) and the (current) best density

(KP) or best length (SPP). The communication between the processes is done asynchronously, as each process decides on its own at which moment shared variables are read or modified.

- The master process as well as the n_{SP} slave processes check initial plans; in the tests performed here on dualcore PCs only one slave process exists besides the master process (n_{SP} = 1). Additional tests were carried out in which three slave processes are run (n_{SP} = 3) besides the master process. To avoid multiple examinations of initial plans, the processes mark checked placements by increasing a shared counter variable that is initialized by the master and indicates the next initial plan to be checked. A new best plan is immediately stored on hard disk, overwriting the old best plan and the shared variable, and the best objective function value is also updated.
- When all initial plans have been checked, the master process simultaneously terminates all processes. As it is possible to find a successful plan when solving the 3DKP, each process of B1.6_3DKP has the possibility of preparing all other processes for termination in case of success, while the termination signal is always sent by the master process. This feature is not implemented in B1.6_3DSPP, as there is no comparable stopping criterion.

The algorithms proposed are construction methods by their very nature instead of local search algorithms. Hence, it is not surprising that the parallelism applied here does not fall into one of the three categories introduced by Crainic and Toulouse (2003) for parallel meta-heuristics. However, the algorithms perform a degenerated tree search (with more than one successor only at the first two tree levels). Correspondingly, the parallelization follows the subtree-distribution model, a well-known approach for parallelizing branch-and-bound methods (cf. Crainic and Toulouse, 2003).

3 New benchmark instances

New benchmark instances for the 3D-SPP and 3D-KP with unequal spheres are introduced in a similar fashion as proposed in Bortfeldt and Gehring (2006) for the 2D-SPP with rectangular items. To generate new instances for the SPP (referred to as KBG_3DSPP), in a first step, five factors (or instance parameters) were identified that probably affect the solution quality and runtime of corresponding solution methods. These factors are: total number of spheres n, number of sphere types n_{dif}, radius of smallest sphere r_{min}, radius of biggest sphere r_{max} and height h of the cuboidal strip. Here, an equal distribution for the radii is applied, so the average radius is automatically varied by the variation of r_{min} and r_{max}. In step 2, multiple values for each of the factors were fixed. To cover a broad spectrum of instances, different values of the factors were

determined as shown in table 1. While the cuboid's width was kept constant for all instances (w = 10) the radii r_{min}, r_{max} and height h are given by different fractions of w.

Finally, in step 3 just one problem instance was generated at random for each admissible combination of the factor values.

Table 1. Values for the instances' parameters.

n	n_{dif}	r_{min}	r_{max}	h
20	n	w/8	w/4	w/2
30	n/2	w/16	w/6	3w/4
40	n/10	w/24	w/10	w
50	–	–	–	–

This procedure resulted in a total number of 4*3*(3*3-1)*3 = 288 instances. Note that the values of the five parameters are varied independently of each other; the combination $r_{min} = w/8$ and $r_{max} = w/10$ is invalid ($r_{min} > r_{max}$).

To obtain further 288 benchmark instances for the KP (referred to as KBG_3DKP), the KBG_3DSPP instances were modified only by adding a certain container length. For each SPP instance the cuboid length l was stipulated in such a way that the resulting cuboid volume $l \times w \times h$ is given by the total volume of all spheres multiplied by the coefficient 20/11. This coefficient ensures that a filling rate of 55% is achieved if all spheres of the instance are accommodated in the cuboid. The said percentage reflects the experience regarding the filling rates that can be reached for 3D-SPP instances. However, if the length l is smaller than the diameter $2 \times r_{max}$ of the largest sphere of the instance then l is set to $2 \times r_{max}$ to guarantee that each sphere can be packed into the cuboid.

The new benchmark instances serve to enable more meaningful and more reliable comparisons of solution methods for the 3D-SPP and 3D-KP with unequal spheres. Moreover, these instances can be used for exploring the influence of instance features, such as the heterogeneity of the sphere stock, on the solution quality achieved by heuristics. Both sets of benchmark instances are available from http://www.fernuni-hagen.de/WINF.

4 Experimental results and analysis

Both methods proposed above were coded in C and tested on the new KBG_3DKP and KBG_3DSPP instances, respectively. Moreover, the six 3D-SPP benchmark instances SYS1 to SYS6 introduced by Stoyan et al. (2003) were also calculated by B1.6_3DSPP. The tests were run on identical PCs using SUSE Linux 10.0 operating system (Linux kernel 2.6.13) with an AMD Athlon64 X2 3800+ (dual core) processor running at 2200MHz (overclocked) and 512 MB RAM each. Additional experiments were carried out by means of a Pentium quadcore PC (Dell Vostro 410, 2400MHz, 3 GB RAM) using SUSE Linux 11.0.

