Fraunhofer Institut
Techno- und
Wirtschaftsmathematik

P. Domínguez-Marín, P. Hansen,
N. Mladenović, S. Nickel

Heuristic Procedures for Solving the Discrete Ordered Median Problem

Berichte des Fraunhofer ITWM, Nr. 46 (2003)

ISSN 1434-9973

Bericht 46 (2003)

Die Veröffentlichungen in der Berichtsreihe des Fraunhofer ITWM können bezogen werden über:

Fraunhofer-Institut für Techno- und
Wirtschaftsmathematik ITWM
Gottlieb-Daimler-Straße, Geb. 49

67663 Kaiserslautern

Telefon: +49 (0) 6 31/2 05-32 42
Telefax: +49 (0) 6 31/2 05-41 39
E-Mail: info@itwm.fraunhofer.de
Internet: www.itwm.fraunhofer.de

Vorwort

Das Tätigkeitsfeld des Fraunhofer Instituts für Techno- und Wirtschaftsmathematik ITWM umfasst anwendungsnahe Grundlagenforschung, angewandte Forschung sowie Beratung und kundenspezifische Lösungen auf allen Gebieten, die für Techno- und Wirtschaftsmathematik bedeutsam sind.

In der Reihe »Berichte des Fraunhofer ITWM« soll die Arbeit des Instituts kontinuierlich einer interessierten Öffentlichkeit in Industrie, Wirtschaft und Wissenschaft vorgestellt werden. Durch die enge Verzahnung mit dem Fachbereich Mathematik der Universität Kaiserslautern sowie durch zahlreiche Kooperationen mit internationalen Institutionen und Hochschulen in den Bereichen Ausbildung und Forschung ist ein großes Potenzial für Forschungsberichte vorhanden. In die Berichtreihe sollen sowohl hervorragende Diplom- und Projektarbeiten und Dissertationen als auch Forschungsberichte der Institutsmitarbeiter und Institutsgäste zu aktuellen Fragen der Techno- und Wirtschaftsmathematik aufgenommen werden.

Darüberhinaus bietet die Reihe ein Forum für die Berichterstattung über die zahlreichen Kooperationsprojekte des Instituts mit Partnern aus Industrie und Wirtschaft.

Berichterstattung heißt hier Dokumentation darüber, wie aktuelle Ergebnisse aus mathematischer Forschungs- und Entwicklungsarbeit in industrielle Anwendungen und Softwareprodukte transferiert werden, und wie umgekehrt Probleme der Praxis neue interessante mathematische Fragestellungen generieren.

Prof. Dr. Dieter Prätzel-Wolters
Institutsleiter

Kaiserslautern, im Juni 2001

Heuristic Procedures for Solving the Discrete Ordered Median Problem

Patricia Domínguez-Marín * Stefan Nickel * Pierre Hansen †
Nenad Mladenović ‡

January 14, 2003

Abstract

We present two heuristic methods for solving the Discrete Ordered Median Problem (DOMP), for which no such approaches have been developed so far. The DOMP generalizes classical discrete facility location problems, such as the p-median, p-center and Uncapacitated Facility Location problems. The first procedure proposed in this paper is based on a genetic algorithm developed by Moreno Vega [MV96] for p-median and p-center problems. Additionally, a second heuristic approach based on the Variable Neighborhood Search metaheuristic (VNS) proposed by Hansen & Mladenović [HM97] for the p-median problem is described. An extensive numerical study is presented to show the efficiency of both heuristics and compare them.

Keywords: Genetic algorithms, Variable Neighborhood Search, Discrete facility location.

1 Introduction

The Discrete Ordered Median Problem (DOMP) is a generalization of classical discrete facility location problems. Such problems have been widely studied due to their importance in practical applications, see for example Daskin [Das95], Drezner & Hamacher [DE02], Mirchandani & Francis [MF90], and references therein. They typically involve a finite set of *sites* at which facilities can be located, and a finite set of *clients*, whose demands have to be satisfied from the facilities. Whilst numerous alternatives have been considered in the literature, we focus on problems in which a fixed number of facilities must be located at sites chosen among a given set of candidate sites, and in which any client can only be

*Dept. of Optimization, Fraunhofer Institut für Techno- und Wirtschaftsmathematik, D-67663 Kaiserslautern, Germany, fax: +49-631-2054139, e-mail:{dominguez}{nickel}@itwm.fhg.de

†GERAD, and MEC Montréal, 3000 Chamin de la Côte-Sainte-Catherine, Montréal (Québec), Canada H3T 2A7

supplied from a single facility. For each client-site pair, there is a given cost for meeting the demand of the client from a facility located at the site.

The DOMP was first introduced by Nickel [Nic01] and later investigated in more detail by Boland et al. [BDMNP03]. The objective function of the DOMP generalizes the most popular facility location objective functions: *median*, *center*, and *centdian*. For planar and network location problems the corresponding model was studied by Puerto & Fernández [PF95] and [PF00], Nickel & Puerto [NP99], Francis et al. [FLT00], Rodríguez-Chía et al. [RCNPF00] and Kalcsics et al. [KNP03]. Several exact procedures to solve the DOMP were proposed by Boland et al. [BDMNP03]. However, these exact methods could only solve instances whose sizes were far from those of real-life problems.

In this paper the first two heuristic approaches to solve the DOMP are proposed. Both procedures provide, at least, near-optimal solutions for large instances (which are close to real-life problem sizes) in an acceptable computing time.

The first heuristic method is an Evolution Program (EP) based on genetic algorithms, which were first introduced by Holland [Hol75] as a powerful tool to solve optimization problems, see for example Davis [Dav87] and Goldberg [Gol89]. This procedure provides, following an easy approach, relative good feasible solutions of the DOMP in reasonable computing time.

The second heuristic method is based on Variable Neighborhood Search (VNS), a metaheuristic to solve combinatorial problems first proposed by Mladenović [Mla95] and Mladenović & Hansen [MH97]. VNS is a very well-known technique often used to solve discrete facility location problems, see e.g. Mladenović et al. [MLH00]. This method usually provides high quality solutions. Therefore, we adapt it for solving the DOMP. Results are helpful in determining the efficiency of the first heuristic.

The remainder of the paper is organized as follows. In the next section, we present a formal definition of the DOMP and an illustrative example. In Sections 3 and 4 we introduce an Evolution Program (EP) and a Variable Neighborhood Search (VNS) to solve the DOMP. A description of our random problem generator, the test problem sets, and the obtained computational results are shown in Section 5. Finally, in Section 6 we present our conclusions and suggestions for further research.

2 Definition of the DOMP

Let A denote a given set of m sites, and identified with the integers $1, \ldots, m$, i.e. $A = \{1, \ldots, m\}$. Let $C = (c_{ij})_{i,j=1,\ldots,m}$ be a given non-negative $m \times m$ cost matrix, where c_{ij} denotes the cost of satisfying the demand of client i from a facility located at site j. As is customary in discrete location problems, we assume without loss of generality that the number of candidate sites is identical to the number of clients. Let $p \leq m$ be the number of facilities to be located at the candidate sites. Since we assume no capacity constraint, a solution to the location problem is given by a set $X \subseteq A$ of p sites, i.e. $|X| = p$. In such a solution, each client i will be served by a facility located at the site j of X with the lowest

cost for satisfying his demand, i.e.

$$c_{ij} = c_i(X) := \min_{k \in X} c_{ik}. \tag{1}$$

What distinguishes the DOMP from a usual single-supplier uncapacitated facility location problem is its objective function. In order to calculate this function, the costs for supplying clients, $c_1(X), \ldots, c_m(X)$, are sorted in non-decreasing order. We define σ_X to be a permutation on $\{1, \ldots, m\}$ for which the inequalities

$$c_{\sigma_X(1)}(X) \leq c_{\sigma_X(2)}(X) \leq \cdots \leq c_{\sigma_X(m)}(X)$$

hold. Then, the objective applies a linear cost factor, with coefficient $\lambda_i \geq 0$, to the ith lowest cost of supplying a client, $c_{\sigma_X(i)}(X)$, for each $i = 1, \ldots, m$. Let $\Lambda = (\lambda_1, \ldots, \lambda_m)$ with $\lambda_i \geq 0$, $i = 1, \ldots, m$. The Discrete Ordered Median Problem (DOMP) is defined as

$$\min_{X \subseteq A\,,\, |X|=p} \sum_{i=1}^{m} \lambda_i \, c_{\sigma_X(i)}(X) \; .$$

For different choices of Λ we obtain different types of objective functions. To see that the DOMP objective generalizes well-known location objectives, note that taking $\Lambda = (1, 1, \ldots, 1)$ yields the p-median problem; taking $\Lambda = (0, 0, \ldots, 0, 1)$ gives the p-center problem; taking $\Lambda = (\mu, \mu, \ldots, \mu, 1)$ for $0 < \mu < 1$ leads to the μ-centdian problem, with a convex combination of the median and the center objective functions; finally taking $\Lambda = (0, \ldots, 0, 1, \ldots, 1)$, where the first $m - k$ entries are zero and the last k entries are one, leads to the k-centra problem of minimizing the average cost of the k most expensive clients. Other objectives may also be of practical interest. One example is to take $\Lambda = (0, \ldots, 0, 1, \ldots, 1, 0, \ldots, 0)$, where the first k_1 and last k_2 entries are zero, and the middle $m - k_1 - k_2$ entries are one: this leads to a problem in which the k_1 lowest costs and the k_2 highest costs are disregarded and the average of the middle part, the so-called $k_1 + k_2$-trimmed mean, which is a robust statistic, is minimized. Another example would be to take $\Lambda = (1, \ldots, 1, 0, \ldots, 0, 1, \ldots, 1)$, where the first k_1 entries are one, the next $m - k_1 - k_2$ entries are zero, and the last k_2 entries are one: this leads to the problem of minimizing the sum of the k_1 lowest costs and the k_2 highest costs; the corresponding DOMP searches for a set of p facilities minimizing the average cost for the clients which are very close and very far away. Hence, classical location problems can easily be modelled. Moreover, new meaningful objective functions are easily derived, as shown above. An example presented by Nickel [Nic01] shows the sensitivity of the objective function on the optimal location of the new facilities.

Observe that DOMP belongs to the class of NP-hard problems, since it is a generalization of the p-median problem, which is NP-hard, see Kariv & Hakimi [KH79].

2.1 An Illustrative Example

Consider a DOMP with $m = 5$, $p = 2$, and the cost matrix

$$C = \begin{pmatrix} 0 & 4 & 5 & 3 & 3 \\ 5 & 0 & 6 & 2 & 2 \\ 7 & 2 & 0 & 5 & 6 \\ 7 & 4 & 3 & 0 & 5 \\ 1 & 3 & 2 & 4 & 0 \end{pmatrix}.$$

Let us show how the objective function value for the case $\Lambda = (0, 0, 1, 1, 0)$ (i.e. the 2+1 trimmed mean problem) is calculated. The optimal solution is formed by the facilities (1,4). Therefore, the demand of facilities 1 and 5 is satisfied by facility 1 while the demand of the remaining facilities is satisfied by facility 4. As a result, the associated cost vector is $(0, 2, 5, 0, 1)$. Thus, the sorted cost vector is $(0, 0, 1, 2, 5)$ and the optimal objective function value is equal to $0 \times 0 + 0 \times 0 + 1 \times 1 + 1 \times 2 + 0 \times 5 = 3$.

In the following section, we describe our evolution program for solving the DOMP.

3 An Evolution Program for the DOMP

The Evolution Program (EP) proposed to solve the DOMP is essentially based on a genetic algorithm developed by Moreno Vega [MV96] for p-median and p-center problems. First, it should be noted that both of these problems can be solved as particular cases of the DOMP. Second, the feasible solutions of these problems and those of the DOMP have a similar structure. These are the reasons to adapt the procedure of [MV96] to our problem. In addition, evolution strategies are used to improve the performance of the EP in each iteration.

In the next section we introduce some general concepts of genetic algorithms which are necessary to present our EP.

3.1 Genetic Algorithms

Genetic algorithms use a vocabulary taken from natural genetics. We talk about individuals in a population, in the literature these individuals are also called chromosomes. A chromosome is divided by units - genes, see Dawkins [Daw89]. These genes contain information about one or several characters.

The evolution in genetic algorithms can be implemented by two processes which mimic nature: natural selection and genetic change in the chromosomes or individuals. Natural selection consists of selecting those individuals that are better adapted to the environment, i.e. those who survive. Genetic changes (produced by genetic operators) can occur either when there exists a crossover between two individuals or when an individual undergoes a kind of mutation. The crossover transformation creates new individuals by combining parts from several (two or more) individuals. The mutations are unary transformations which

create new individuals by a small change in a single individual. After some generations the procedure converges - it is expected that the best individual represents a near-optimal (reasonable) solution.

In addition, several extensions of genetic algorithms have been developed (evolutionary algorithms, evolution algorithms and evolutive algorithms). These extensions mainly consist of using new data structures for representing the population members and including different types of genetic operators and natural selection, see Michalewicz [Mic96].

In the next section we introduce an EP to solve the DOMP.

3.2 Evolution Program

Classical genetic algorithms use a binary codification to define the chromosomes. But sometimes this representation is very difficult to handle and therefore, some authors decided not to use it, see Davis [Dav91] and Michalewicz [Mic96]. Genetic algorithms, which use codifications different from the binary one and genetic operators adapted to these particular codifications, are called evolution programs (see Michalewicz [Mic96]). In the following we will use a non-binary representation scheme for the individuals of the population.

An EP is a probabilistic algorithm which maintains a population of H individuals, $P(t) = \{x_1^t, \ldots, x_H^t\}$ in each iteration t. Each individual stands for a potential solution to the problem at hand, and is represented by some data structure. Some members of the population undergo transformations (alteration step) by means of genetic operators to form new solutions. Each solution is evaluated to give some measure of its "fitness". Then, a new population (iteration $t+1$) is formed by selecting the fittest individuals (selection step). The program finishes after a fixed number of iterations where the best individual of the last population is considered as the approximative solution of the problem. We present a scheme of an EP as follows:

procedure evolution program

$t \leftarrow 0$
initialize $P(t)$
while (not termination-condition) **do**
  modify $P(t)$
  evaluate $P(t)$
  $t \leftarrow t + 1$
  generate $P(t)$ from $P(t-1)$
end while

An EP for a particular problem must have the following six components (see Michalewicz [Mic96]):

- ◇ a genetic representation of potential solutions of the problem,
- ◇ a way to create an initial population of potential solutions,

- ◇ an evaluation function that plays the role of the environment, rating solutions in terms of their "fitness",
- ◇ genetic operators (crossover and mutation) that alter the composition of children,
- ◇ a selection criterion that determines the survival of every individual, allowing an individual to survive or not in the next iteration,
- ◇ values for various parameters that the genetic algorithm uses (population size, probabilities of applying genetic operators, number of generations, etc.).