For algorithm B1.6_3DKP as well as for B1.6_3DSPP, 14 different threshold values were explored: -1000, -1, -0.5, 0, 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9, and 1.0. For value $\tau = -1000$, the algorithms correspond to method B1.0 from Huang et al. (2005), and for value $\tau = 1$, they correspond to method B1.5. The results are primarily presented that were calculated for $\tau = 0.8$. In addition, for B1.6_3DKP the results for $\tau = 1.0$ will also be shown for the purpose of comparison. On the one hand, the solution quality that is achieved for $\tau = 0.8$ is not much worse compared to $\tau = 1$. On the other hand, the results are calculated using only a small fraction (less than one tenth) of the runtime that is needed for $\tau = 1.0$.

In the following the results for the new benchmark instances are presented and analyzed. Afterwards algorithm B1.6_3DSPP is compared to the method from Stoyan et al. (2003). Runtimes are given in seconds throughout.

4.1 Results and analysis for the new benchmark instances

For an analysis of the influence of different factors on runtime and solution quality, the results obtained with B1.6_3DKP and B1.6_3DSPP were averaged over groups of instances with equal values of instance parameters (see http://www.fernuni-hagen.de/WINF). Table 2 and Table 3 show the results for groups of KBG_3DKP instances with equal values for n and n_{dif} which were calculated for the threshold values $\tau = 0.8$ and $\tau = 1.0$.

In the last two columns the average density d_{avg} and runtime t_{avg} over all instances per group (n, n_{dif}) are given. q_{suc} gives the ratio of the number of instances per group for which a successful plan (including all spheres and therefore optimal) was found and the number of all instances of that group. $d_{avg,fail}$ and $t_{avg,fail}$ specify the average density and runtime for instances for which no successful plan was achieved, while $t_{avg,suc}$ indicates the average runtime for instances for which a successful plan was found. The densities of successful plans result from the data of the belonging instances and are therefore omitted.

Table 2. Averaged results for groups of KBG_3DKP instances with equal n and n_{dif} values ($\tau = 0.8$).

n	n_{dif}	$d_{avg,fail}$ (%)	$t_{avg,fail}$ (s)	$t_{avg,suc}$ (s)	q_{suc}	d_{avg} (%)	t_{avg} (s)
20	20	46.368	16	4	7/24	41.798	13
	10	46.433	5	<0.01	7/24	42.389	3
	2	43.792	<0.01	<0.01	10/24	38.748	<0.01
30	30	49.224	550	<0.01	4/24	46.519	458
	15	48.877	116	<0.01	4/24	46.711	96
	3	46.194	2	<0.01	2/24	46.210	2
40	40	50.553	9438	<0.01	3/24	49.439	8259
	20	50.941	1047	<0.01	2/24	49.845	960
	4	48.670	15	<0.01	1/24	48.486	14
50	50	51.042	8273	759	2/24	51.020	7647
	25	51.203	6585	<0.01	2/24	50.564	6037
	5	49.845	77	19	4/24	50.573	67

Some important facts and tendencies are the following:

- For many groups the inequality $d_{avg,fail} > d_{avg}$ holds while a successful plan should have a density of about 55%. But there is no contradiction since in each such group some instances exist for which the length of the cuboid was enlarged to the biggest sphere's diameter (see above).
- The average density d_{avg} increases generally with the number of sphere types n_{dif} for constant n. This trend seems plausible, as a larger heterogeneity of spheres allows for a better interlocking of the spheres and, furthermore, yields a larger number of possible corner placements during the whole process of creating a plan.
- The average runtime t_{avg} increases when n_{dif} is increased for constant n values, as more initial plans have to be checked and the list of possible placements has more elements leading also to larger numbers of possible corner placements.
- Due to the way the benchmark instances were generated, there are instances for which successful plans can be found. If a successful plan is found, the algorithm stops immediately and no further initial plans are checked. Thus, the average runtime $t_{avg,fail}$ is often orders of magnitude higher than $t_{avg,suc}$.

- A comparison of the average runtimes for $n_{dif} = n$ and $n_{dif} = n/10$ for equal n values indicates that the speedup achieved with the avoidance of redundancy in corner placements (initial plans, list elements) is approximately in the range of 200 to 2000.
- For $\tau = 1.0$ more successful plans are generated and the average density d_{avg} is significantly higher but runtimes increase considerably compared to $\tau = 0.8$.