All these components will be described in the following sections. First of all, we introduce a codification in order to have an appropriate representation of the individuals, that is, of the feasible solutions of the DOMP.

3.2.1 Codification of the Individuals

Taking into account that the set of existing facilities is finite, we can assume that it is indexed (see Section 2). Thus, the feasible solutions of a discrete facility location problem can be represented by an m-dimensional binary vector with exactly p entries equal to 1, see Hosage & Goodchild [HG86] and Jaramillo et al. [JBB02]. An i-th entry with value 1 means that facility i is open, the value 0 means that it is closed. The advantage of this codification is that the classical genetic operators (see Michalewicz [Mic96]) can be used. The disadvantages are that these operators do not generate, in general, feasible solutions and that the $m - p$ positions containing a zero also use memory while not providing any additional information.

Obviously, the classical binary codification can be used for the DOMP. But the disadvantage of the inefficiently used memory is especially clear for examples with $p \ll m$. For this reason, we represent the individuals as p-dimensional vectors containing the indices of the open facilities, as Moreno Pérez et al. [MPRGMV94] and Moreno Vega [MV96] proposed for the p-median and p-center problems, respectively. In addition, the entries of each vector (individual) are sorted in increasing order. The sorting is to assure that under the same conditions the crossover operator, to be defined in Section 3.2.4, always yields the same children solutions, see Moreno Vega [MV96]. We illustrate this representation of the individuals with a small example: if $m = 7$ and $p = 5$, the feasible solution $X = (0, 1, 0, 1, 1, 1, 1)$ is codified as $(2, 4, 5, 6, 7)$.

To start on evolution, an initial population is necessary. The process to generate this population is described in the following subsection.

3.2.2 Initial Population

Two kinds of initial populations were considered. The first one is completely randomly generated (denoted by *random* $P(0)$). The individuals of the second one (denoted by *greedy* $P(0)$) are all but one randomly generated, while the last one is a greedy solution of the DOMP. The number of individuals of the population in every generation, denoted

by H, is constant. In this way, the population *random* $P(0)$ is made up of H randomly generated individuals, and the *greedy* $P(0)$ of $H-1$ randomly generated and one solution of the DOMP constructed with the Greedy Algorithm.

A greedy solution of the DOMP is obtained as follows: the first chosen facility is the one that minimizes the ordered median objective function assuming that we are interested in the 1-facility case. After that, at every step we choose the facility with minimal objective function value, taking into account the facilities already selected. This procedure ends after exactly p facilities have been chosen.

Each individual of the population has an associated "fitness" value. In the following subsection the evaluation function that defines the "fitness" measure is described.

3.2.3 Evaluation Function

In order to solve the DOMP using an EP, the "fitness" of an individual is determined by its corresponding ordered median function value. Therefore, an individual will be better adapted to the environment than another one if and only if it yields a smaller ordered median function value. Thus, the best adapted individual of a population will be one that provides the minimal objective function value of the DOMP among all the individuals of this population.

3.2.4 Genetic Operators

The genetic operators presented in this section are replicas of the classical crossover and mutation operators, see Hosage & Goodchild [HG86]. These operators are adapted to the used codification (see Section 3.2.1) and they always provide feasible solutions, i.e. vectors of size p in which all entries are different. There are two of them:

- ▷ Crossover Operator

  In order to present the crossover operator we define the breaking position as the component where the two parent individuals break to generate two children. The crossover operator interchanges the indices placed on the right-hand side of the breaking position (randomly obtained). When the breaking position has been generated, the output of this operator depends only on the two parent individuals, i.e. their crossing always provides the same children. This is possible because of the sorting in the individual codification, as we mentioned in Section 3.2.1. Moreover, to ensure the feasibility during the crossing procedure, the indices of the parent individuals that should be interchanged (i.e. those indices which are common for both parent individuals) are marked. Observe that feasibility of an individual will be lost if it contains the same index more than once.

  The breaking position is randomly chosen. The indices placed to the right-hand side of the breaking position are called cross positions. Then the children are obtained as follows:

1. both parents are compared, the indices presented in both vectors are marked;
2. the non-marked indices are sorted (in increasing order) and moved to the left;
3. the indices of the transformed parents that lie on the right-hand side of the breaking position are interchanged;
4. the marks are eliminated and both children codifications are sorted.

▷ Mutation Operator

The mutation operator is defined as the classical one derived for the binary codification but guaranteeing the feasibility of the solution: interchange one index of the individual with another not presented in the individual. After the interchange the indices of the new individual are sorted.

These two operators are illustrated in the following example.

Example 3.1 *Assume that there are seven sites and five new facilities should be open, i.e.* $m = 7$ *and* $p = 5$*. Let us consider two feasible solutions:* $(2, 4, 5, 6, 7)$ *and* $(1, 2, 3, 5, 6)$*.*
Assume that the breaking position is 1. Then:

parents	*marks*	*sorted non-marked indices*	*interchange*	*children*
$(2, 4, 5, 6, 7)$	$(2^*, 4, 5^*, 6^*, 7)$	$(4, \underline{7}, \underline{2}^*, \underline{5}^*, \underline{6}^*)$	$(4, 3, 2, 5, 6)$	$(2, 3, 4, 5, 6)$
$(1, 2, 3, 5, 6)$	$(1, 2^*, 3, 5^*, 6^*)$	$(1, \underline{3}, \underline{2}^*, \underline{5}^*, \underline{6}^*)$	$(1, 7, 2, 5, 6)$	$(1, 2, 5, 6, 7)$

A mutation of the feasible solution $(2, 4, 5, 6, 7)$ *can be originated by the interchange between any index of this individual and an index of the set* $\{1, 3\}$*. Then, the indices of the new individual are sorted.*

A constant probability for all individuals along all iterations is associated to each genetic operator (crossover and mutation). The determination of this probability is based on empirical results, as shown in Subsection 3.2.6.

After the generation of the children, a selection criterion is applied to mimic the natural selection in genetics. This selection depends on the evaluation function ("fitness") presented in Subsection 3.2.3. Our selection criterion is described in the following subsection.

3.2.5 Selection Criterion

Our first idea was to use the same selection criterion as Moreno Vega [MV96]. This criterion is based on a probability function associated to each individual and depends on its "fitness". But empirically we could observe that by simply taking the H best individuals (i.e. those with minimal objective function value) at each iteration, the results were even better.

In our algorithm we use evolution strategies in order to ensure a kind of convergence in each generation, i.e. to avoid that the new population be worse than the original one,

see Bäck et al. [BHS91] and Schwefel [Sch81]. Hence, we include in the original population all the children generated by crossover and all the mutated individuals. Obviously, the number of individuals in the population after these transformations is normally larger than H. Thus, the selection criterion consists of dropping the worst individuals (i.e. those individuals with the largest objective function values) until the population contains again exactly H individuals. Clearly, this selection criterion ensures that the population size is constant at each iteration.

This method of replacing the population is called *incremental replacement*, since the child solutions will replace "less fit" members of the population, see Jaramillo et al. [JBB02]. Figure 1 illustrates one of the advantages of this method. After a few generations (100), we obtain a population containing a set of different good solutions all of them at most a modest percentage away from the best solution (1.79%). Figure 1 shows the best and the worst solution found at each generation as well as the optimal solution.

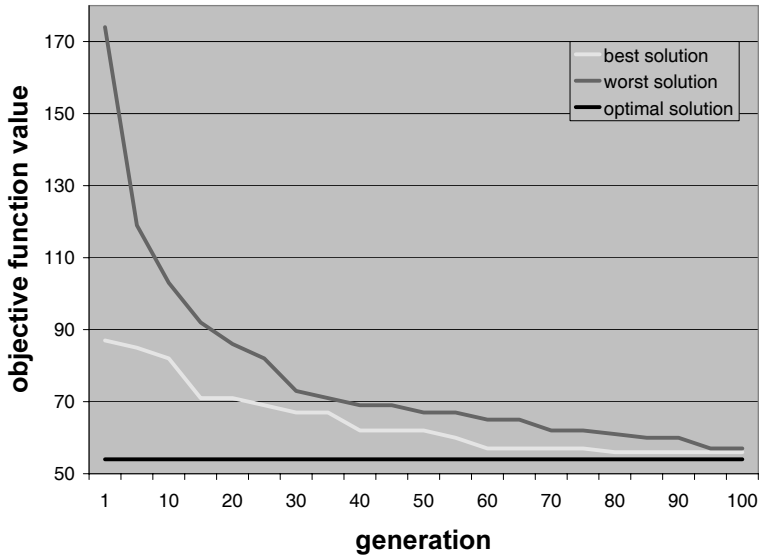

Figure 1: Evolution of the population

The behaviour of a genetic algorithm depends very much on the choice of a number of parameter values, as will be shown in the following subsection.

3.2.6 Parameter Values

One of the most difficult tasks in order to obtain an efficient genetic algorithm is the determination of good parameters. In our case, some of them were chosen a priori such as the population size, $H = 25$, and the total number of iterations, 1000.

We now describe the experiments performed in order to set the probabilities of crossover and mutation operators for the individuals. We distinguish two cases depending on the selected initial population, i.e. considering or not a greedy solution of the DOMP, see Section 3.2.2. To be able to compare the different values of probabilities, we always used the

same seed for the random number generator. In this way, the solutions of the different problems depend only on the values of the parameters but not on the random character.

We considered two instances with $m = 30$ and $p = 10$ and solved them for eight different types of Λ (in total 16 examples were tested).

- T1: $\Lambda = (1, \ldots, 1)$, vector corresponding to the p-median problem.
- T2: $\Lambda = (0, \ldots, 0, 1)$, vector corresponding to the p-center problem.
- T3: $\Lambda = (0, \ldots, 0, \underbrace{1, \ldots, 1}_{k})$, vector corresponding to the k-centra problem, where $k = \lfloor \frac{m}{3} \rfloor = 10$.
- T4: $\Lambda = (\underbrace{0, \ldots, 0}_{k_1}, 1, \ldots, 1, \underbrace{0, \ldots, 0}_{k_2})$, vector corresponding to the $k_1 + k_2$-trimmed mean problem, where $k_1 = p + \lceil \frac{m}{10} \rceil = 13$ and $k_2 = \lceil \frac{m}{10} \rceil = 3$.
- T5: $\Lambda = (0, 1, 0, 1, \ldots, 0, 1, 0, 1)$.
- T6: $\Lambda = (1, 0, 1, 0, \ldots, 1, 0, 1, 0)$.
- T7: $\Lambda = (0, 1, 1, 0, 1, 1, \ldots, 0, 1, 1, 0, 1, 1)$.
- T8: $\Lambda = (0, 0, 1, 0, 0, 1, \ldots, 0, 0, 1, 0, 0, 1)$.

Based on preliminary tests, we decided to select different values for the probabilities of mutation (0.05, 0.075, 0.1, 0.125) and crossover (0.1, 0.3, 0.5). Then we compared the gap between the optimal solution and that one given by the genetic algorithm, initialized either by *random* $P(0)$ or *greedy* $P(0)$:

$$\text{gap} = \frac{z_{\text{heu}} - z^*}{z^*} \times 100, \tag{2}$$

with z^* denoting the optimal objective function value and z_{heu} denoting the objective function value of the best solution obtained by the evolution program. The optimal solutions were determined using the exact procedure based on a branch-and-bound (B&B) method presented in Boland et al. [BDMNP03]. Table 1 summarizes the computational results obtained for the different probabilities.

		EP *random* $P(0)$			EP *greedy* $P(0)$		
probabilities		gap(%)			gap(%)		
mut	*cross*	aver	min	max	aver	min	max
0.05	0.1	5.05	0.00	25.00	4.50	0.00	22.22
0.05	0.3	3.35	0.00	22.00	6.06	0.00	25.00
0.05	0.5	7.98	0.00	66.67	11.63	0.00	100.00
0.075	0.1	1.85	0.00	11.11	3.89	0.00	12.50
0.075	0.3	2.54	0.00	16.89	4.66	0.00	12.50
0.075	0.5	4.27	0.00	25.00	5.14	0.00	22.22
0.1	0.1	1.49	0.00	12.50	3.11	0.00	10.71
0.1	0.3	6.15	0.00	22.22	5.14	0.00	25.00
0.1	0.5	4.28	0.00	22.22	4.36	0.00	12.50
0.125	**0.1**	**1.49**	**0.00**	**12.50**	**2.89**	**0.00**	**10.71**
0.125	0.3	5.31	0.00	25.00	6.06	0.00	25.00
0.125	0.5	6.32	0.00	25.00	4.33	0.00	12.50

Table 1: Computational results obtained by using different values for the probabilities of mutation and crossover.

Based on the results presented in Table 1, we decided to fix the probability of mutation equal to 0.125 and the crossover probability equal to 0.1 for the EP initialized by both types of populations (*random* $P(0)$ and *greedy* $P(0)$). Note that even though the averaged performance of *random* $P(0)$, for the selected probability values, is better than that of *greedy* $P(0)$ (see corresponding row in Table 1), there are cases for which *greedy* $P(0)$ yields much better results (as Table 2 shows). That is, the behaviour of both procedures can be seen as complementary.

	Problem Type	optimal value	EP *random* $P(0)$ best found	EP *random* $P(0)$ gap (%)	EP *greedy* $P(0)$ best found	EP *greedy* $P(0)$ gap (%)
example 1	T1	81	81	0.00	88	8.64
	T2	9	9	0.00	9	0.00
	T3	61	61	0.00	61	0.00
	T4	44	44	0.00	47	6.82
	T5	43	43	0.00	46	6.98
	T6	36	36	0.00	37	2.78
	T7	56	56	0.00	62	10.71
	T8	29	29	0.00	32	10.34
example 2	T1	78	78	0.00	78	0.00
	T2	8	9	12.50	8	0.00
	T3	54	54	0.00	54	0.00
	T4	46	46	0.00	46	0.00
	T5	41	42	2.44	41	0.00
	T6	36	36	0.00	36	0.00
	T7	55	58	5.45	55	0.00
	T8	28	29	3.57	28	0.00

Table 2: Computational results corresponding to the EP initialized either by *random* $P(0)$ or *greedy* $P(0)$.