Table 3. Averaged results for groups of KBG_3DKP instances with equal n and n_{dif} values ($\tau = 1.0$).

n	n_{dif}	$d_{avg,fail}$ (%)	$t_{avg,fail}$ (s)	$t_{avg,suc}$ (s)	q_{suc}	d_{avg} (%)	t_{avg} (s)
20	20	49.571	2208	6	7/24	44.067	1566
	10	48.086	301	<0.01	7/24	43.560	213
	2	44.804	1	<0.01	10/24	39.338	1
30	30	50.901	53301	10646	10/24	48.941	35528
	15	50.962	7089	52	6/24	48.786	5330
	3	47.405	29	4	14/24	47.479	25
40	40	51.148	469900	12978	13/24	51.565	222400
	20	51.090	57697	5357	10/24	51.285	35889
	4	50.166	278	<0.01	1/24	49.920	266
50	50	51.605	405077	1250	12/24	52.950	203163
	25	51.600	311877	6344	8/24	52.345	159110
	5	50.639	1184	675	8/24	51.962	1014

For the KBG_3DSPP instances no successful plans exist and so only the averaged density and the averaged runtime are displayed. Table 4 includes the results for groups of KBG_3DSPP instances with equal values for n and n_{dif} which were calculated by algorithm B1.6_3DSPP for threshold value $\tau = 0.8$.

The average density d_{avg} increases with increasing n_{dif} for constant n in a similar way and for the same reasons as for B1.6_3DKP (exceptions: $n = 20$, $n_{dif} = 10, 20$, $n = 30$, $n_{dif} = 15, 30$). The average runtime increases monotonically with increasing n_{dif} at constant n, which can be explained in the same way as for B1.6_3DKP. The speedup achieved with the avoidance of redundancy in corner placements is in the same range as for the KBG_3DKP instances.

The runtime effort of the parallel versions of both of the algorithms is reduced to 52% compared to the sequential versions; the corresponding speedup factor is 96%.

Table 4. Averaged results for groups of KBG_3DSPP instances with equal n and n_{dif} values ($\tau = 0.8$).

n	n_{dif}	d_{avg} (%)	t_{avg} (s)
20	20	42.631	69
	10	42.762	10
	2	38.645	<0.1
30	30	47.198	1331
	15	47.612	180
	3	46.760	2
40	40	50.395	11080
	20	49.714	1308
	4	48.009	10
50	50	52.106	21793
	25	51.215	6225
	5	50.287	33

To study the runtime consumption for a slightly greater number of processes further tests with two and four processes, respectively, were performed using the above-mentioned quadcore processor. The threshold value $\tau = 0.8$ was chosen throughout. Results are available for one third of the new 3D-KP and 3D-SPP instances, namely for all instances with height $h = 3/4w$ (cf. table 1). Let $q_{4,2}$ denote the quotient (runtime with 4 processes)/ (runtime with 2 processes) (in %). The ratio $q_{4,2}$ amounts to 55% for the 96 3D-KPP instances (corresponding speedup 91%) and $q_{4,2}$ equals 54% for the 96 3D-SPP instances (corresponding speedup 93%). Thus, doubling the number of processes leads again to a high, nearly linear speedup.

Finally, all the 576 new instances were calculated with four processes using the quadcore processor and $\tau = 0.8$. This time the trade-off between solution quality and runtime is studied in greater detail. In table 5 mean relative runtimes are presented for each problem type (KP, SPP) and for three different levels *l* of solution quality, namely 0.97, 0.98 and 0.99. If an instance *p* is solved the level *l* is achieved if the density of the incumbent best solution exceeds the product (max. calculated density(*p*))×*l*; runtime(*l*) denotes the time needed to reach level *l* while the *relative* runtime according to level *l* is given by the ratio *(*runtime(*l*)) / (total runtime (*p*)) (in %).

Table 5. Relative runtimes for different quality levels.

Problem type	mean relative runtimes (in %)		
	quality level 0.97	quality level 0.98	quality level 0.99
3D-Knapsack	2.9	7.5	15.8
3D-Strip Packing	8.9	14.1	24.5

As table 5 shows the runtimes can be reduced on average to (e.g.) 3% (KP) or 9% (SPP) of the original mean value if the solving process is stopped after a filling rate of 97% of the original final density was reached. Thus a slight or moderate deterioration of solution quality might be compensated by a considerable saving of runtime and this can help to make the methods applicable even if runtime is an important issue.

4.2 Comparison of methods B1.6_3DSPP and SYS

In table 6 algorithm B1.6_3DSPP is compared to method SYS proposed by Stoyan et al. (2003) using the six benchmark instances SYS1 to SYS6. For method SYS the achieved lengths (originally termed as heights) are shown while runtimes for single instances are not available. For algorithm B1.6_3DSPP the lengths obtained for the threshold values $\tau = 0.8$ and $\tau = 1.0$ are listed as well as the runtimes per instance. Improvements are calculated as (length(SYS) – length(B1.6_3DSPP)) / length(SYS) (in %).

Table 6. Comparison of methods SYS and B1.6_3DSPP.