Therefore, from the computational results given by Table 2, we propose to solve the DOMP by running the EP twice, once initialized by *random* $P(0)$ and once more initialized by *greedy* $P(0)$, and taking the best solution found. An extensive numerical study is presented in Section 5.

In the following section a second heuristic algorithm to solve the DOMP is developed.

4 A Variable Neighborhood Search for the DOMP

Our second heuristic procedure to solve the DOMP is based on the Variable Neighborhood Search (VNS) proposed by Hansen & Mladenović [HM97] for the p-median problem. As mentioned above p-median is a particular case of DOMP. Moreover, their objective functions are often close, which was the main reason to adapt the approach of [HM97] to our problem. However, computation of the objective function value is much harder for DOMP than for p-median. Indeed, a major difficulty is to compute the variation between the objective function values when an interchange between two facilities is performed. We are forced to update and sort the whole cost vector after this interchange takes place. For the p-median problem updating the value of the objective function can be done step by step. As a consequence, the complexity of our procedure is significantly higher.

In the following section we present a modified fast interchange algorithm, which is essential to describe the VNS developed to solve the DOMP.

4.1 An Implementation of the Modified Fast Interchange Heuristic

In this section we present an implementation of the basic move of many heuristics, i.e. an interchange (or a change of location for one facility). This procedure is based on the fast interchange heuristic proposed by Whitaker [Whi83] and implemented by, among others, Hansen & Mladenović [HM97] for the p-median problem. Two ingredients are incorporated in the interchange heuristic: *move evaluation*, where a best removal of a facility is found when the facility to be added is known; and *updating* the first and the second closest facility for each client.

Moreover, the variation of the ordered objective function value is computed after each interchange in *move*.

Thus using this interchange only from a random initial solution gives a fairly good heuristic. Results are even better with an initial solution obtained with the Greedy Algorithm.

In the description of the heuristic we use the following notation:

- $d1(i)$: index of the closest facility with respect to client i, for each $i = 1, \ldots, m$;
- $d2(i)$: index of the second closest facility with respect to client i, for each $i = 1, \ldots, m$;
- $c(i, j)$: cost of satisfying the total demand of client i from facility j, for each $i, j = 1, \ldots, m$;
- $x_{cur}(i)$ for each $i = 1, \ldots, p$: current solution (new facilities);
- $cost_{cur}$: current cost vector;
- f_{cur}: current objective function value;
- $goin$: index of facility to be inserted in the current solution;
- $goout$: index of facility to be deleted from the current solution;
- g^*: change in the objective function value obtained by the best interchange;

In the following four subsections we describe components of our second heuristic for DOMP.

4.1.1 Initial Solution

Our heuristic is initialized with a solution constructed with the Greedy Algorithm, as done for the EP in Section 3.2.2.

The move evaluation is presented in the following section.

4.1.2 Move Evaluation

In the next procedure called *Modified Move*, the change in the objective function g^* is evaluated when the facility that is added (denoted by *goin*) to the current solution is known, while the best one to go out (denoted by *goout*) is to be found.

Algorithm Modified Move $(d1, d2, C, \Lambda, x_{cur}, cost_{cur}, f_{cur}, goin, m, p, g^*, goout^*)$

Initialization

Set $g^* \leftarrow \infty$

Best deletion

for $goout = x_{cur}(1)$ to $x_{cur}(p)$ **do**
  Set $cost_{new} \leftarrow cost_{cur}$
  for each client i $(i = 1, \ldots, m)$ **do**
    if $d1(i) = goout$ **then**
      $cost_{new}(i) \leftarrow \min\{c(i, goin), c(i, d2(i))\}$
    else
      if $c(i, goin) < c(i, d1(i))$ **then**
        $cost_{new}(i) \leftarrow c(i, goin)$
      end if
    end if
  end for
  Find the corresponding objective function value f_{new}
  $g \leftarrow f_{new} - f_{cur}$
  if $g < g^*$ **then**
    $g^* \leftarrow g$ and $goout^* \leftarrow goout$
  end if
end for

Using algorithm *Modified Move*, each potential facility belonging to the current solution can be removed, i.e. be the facility *goout*. Furthermore, for each site we have to compute the objective function value corresponding to the new current solution for which facility *goout* is deleted and *goin* is inserted. Therefore, a new cost vector has to be sorted, as shown in Section 2, which leads to a complexity of $O(m \log m)$ for each of the p values of *goout*. Thus, the number of operations needed for this algorithm is $O(pm \log m)$.

4.1.3 Updating First and Second Closest Facilities

In the *Modified Move* procedure both the closest (denoted by $d1(i)$) and the second closest facility (denoted by $d2(i)$) for each client i must be known in advance. Among formal variables in the description of algorithm *Modified Update* that follows, arrays $d1$ and $d2$ are both input and output variables. In this way, for each site i, if either $d1(i)$ or $d2(i)$ is removed from the current solution, we update their values. Furthermore, the current cost vector is also updated, being an input and an output variable too. This is what distinguishes this procedure from the update approach presented in [HM97].

Algorithm Modified Update $(C, goin, goout, m, p, d1, d2, cost)$
for each site i $(i = 1, \ldots, m)$ **do**
(* For clients whose closest facility is deleted, find a new one *)
if $d1(i) = goout$ **then**
if $c(i, goin) \leq c(i, d2(i))$ **then**
$d1(i) \leftarrow goin$
$cost(i) \leftarrow c(i, goin)$
else
$d1(i) \leftarrow d2(i)$;
$cost(i) \leftarrow c(i, d2(i))$;
(* Find second closest facility for client i *)
find l^* where $c(i, l)$ is minimum (for $l = 1, \ldots, p,\ l \neq d1(i)$);
$d2(i) \leftarrow l^*$
end if
else
if $c(i, d1(i)) > c(i, goin)$ **then**
$d2(i) \leftarrow d1(i)$ and $d1(i) \leftarrow goin$
$cost(i) \leftarrow c(i, goin)$
else
if $c(i, goin) < c(i, d2(i))$ **then**
$d2(i) \leftarrow goin$
else
if $d2(i) = goout$ **then**
find l^* where $c(i, l)$ is minimum (for $l = 1, \ldots, p,\ l \neq d1(i)$);
$d2(i) \leftarrow l^*$
end if
end if
end if
end if
end for

The worst case complexity of the procedure *Modified Update* is $O(m(p-1)) = O(mp)$ as the index $d2(i)$ of the second closest facility must be recomputed without any additional information if it changes.

4.1.4 Modified Fast Interchange Heuristic

Our modified fast interchange algorithm, that uses procedures *Modified Move* and *Modified Update* described before, is as follows:

Initialization

Let x_{opt} be an initial solution;
find the corresponding cost vector $cost_{opt}$ and objective function value f_{opt};
find closest and second closest facilities for each client $i = 1, \ldots, m$, i.e. find arrays $d1$ and $d2$;

Iteration step

Set $g^* \leftarrow \infty$
for $goin = x_{opt}(p+1)$ to $x_{opt}(m)$ **do**
(* Add facility $goin$ in the solution and find the best deletion *)
Run procedure
Modified Move $(d1, d2, C, \Lambda, x_{opt}, cost_{opt}, f_{opt}, goin, m, p, g, goout)$;
(* Keep the best pair of facilities to be interchanged *)
if $g < g^*$ **then**
$g^* \leftarrow g$, $goin^* \leftarrow goin$, $goout^* \leftarrow goout$
end if
end for

Termination

if $g^* \geq 0$ **then**
Stop (* If no improvement in the neighborhood, Stop *)
end if

Updating step

(* Update objective function value *)
$f_{opt} \leftarrow f_{opt} + g^*$
Update x_{opt}: interchange position of $x_{opt}(goout^*)$ with $x_{opt}(goin^*)$
(* Update closest, second closest facilities and cost vector *)
Run procedure *Modified Update* $(C, goin^*, goout^*, m, p, d1, d2, cost_{opt})$;
Return to *Iteration step*

The complexity of one iteration of this algorithm is $O((m-p)pm \log m) = O(pm^2 \log m)$. This follows from the fact that procedure *Modified Move* is used $m-p$ times, its complexity is $O(pm \log m)$ and the complexity of *Modified Update* is $O(mp)$.

In the following section we present a heuristic based on VNS that solves the DOMP using our modified fast interchange algorithm.

4.2 Variable Neighborhood Search

The basic idea of VNS is to implement a systematic change of neighborhood within a local search algorithm, see Hansen & Mladenović [HM01a], [HM01b] and [HM01c]. Exploration of these neighborhoods can be done in two ways. The first one consists of systematically exploring the smallest neighborhoods, i.e. those closest to the current solution, until a better

solution is found. The second one consists in partially exploring the largest neighborhoods, i.e. those far from the current solution, by drawing a solution at random from them and beginning a (variable neighborhood) local search from there. The algorithm remains in the same solution until a better solution is found and then jumps there. We rank the neighborhoods to be explored in such a way that they are increasingly far from the current solution. We may view VNS as a way of escaping local optima, i.e. a "shaking" process, where movement to a neighborhood further from the current solution corresponds to a harder shake. In contrast to random restart, VNS allows a controlled increase in the level of the shake.

As in Hansen & Mladenović [HM97] let us denote by $S = \{s : s = \text{set of } p \text{ potential locations of the new facilities}\}$ a solution space of the problem. We say that the distance between two potential solutions s_1 and s_2 ($s_1, s_2 \in S$) is equal to k, if and only if they differ in k locations. Since S is a set of sets of equal cardinality, a (symmetric) distance function ρ can be defined as

$$\rho(s_1, s_2) = |s_1 \setminus s_2| = |s_2 \setminus s_1|, \forall s_1, s_2 \in S. \tag{3}$$

It can easily be checked that ρ is a metric function in S, thus S is a metric space. As in [HM97], the neighborhoods structures are induced by metric ρ, i.e. k locations of facilities ($k \leq p$) from the current solution are replaced by k others. We denote by $\mathcal{N}_k$, $k = 1, \ldots, k_{max}$ ($k_{max} \leq p$) the set of such neighborhood structures and by $\mathcal{N}_k(s)$ the set of solutions forming neighborhood $\mathcal{N}_k$ of a current solution s. More formally

$$s_1 \in \mathcal{N}_k(s_2) \Leftrightarrow \rho(s_1, s_2) = k. \tag{4}$$

Note that the cardinality of $\mathcal{N}_k(s)$ is

$$|\mathcal{N}_k(s)| = \binom{p}{k}\binom{m-p}{k}$$

since k out of p facilities are dropped and k out of $m - p$ added into the solution. This number first increases then decreases with k.

Note also that sets $\mathcal{N}_k(s)$ are disjoint, and their union, together with s, is S.

We now present our VNS algorithm for the DOMP as pseudo-code:

Initialization

Find arrays x_{opt}, $d1$ and $d2$, $cost_{opt}$ and f_{opt} as initialization of *Modified Fast Interchange*;
the set of neighborhood structures $\mathcal{N}_k$, $k = 1, \ldots, k_{max}$ is induced by distance function ρ (see (3) and (4));
copy initial solution into the current one, i.e. copy x_{opt}, $d1$, $d2$, $cost_{opt}$ and f_{opt} into x_{cur}, $d1_{cur}$, $d2_{cur}$, $cost_{cur}$ and f_{cur}, respectively.
Choose stopping condition

Main step

$k \leftarrow 1$;
Until $(k = k_{max})$ or (stopping condition is met), repeat the following steps:

Shaking operator

(* Generate a solution at random from the kth neighborhood, $\mathcal{N}_k$ *)
for $j = 1$ to k **do**
Take facility to be inserted *goin* at random;
Find facility to be deleted *goout* by using procedure
Modified Move $(d1, d2, C, \Lambda, x_{cur}, cost_{cur}, f_{cur}, goin, m, p, g, goout)$;
Find $d1_{cur}$, $d2_{cur}$ and $cost_{cur}$ for such interchange, i.e. run subroutine
Modified Update$(C, goin^*, goout^*, m, p, d1_{cur}, d2_{cur}, cost_{cur})$;
Update x_{cur} and f_{cur} accordingly;
end for

Local Search

Apply algorithm *Modified Fast Interchange* (without *Initialization* step), with x_{cur}, f_{cur}, $cost_{cur}$, $d1_{cur}$ and $d2_{cur}$ as input and output values;

Move or not

if $f_{cur} < f_{opt}$ **then**
(* Save current solution to be incumbent; return to $\mathcal{N}_1$ *)
$f_{opt} \leftarrow f_{cur}$; $x_{opt} \leftarrow x_{cur}$; $d1 \leftarrow d1_{cur}$; $d2 \leftarrow d2_{cur}$; $cost_{opt} \leftarrow cost_{cur}$; and set $k \leftarrow 1$;
else
(* Current solution is the incumbent; change the neighborhood *)
$f_{cur} \leftarrow f_{opt}$; $x_{cur} \leftarrow x_{opt}$; $d1_{cur} \leftarrow d1$; $d2_{cur} \leftarrow d2$; $cost_{cur} \leftarrow cost_{opt}$; and set $k \leftarrow k + 1$;
end if

In *Shaking operator* step the incumbent solution x_{opt} is perturbed in such a way that $\rho(x_{cur}, x_{opt}) = k$. Nevertheless this step does not guarantee that x_{cur} belongs to $\mathcal{N}_k(x_{opt})$ due to randomize of the choice of *goin* and possible reinsertion of the same facility after it has left. Then x_{cur} is used as initial solution for *Modified Fast Interchange* in *Local Search* step. If a better solution than x_{opt} is obtained, we move there and start again with small

perturbations of this new best solution, i.e. $k \leftarrow 1$. Otherwise, we increase the distance between x_{opt} and the new randomly generated point, i.e. we set $k \leftarrow k + 1$. If k reaches k_{max} (this parameter can be chosen equal to p), we return to *Main* step, i.e. the main step can be iterated until some other stopping condition is met (e.g. maximum number of iterations, maximum CPU time allowed, or maximum number of iterations between two improvements). Note that the point x_{cur} is generated at random in *Shaking operator* step in order to avoid cycling which might occur if any deterministic rule were used.

In the following section computational results are reported which show the efficiency of these two heuristic approaches.

5 Computational Results

In order to test our heuristic procedures we considered two groups of experiments. The instances belonging to the first group have been randomly generated with different combinations of the number of existing facilities, the number of new facilities, and the Λ-vectors. The second group of experiments consists of p-median problems (whose optimal solutions are provided by Beasley [Bea85]) and $k_1 + k_2$-trimmed mean problems using the data publicly available electronically from *http://mscmga.ms.ic.ac.uk/info.html*, see Beasley [Bea90]. The first group of experiments allows investigating the behaviour of our heuristic approaches with different types of Λ. The second one helps in determining their capability to solve large problems.