Instance	SYS1	SYS2	SYS3	SYS4	SYS5	SYS6
No. of spheres	25	35	40	45	50	60
Method SYS:						
Length	9.8668	9.6221	9.4729	11.0862	11.6453	12.8416
Method B1.6_3DSPP:						
$\tau = 0.8$						
Time (s)	2	11	23	123	199	900
Length	9.6118	9.3952	9.2264	11.0778	11.4228	12.5074
Improvement (%)	2.58	2.36	2.60	0.08	1.91	2.60
$\tau = 1.0$						
Time (s)	3630	53845	178745	522235	382845	1694954
Length	9.2656	8.9301	8.7178	10.4042	10.9865	11.8399
Improvement (%)	6.09	7.19	7.97	6.15	5.66	7.80

B1.6_3DSPP provides better solutions than SYS for all six instances. For $\tau = 0.8$ the improvement is about 2.0% on average and instance SYS4 is an exception with an improvement of only 0.08%. For $\tau = 1.0$ the improvement is between 5.66% and 7.97%.

The runtimes of method SYS range between some seconds and 100 minutes on a Pentium PC with a cycle frequency of 90 MHz only. The runtimes of B1.6_3DSPP remain moderate for threshold value $\tau = 0.8$ and don't cross 15 minutes. But for $\tau = 1.0$ the runtimes are in part huge and lie between one hour (SYS1) and 470.8 hours (SYS6). For instance SYS3 the factor of runtime increase amounts to 7772.

However, a deeper analysis of the calculations carried out with threshold $\tau = 1.0$ makes aware that a considerably better solution quality compared to method SYS can also be achieved with relatively low to moderate run times. Table 7 includes the runtimes necessary to obtain a length improvement of 0.5%, 1%, 2%, 3%, 4% and 5% compared to the results of SYS. Obviously, a substantial improvement of 2 percent or more can be achieved with moderate runtimes of less than 8 minutes.

Table 7. Runtimes of method B1.6_3DSPP ($\tau = 1.0$) necessary for different length improvements compared to results of method SYS.

Improvement (%)	SYS1 (s)	SYS2 (s)	SYS3 (s)	SYS4 (s)	SYS5 (s)	SYS6 (s)
0.5	1	1	6	157	91	153
1	4	3	9	157	94	163
2	12	10	64	175	163	469
3	17	194	71	5624	593	1603
4	164	701	118	51860	679	1629
5	165	725	147	354762	55244	3819

5 Summary

This paper presents two greedy algorithms for the 3D Knapsack Problem (3D-KP) and the 3D Strip Packing Problem (3D-SPP), each with unequal spheres. Both algorithms are derived from the methods B1.0 und B1.5 put forward by Huang et al. (2005) which, however, only address the decision problem corresponding to the 2D-KP. Important enhancements to the new algorithms, called B1.6_3DKP and B1.6_3DSPP, are dedicated to controlling the trade-off between solution quality and runtime effort, to select suitable sets of initial plans for constructing complete solutions, and to avoid redundancy in handling of sphere placements. Finally, the greedy algorithms are

parallelized using a shared memory master-slave approach to investigate initial plans simultaneously.

In a comparison algorithm B1.6_3DSPP achieved a considerably better solution quality than the method put forward by Stoyan et al. (2003). Moreover, the test revealed that for both the 3D-SPP and the 3D-KP a relatively high solution quality can be also be achieved using a moderate amount of runtime. Nevertheless, the results obtained indicate that further work is necessary to reach a similar solution quality at lower computational cost.

Furthermore, 288 new benchmark instances each for the spherical 3D-KP and for the spherical 3D-SPP were introduced to enable more meaningful and reliable comparisons of solution methods. First results for the new instances generated by B1.6_3DKP and B1.6_3DSPP show the usefulness of the enhancements mentioned above including the parallelization.

References

Bortfeldt A. and Gehring H.: *New Large benchmark instances for the two-dimensional strip packing problem with rectangular pieces*. Proceedings of the 39th Annual Hawaii International Conference of System Sciences (HICCS'06) 2006, Track 2, p. 30b.

Crainic T.G. and Toulouse M.: *Parallel strategies for meta-heuristics*. In: Glover F. and Kochenberger G. (Eds): *Handbook on Metaheuristics*, Kluwer Academic Publishers, Boston, 2003, pp. 475–513.

George J.A., George J.M. and Lamer B.W.: *Packing different-sized circles into a rectangular container*. European Journal of Operational Research 84(1995), pp. 693–712.

Glover F. and Kochenberger G. (Eds): *Handbook on metaheuristics*, Kluwer Academic Publishers, Boston, 2003.

Hifi M. and M'Hallah R.: *Approximate algorithms for constrained circular cutting problems*. Computers and Operations Research 31(2004), pp. 675–694.

Hifi M., Paschos V.Th. and Zissimopoulos V.A.: *Simulated annealing approach for the circular cutting problem*. European Journal of Operational Research 159(2004), pp. 430–448.

Huang W.Q., Li Y., Akeb H. and Li C.M.: *Greedy algorithms for packing unequal circles into a rectangular container.* Journal of the Operational Research Society 56(2005), pp. 539–548.