In the following sections we describe in an exhaustive way these two groups of experiments and the corresponding computational results. All test problems were solved using a Pentium III 800 Mhz with 1 GB RAM.

5.1 Numerical experiments with different parameter combinations

The first group of experimental data was designed considering four different values for the number of sites, $m = 15, 18, 25, 30$, four values for the number of new facilities, $p = \lceil \frac{m}{4} \rceil, \lceil \frac{m}{3} \rceil, \lceil \frac{m}{2} \rceil, \lceil \frac{m}{2} \rceil + 1$, and eight different Λ-vectors, see Section 3.2.6. In total, 1920 problems were solved by both heuristic approaches (4 different values of m $\times$ 4 values of p $\times$ 8 values of Λ $\times$ 15 instances randomly generated for each combination of m, p and Λ).

As mentioned in Subsection 3.2.6, we ran the evolution program twice (with *random* $P(0)$ and *greedy* $P(0)$). The solution of the evolution program is the best obtained by both procedures. Obviously, computation time increases with size but nevertheless it is worthwhile due to the difficulty of solving the problem with a B&B method (see Boland et al. [BDMNP03]) and the quality of the solution obtained.

In order to compare the solutions given by the EP and the VNS with the optimal ones, the instances known from the literature have been used. These problems are of small size ($m = 15, 18, 25, 30$). The gap between the optimal solution and that one obtained by each heuristic algorithm is computed according to (2) with z^* denoting the optimal objective

function value and z_{heu} denoting the objective function value of the best solution provided by the heuristic procedure.

Tables 3 and 4 show computational results for instances with $m = 30$, given by the EP and the VNS, respectively. In each row we present a summary of the outcomes for 15 replications of each combination (Λ, p). Each row reports information about the frequency that the optimal solution is reached, gap between the optimal solution and that provided by the corresponding heuristic approach and computing time.

		Evolution program							B&B		
Example		#opt.	gap (%)			CPU(s)			CPU(s)		
Λ	p	found	aver	min	max	aver	min	max	aver	min	max
T1	8	11	2.45	0.00	21.78	22.57	22.30	22.88	167.19	40.92	408.73
	10	10	1.34	0.00	7.04	25.42	25.09	26.41	303.11	21.75	762.08
	15	11	0.87	0.00	5.00	32.25	31.97	32.67	274.74	28.03	562.25
	16	14	0.25	0.00	3.70	33.71	33.48	34.25	198.65	41.19	417.09
T2	8	10	7.79	0.00	55.56	22.46	22.03	22.73	13.54	2.14	44.19
	10	10	7.08	0.00	42.86	25.19	24.91	25.50	6.39	1.06	16.59
	15	12	4.44	0.00	25.00	32.16	31.81	32.45	4.81	0.00	14.84
	16	12	6.11	0.00	33.33	33.74	33.38	34.08	5.03	0.09	14.23
T3	8	11	1.09	0.00	7.92	22.35	22.16	22.69	86.44	39.11	197.08
	10	12	0.33	0.00	1.85	25.06	24.92	25.16	117.02	25.20	370.39
	15	11	0.88	0.00	3.70	31.99	31.72	32.27	125.19	15.36	246.95
	16	15	0.00	0.00	0.00	33.43	33.13	33.77	99.51	21.83	217.92
T4	8	12	0.37	0.00	2.47	22.36	22.14	22.72	56.37	5.06	125.25
	10	11	0.94	0.00	8.11	25.13	24.91	25.33	154.46	2.31	476.23
	15	15	0.00	0.00	0.00	32.04	31.80	32.22	649.56	20.81	1457.81
	16	15	0.00	0.00	0.00	33.43	33.22	33.78	498.32	42.88	1460.81
T5	8	11	1.00	0.00	5.88	22.41	22.17	22.73	160.75	45.42	332.02
	10	9	1.52	0.00	5.26	25.22	24.84	25.75	242.00	25.72	601.64
	15	14	0.83	0.00	12.50	32.20	31.63	32.81	209.68	15.55	502.16
	16	12	1.66	0.00	10.00	33.60	33.05	34.33	106.49	18.88	239.69
T6	8	10	1.30	0.00	9.52	22.54	22.25	22.83	131.82	15.64	272.22
	10	9	1.36	0.00	5.41	25.40	25.09	25.97	253.11	12.58	561.84
	15	14	0.39	0.00	5.88	32.38	31.97	32.91	327.44	26.34	777.64
	16	14	0.56	0.00	8.33	33.75	33.36	34.28	311.89	19.88	792.27
T7	8	8	1.66	0.00	4.94	22.53	22.33	22.72	170.57	40.69	368.23
	10	13	0.26	0.00	1.96	25.38	25.06	25.70	269.65	15.44	712.63
	15	12	1.08	0.00	8.00	32.41	32.02	33.41	179.30	12.47	442.06
	16	15	0.00	0.00	0.00	33.89	33.59	34.25	142.95	29.02	313.39
T8	8	10	2.83	0.00	21.05	22.60	22.33	23.30	151.38	38.84	328.09
	10	9	1.98	0.00	11.54	25.38	25.03	25.80	188.97	11.33	503.84
	15	14	0.44	0.00	6.67	32.39	31.98	32.69	101.19	0.91	436.17
	16	14	0.61	0.00	9.09	33.87	33.44	34.19	76.59	18.30	173.09

cont.

Example		Evolution program							B&B		
		#opt.	gap (%)			CPU(s)			CPU(s)		
Λ	p	found	aver	min	max	aver	min	max	aver	min	max

Table 3: Computational results obtained for instances with $m = 30$ using the EP.

From Table 3 we can observe that the average gap over all instances is 1.61%. Moreover, in many cases, the optimal objective function value is reached, even for problems of type T2 (for which at least 60% of the 15 instances for each parameter combination were solved optimally). In average, the optimal solution is reached in 79.17% of the instances with $m = 30$. In addition, the required computing time is, in general (except for problems of type T2), shorter than that needed by the exact procedure. Observe that the average computing time required by the EP is 28.41 seconds, much shorter than 180.75 seconds given by the specific B&B method for the same instances.