Lenstra J.K. and Rinnooy Kan A.H.G.: *Complexity of packing, covering, and partitioning problems*. In: Schrijver A (Ed): *Packing and Covering in Combinatorics*. Mathematisch Centrum, Amsterdam, 1979, pp. 275–291.

Stoyan Y.Y., Yaskow G. and Scheithauer G.: *Packing of various radii solid spheres into a parallelepiped*. Central European Journal of Operational Research 11(2003), pp. 389–407.

Stoyan Y.Y. and Yaskov G.: A mathematical model and a solution method for the problem of placing various-sized circles into a strip. European Journal of Operational Research 156(2004), pp. 590–600.

Sutou, A. and Dai, Y.: *Global optimization approach to unequal sphere packing problems in 3D*, Journal of Optimization Theory and Applications, 114 (2002), pp. 671–694.

Visscher, W. and Bolsterli, M.: Random packing of equal and unequal spheres in two and three dimensions, Nature, 239(1972), pp. 504–507.

Wang, J.: *Packing of unequal spheres and automated radiosurgical treatment planning*, Journal of Combinatorial Optimization, 3(1999), pp. 453–463.

Die Diskussionspapiere ab Nr. 183 (1992) bis heute, können Sie im Internet unter http://www.fernuni-hagen.de/FBWIWI/ einsehen und zum Teil downloaden.
Die **Titel** der Diskussionspapiere von Nr 1 (1975) bis 182 (1991) können bei Bedarf in der Fakultät für Wirtschaftswissenschaft angefordert werden:
FernUniversität, z. Hd. Frau Huber oder Frau Mette, Postfach 940, 58084 Hagen .
Die Diskussionspapiere selber erhalten Sie nur in den Bibliotheken.

Nr	Jahr	Titel	Autor/en
322	2001	Spreading Currency Crises: The Role of Economic Interdependence	Berger, Wolfram Wagner, Helmut
323	2002	Planung des Fahrzeugumschlags in einem Seehafen-Automobilterminal mittels eines Multi-Agenten-Systems	Fischer, Torsten Gehring, Hermann
324	2002	A parallel tabu search algorithm for solving the container loading problem	Bortfeldt, Andreas Gehring, Hermann Mack, Daniel
325	2002	Die Wahrheit entscheidungstheoretischer Maximen zur Lösung von Individualkonflikten - Unsicherheitssituationen -	Mus, Gerold
326	2002	Zur Abbildungsgenauigkeit des Gini-Koeffizienten bei relativer wirtschaftlicher Konzentration	Steinrücke, Martin
327	2002	Entscheidungsunterstützung bilateraler Verhandlungen über Auftragsproduktionen - eine Analyse aus Anbietersicht	Steinrücke, Martin
328	2002	Die Relevanz von Marktzinssätzen für die Investitionsbeurteilung – zugleich eine Einordnung der Diskussion um die Marktzinsmethode	Terstege, Udo
329	2002	Evaluating representatives, parliament-like, and cabinet-like representative bodies with application to German parliament elections 2002	Tangian, Andranik S.
330	2002	Konzernabschluss und Ausschüttungsregelung im Konzern. Ein Beitrag zur Frage der Eignung des Konzernabschlusses als Ausschüttungsbemessungsinstrument	Hinz, Michael
331	2002	Theoretische Grundlagen der Gründungsfinanzierung	Bitz, Michael
332	2003	Historical background of the mathematical theory of democracy	Tangian, Andranik S.
333	2003	MCDM-applications of the mathematical theory of democracy: choosing travel destinations, preventing traffic jams, and predicting stock exchange trends	Tangian, Andranik S.
334	2003	Sprachregelungen für Kundenkontaktmitarbeiter – Möglichkeiten und Grenzen	Fließ, Sabine Möller, Sabine Momma, Sabine Beate

335	2003	A Non-cooperative Foundation of Core-Stability in Positive Externality NTU-Coalition Games	Finus, Michael Rundshagen, Bianca
336	2003	Combinatorial and Probabilistic Investigation of Arrow's dictator	Tangian, Andranik
337	2003	A Grouping Genetic Algorithm for the Pickup and Delivery Problem with Time Windows	Pankratz, Giselher
338	2003	Planen, Lernen, Optimieren: Beiträge zu Logistik und E-Learning. Festschrift zum 60 Geburtstag von Hermann Gehring	Bortfeldt, Andreas Fischer, Torsten Homberger, Jörg Pankratz, Giselher Strangmeier, Reinhard
339a	2003	Erinnerung und Abruf aus dem Gedächtnis Ein informationstheoretisches Modell kognitiver Prozesse	Rödder, Wilhelm Kuhlmann, Friedhelm
339b	2003	Zweck und Inhalt des Jahresabschlusses nach HGB, IAS/IFRS und US-GAAP	Hinz, Michael
340	2003	Voraussetzungen, Alternativen und Interpretationen einer zielkonformen Transformation von Periodenerfolgsrechnungen – ein Diskussionsbeitrag zum LÜCKE-Theorem	Terstege, Udo
341	2003	Equalizing regional unemployment indices in West and East Germany	Tangian, Andranik
342	2003	Coalition Formation in a Global Warming Game: How the Design of Protocols Affects the Success of Environmental Treaty-Making	Eyckmans, Johan Finus, Michael
343	2003	Stability of Climate Coalitions in a Cartel Formation Game	Finus, Michael van Ierland, Ekko Dellink, Rob
344	2003	The Effect of Membership Rules and Voting Schemes on the Success of International Climate Agreements	Finus, Michael J.-C., Altamirano-Cabrera van Ierland, Ekko
345	2003	Equalizing structural disproportions between East and West German labour market regions	Tangian, Andranik
346	2003	Auf dem Prüfstand: Die geldpolitische Strategie der EZB	Kißmer, Friedrich Wagner, Helmut
347	2003	Globalization and Financial Instability: Challenges for Exchange Rate and Monetary Policy	Wagner, Helmut

348	2003	Anreizsystem Frauenförderung – Informationssystem Gleichstellung am Fachbereich Wirtschaftswissenschaft der FernUniversität in Hagen	Fließ, Sabine Nonnenmacher, Dirk
349	2003	Legitimation und Controller	Pietsch, Gotthard Scherm, Ewald
350	2003	Controlling im Stadtmarketing – Ergebnisse einer Primärerhebung zum Hagener Schaufenster-Wettbewerb	Fließ, Sabine Nonnenmacher, Dirk
351	2003	Zweiseitige kombinatorische Auktionen in elektronischen Transportmärkten – Potenziale und Probleme	Pankratz, Giselher
352	2003	Methodisierung und E-Learning	Strangmeier, Reinhard Bankwitz, Johannes
353 a	2003	A parallel hybrid local search algorithm for the container loading problem	Mack, Daniel Bortfeldt, Andreas Gehring, Hermann
353 b	2004	Übernahmeangebote und sonstige öffentliche Angebote zum Erwerb von Aktien – Ausgestaltungsmöglichkeiten und deren Beschränkung durch das Wertpapiererwerbs- und Übernahmegesetz	Wirtz, Harald
354	2004	Open Source, Netzeffekte und Standardisierung	Maaß, Christian Scherm, Ewald
355	2004	Modesty Pays: Sometimes!	Finus, Michael
356	2004	Nachhaltigkeit und Biodiversität	Endres, Alfred Bertram, Regina
357	2004	Eine Heuristik für das dreidimensionale Strip-Packing-Problem	Bortfeldt, Andreas Mack, Daniel
358	2004	Netzwerkökonomik	Martiensen, Jörn
359	2004	Competitive versus cooperative Federalism: Is a fiscal equalization scheme necessary from an allocative point of view?	Arnold, Volker
360	2004	Gefangenendilemma bei Übernahmeangeboten? Eine entscheidungs- und spieltheoretische Analyse unter Einbeziehung der verlängerten Annahmefrist gem. § 16 Abs. 2 WpÜG	Wirtz, Harald
361	2004	Dynamic Planning of Pickup and Delivery Operations by means of Genetic Algorithms	Pankratz, Giselher

362	2004	Möglichkeiten der Integration eines Zeitmanagements in das Blueprinting von Dienstleistungsprozessen	Fließ, Sabine Lasshof, Britta Meckel, Monika
363	2004	Controlling im Stadtmarketing - Eine Analyse des Hagener Schaufensterwettbewerbs 2003	Fließ, Sabine Wittko, Ole
364	2004	Ein Tabu Search-Verfahren zur Lösung des Timetabling-Problems an deutschen Grundschulen	Desef, Thorsten Bortfeldt, Andreas Gehring, Hermann
365	2004	Die Bedeutung von Informationen, Garantien und Reputation bei integrativer Leistungserstellung	Prechtl, Anja Völker-Albert, Jan-Hendrik
366	2004	The Influence of Control Systems on Innovation: An empirical Investigation	Littkemann, Jörn Derfuß, Klaus
367	2004	Permit Trading and Stability of International Climate Agreements	Altamirano-Cabrera, Juan-Carlos Finus, Michael
368	2004	Zeitdiskrete vs. zeitstetige Modellierung von Preismechanismen zur Regulierung von Angebots- und Nachfragemengen	Mazzoni, Thomas
369	2004	Marktversagen auf dem Softwaremarkt? Zur Förderung der quelloffenen Softwareentwicklung	Christian Maaß Ewald Scherm
370	2004	Die Konzentration der Forschung als Weg in die Sackgasse? Neo-Institutionalistische Überlegungen zu 10 Jahren Anreizsystemforschung in der deutschsprachigen Betriebswirtschaftslehre	Süß, Stefan Muth, Insa
371	2004	Economic Analysis of Cross-Border Legal Uncertainty: the Example of the European Union	Wagner, Helmut
372	2004	Pension Reforms in the New EU Member States	Wagner, Helmut
373	2005	Die Bundestrainer-Scorecard Zur Anwendbarkeit des Balanced Scorecard Konzepts in nicht-ökonomischen Fragestellungen	Eisenberg, David Schulte, Klaus
374	2005	Monetary Policy and Asset Prices: More Bad News for ‚Benign Neglect“	Berger, Wolfram Kißmer, Friedrich Wagner, Helmut
375	2005	Zeitstetige Modellbildung am Beispiel einer volkswirtschaftlichen Produktionsstruktur	Mazzoni, Thomas

376	2005	Economic Valuation of the Environment	Endres, Alfred
377	2005	Netzwerkökonomik – Eine vernachlässigte theoretische Perspektive in der Strategie-/Marketingforschung?	Maaß, Christian Scherm, Ewald
378	2005	Diversity management`s diffusion and design: a study of German DAX-companies and Top-50-U.S.-companies in Germany	Süß, Stefan Kleiner, Markus
379	2005	Fiscal Issues in the New EU Member Countries – Prospects and Challenges	Wagner, Helmut
380	2005	Mobile Learning – Modetrend oder wesentlicher Bestandteil lebenslangen Lernens?	Kuszpa, Maciej Scherm, Ewald
381	2005	Zur Berücksichtigung von Unsicherheit in der Theorie der Zentralbankpolitik	Wagner, Helmut
382	2006	Effort, Trade, and Unemployment	Altenburg, Lutz Brenken, Anke
383	2006	Do Abatement Quotas Lead to More Successful Climate Coalitions?	Altamirano-Cabrera, Juan-Carlos Finus, Michael Dellink, Rob
384	2006	Continuous-Discrete Unscented Kalman Filtering	Singer, Hermann
385	2006	Informationsbewertung im Spannungsfeld zwischen der Informationstheorie und der Betriebswirtschaftslehre	Reucher, Elmar
386	2006	The Rate Structure Pattern: An Analysis Pattern for the Flexible Parameterization of Charges, Fees and Prices	Pleß, Volker Pankratz, Giselher Bortfeldt, Andreas
387a	2006	On the Relevance of Technical Inefficiencies	Fandel, Günter Lorth, Michael
387b	2006	Open Source und Wettbewerbsstrategie - Theoretische Fundierung und Gestaltung	Maaß, Christian
388	2006	Induktives Lernen bei unvollständigen Daten unter Wahrung des Entropieprinzips	Rödder, Wilhelm
389	2006	Banken als Einrichtungen zur Risikotransformation	Bitz, Michael
390	2006	Kapitalerhöhungen börsennotierter Gesellschaften ohne börslichen Bezugsrechtshandel	Terstege, Udo Stark, Gunnar
391	2006	Generalized Gauss-Hermite Filtering	Singer, Hermann

392	2006	Das Göteborg Protokoll zur Bekämpfung grenzüberschreitender Luftschadstoffe in Europa: Eine ökonomische und spieltheoretische Evaluierung	Ansel, Wolfgang Finus, Michael
393	2006	Why do monetary policymakers lean with the wind during asset price booms?	Berger, Wolfram Kißmer, Friedrich
394	2006	On Supply Functions of Multi-product Firms with Linear Technologies	Steinrücke, Martin
395	2006	Ein Überblick zur Theorie der Produktionsplanung	Steinrücke, Martin
396	2006	Parallel greedy algorithms for packing unequal circles into a strip or a rectangle	Timo Kubach, Bortfeldt, Andreas Gehring, Hermann
397	2006	C&P Software for a cutting problem of a German wood panel manufacturer – a case study	Papke, Tracy Bortfeldt, Andreas Gehring, Hermann
398	2006	Nonlinear Continuous Time Modeling Approaches in Panel Research	Singer, Hermann
399	2006	Auftragsterminierung und Materialflussplanung bei Werkstattfertigung	Steinrücke, Martin
400	2006	Import-Penetration und der Kollaps der Phillips-Kurve	Mazzoni, Thomas
401	2006	Bayesian Estimation of Volatility with Moment-Based Nonlinear Stochastic Filters	Grothe, Oliver Singer, Hermann
402	2006	Generalized Gauss-H ermite Filtering for Multivariate Diffusion Processes	Singer, Hermann
403	2007	A Note on Nash Equilibrium in Soccer	Sonnabend, Hendrik Schlepütz, Volker
404	2007	Der Einfluss von Schaufenstern auf die Erwartungen der Konsumenten - eine explorative Studie	Fließ, Sabine Kudermann, Sarah Trell, Esther
405	2007	Die psychologische Beziehung zwischen Unternehmen und freien Mitarbeitern: Eine empirische Untersuchung des Commitments und der arbeitsbezogenen Erwartungen von IT-Freelancern	Süß, Stefan
406	2007	An Alternative Derivation of the Black-Scholes Formula	Zucker, Max Singer, Hermann
407	2007	Computational Aspects of Continuous-Discrete Extended Kalman-Filtering	Mazzoni, Thomas
408	2007	Web 2.0 als Mythos, Symbol und Erwartung	Maaß, Christian Pietsch, Gotthard
409	2007	„Beyond Balanced Growth“: Some Further Results	Stijepic, Denis Wagner, Helmut

410	2007	Herausforderungen der Globalisierung für die Entwicklungsländer: Unsicherheit und geldpolitisches Risikomanagement	Wagner, Helmut
411	2007	Graphical Analysis in the New Neoclassical Synthesis	Giese, Guido Wagner, Helmut
412	2007	Monetary Policy and Asset Prices: The Impact of Globalization on Monetary Policy Trade-Offs	Berger, Wolfram Kißmer, Friedrich Knütter, Rolf
413	2007	Entropiebasiertes Data Mining im Produktdesign	Rudolph, Sandra Rödder, Wilhelm
414	2007	Game Theoretic Research on the Design of International Environmental Agreements: Insights, Critical Remarks and Future Challenges	Finus, Michael
415	2007	Qualitätsmanagement in Unternehmenskooperationen - Steuerungsmöglichkeiten und Datenintegrationsprobleme	Meschke, Martina
416	2007	Modernisierung im Bund: Akteursanalyse hat Vorrang	Pietsch, Gotthard Jamin, Leander
417	2007	Inducing Technical Change by Standard Oriented Evirnonmental Policy: The Role of Information	Endres, Alfred Bertram, Regina Rundshagen, Bianca
418	2007	Der Einfluss des Kontextes auf die Phasen einer SAP-Systemimplementierung	Littkemann, Jörn Eisenberg, David Kuboth, Meike
419	2007	Endogenous in Uncertainty and optimal Monetary Policy	Giese, Guido Wagner, Helmut
420	2008	Stockkeeping and controlling under game theoretic aspects	Fandel, Günter Trockel, Jan
421	2008	On Overdissipation of Rents in Contests with Endogenous Intrinsic Motivation	Schlepütz, Volker
422	2008	Maximum Entropy Inference for Mixed Continuous-Discrete Variables	Singer, Hermann
423	2008	Eine Heuristik für das mehrdimensionale Bin Packing Problem	Mack, Daniel Bortfeldt, Andreas
424	2008	Expected A Posteriori Estimation in Financial Applications	Mazzoni, Thomas
425	2008	A Genetic Algorithm for the Two-Dimensional Knapsack Problem with Rectangular Pieces	Bortfeldt, Andreas Winter, Tobias
426	2008	A Tree Search Algorithm for Solving the Container Loading Problem	Fanslau, Tobias Bortfeldt, Andreas
427	2008	Dynamic Effects of Offshoring	Stijepic, Denis Wagner, Helmut
428	2008	Der Einfluss von Kostenabweichungen auf das Nash-Gleichgewicht in einem nicht-kooperativen Disponenten-Controller-Spiel	Fandel, Günter Trockel, Jan
429	2008	Fast Analytic Option Valuation with GARCH	Mazzoni, Thomas
430	2008	Conditional Gauss-Hermite Filtering with Application to Volatility Estimation	Singer, Hermann
431	2008	Web 2.0 auf dem Prüfstand: Zur Bewertung von Internet-Unternehmen	Christian Maaß Gotthard Pietsch

432	2008	Zentralbank-Kommunikation und Finanzstabilität – Eine Bestandsaufnahme	Knütter, Rolf Mohr, Benjamin
433	2008	Globalization and Asset Prices: Which Trade-Offs Do Central Banks Face in Small Open Economies?	Knütter, Rolf Wagner, Helmut
434	2008	International Policy Coordination and Simple Monetary Policy Rules	Berger, Wolfram Wagner, Helmut
435	2009	Matchingprozesse auf beruflichen Teilarbeitsmärkten	Stops, Michael Mazzoni, Thomas
436	2009	Wayfindingprozesse in Parksituationen - eine empirische Analyse	Fließ, Sabine Tetzner, Stefan
437	2009	ENTROPY-DRIVEN PORTFOLIO SELECTION a downside and upside risk framework	Rödder, Wilhelm Gartner, Ivan Ricardo Rudolph, Sandra
438	2009	Consulting Incentives in Contests	Schlepütz, Volker
439	2009	A Genetic Algorithm for a Bi-Objective Winner-Determination Problem in a Transportation-Procurement Auction	Buer, Tobias Pankratz, Giselher
440	2009	Parallel greedy algorithms for packing unequal spheres into a cuboidal strip or a cuboid	Kubach, Timo Bortfeldt, Andreas Tilli, Thomas Gehring, Hermann