Example		Var. Neigh. Search							B&B		
		#opt.	gap (%)			CPU(s)			CPU(s)		
Λ	p	found	aver	min	max	aver	min	max	aver	min	max
T1	8	10	2.13	0.00	11.86	0.31	0.20	0.47	167.19	40.92	408.73
	10	14	0.08	0.00	1.20	0.44	0.31	0.69	303.11	21.75	762.08
	15	13	0.39	0.00	3.13	0.71	0.52	0.86	274.74	28.03	562.25
	16	15	0.00	0.00	0.00	0.79	0.64	1.36	198.65	41.19	417.09
T2	8	5	14.74	0.00	55.56	0.30	0.11	0.50	13.54	2.14	44.19
	10	10	8.89	0.00	71.43	0.43	0.20	1.00	6.39	1.06	16.59
	15	12	5.78	0.00	50.00	0.60	0.38	1.05	4.81	0.00	14.84
	16	11	9.44	0.00	50.00	0.56	0.39	0.91	5.03	0.09	14.23
T3	8	9	2.61	0.00	11.29	0.27	0.22	0.41	86.44	39.11	197.08
	10	10	1.62	0.00	9.80	0.44	0.31	0.67	117.02	25.20	370.39
	15	14	0.19	0.00	2.78	0.76	0.52	1.06	125.19	15.36	246.95
	16	15	0.00	0.00	0.00	0.76	0.63	1.11	99.51	21.83	217.92
T4	8	13	0.28	0.00	2.47	0.27	0.19	0.38	56.37	5.06	125.25
	10	13	1.07	0.00	8.11	0.40	0.25	0.61	154.46	2.31	476.23
	15	15	0.00	0.00	0.00	0.60	0.42	0.78	649.56	20.81	1457.81
	16	15	0.00	0.00	0.00	0.62	0.45	0.78	498.32	42.88	1460.81
T5	8	8	2.61	0.00	8.00	0.29	0.20	0.47	160.75	45.42	332.02
	10	13	0.50	0.00	5.26	0.46	0.33	0.80	242.00	25.72	601.64
	15	15	0.00	0.00	0.00	0.64	0.48	0.73	209.68	15.55	502.16
	16	15	0.00	0.00	0.00	0.75	0.64	0.89	106.49	18.88	239.69

cont.

Example		#opt.	Var. Neigh. Search gap (%)			Var. Neigh. Search CPU(s)			B&B CPU(s)		
Λ	p	found	aver	min	max	aver	min	max	aver	min	max
T6	8	13	0.60	0.00	6.67	0.32	0.19	0.56	131.82	15.64	272.22
	10	13	0.56	0.00	5.26	0.41	0.31	0.69	253.11	12.58	561.84
	15	15	0.00	0.00	0.00	0.70	0.47	1.17	327.44	26.34	777.64
	16	14	0.56	0.00	8.33	0.70	0.48	1.20	311.89	19.88	792.27
T7	8	9	1.50	0.00	7.46	0.26	0.20	0.41	170.57	40.69	368.23
	10	14	0.11	0.00	1.67	0.41	0.31	0.58	269.65	15.44	712.63
	15	14	0.32	0.00	4.76	0.73	0.50	1.11	179.30	12.47	442.06
	16	15	0.00	0.00	0.00	0.75	0.53	1.14	142.95	29.02	313.39
T8	8	10	1.46	0.00	7.69	0.28	0.19	0.41	151.38	38.84	328.09
	10	11	1.27	0.00	7.69	0.44	0.31	0.72	188.97	11.33	503.84
	15	15	0.00	0.00	0.00	0.73	0.45	1.06	101.19	0.91	436.17
	16	13	1.96	0.00	22.22	0.65	0.44	0.89	76.59	18.30	173.09

Table 4: Computational results obtained for instances with $m = 30$ using the VNS.

From Table 4 we can notice that the average gap provided by the VNS, 1.83%, over all instances is slightly higher than that given by the EP, 1.61%. Nevertheless, the optimal objective function is reached in 83.54% of the instances. In addition, the average computing time, 0.52 seconds, is much shorter than that required by the EP (28.41 seconds), and therefore, shorter than that given by the B&B algorithm (180.75 seconds).

It should be mentioned that the performance of both procedures on problems of type T2 (i.e. p-center problems) is rather poor, since the gap obtained is relatively large. However, the quality of the solution given by the EP for problems of type T2 is superior than that provided by the VNS. We point out that the new formulation proposed by Elloumi et al. [ELP01] specifically developed for the p-center problem yields results considerably better than EP.

To compare both heuristic procedures on examples close to real-life problems, instances of larger size ($m = 100$) have been generated and solved. The structure of these examples is similar to that presented for instances with $m = 30$, i.e. with four different values of p, eight of Λ, and 15 replications for each combination (p, Λ). The optimal solutions of these instances are not available, therefore, we can only compare the results given by both heuristic approaches. To this aim, we compute the relation between the solution given by the EP and that provided by the VNS, as follows

$$\text{ratio} = \frac{z_{\text{EP}}}{z_{\text{VNS}}}, \tag{5}$$

with z_{EP} denoting the objective function value of the best solution obtained by the evolution program and z_{VNS} denoting the objective function value of the solution obtained by the

variable neighborhood search. Therefore,

$$\text{ratio} \begin{cases} > 1 & \text{if VNS provides a solution better than that given by EP} \\ = 1 & \text{if VNS and EP provide the same solution} \\ < 1 & \text{if VNS provides a solution worse than that given by EP} \end{cases} \quad (6)$$

Figure 2 shows a summary of the results obtained among the 480 test problems.

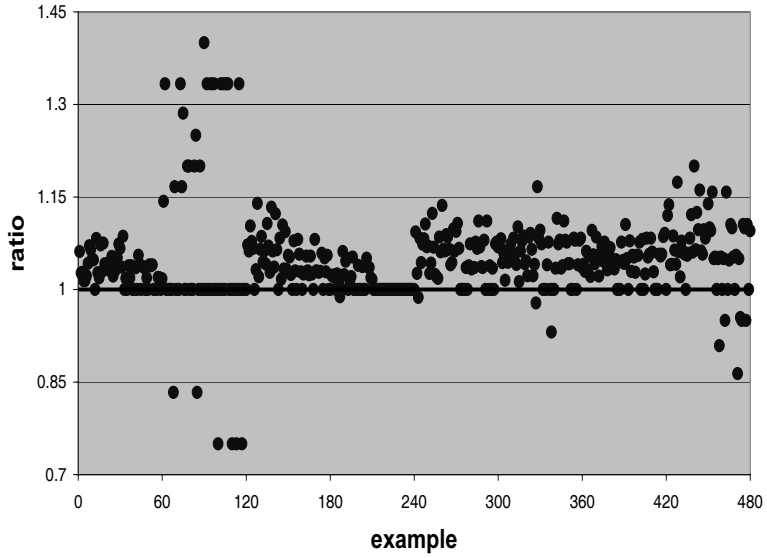

Figure 2: Comparison between the solutions given by EP and VNS for problems with $m = 100$.

From Figure 2 we can conclude that the quality of the solution provided by the VNS is usually superior to that given by the EP. Furthermore, the computing time required by VNS (63.22 seconds, in average) is also shorter than that required by EP (105.45 seconds, in average).

In the following section we investigate the behaviour of both heuristic approaches for solving large p-median and $k_1 + k_2$-trimmed mean problems.

5.2 Additional Tests for Large Problems

The exact procedures presented in the literature (Nickel [Nic01], Boland et al. [BDMNP03]) are not appropriate for solving large instances of the DOMP. Therefore, we call upon the existing data often used in the literature for p-median, see Beasley [Bea90]. In addition, these data have been used to solve the $k_1 + k_2$-trimmed mean problem. To this aim we have set $k_1 = p + \lceil \frac{m}{10} \rceil$ and $k_2 = \lceil \frac{m}{10} \rceil$.

The second group of experiments consists in solving p-median and $k_1 + k_2$-trimmed mean problems for 40 large instances. We denote these instances by *pmed1*,..., *pmed40*. The data are available from *http://mscmga.ms.ic.ac.uk/info.html*. Exhaustive information

about how these problems were generated can be found in Beasley [Bea85]. The number of existing facilities, m, in these instances varies from 100 to 900 in steps of 100, and the number of new facilities, p, takes values equal to 5, 10, $\frac{m}{10}$, $\frac{m}{5}$ and $\frac{m}{3}$ or, depending on the case, rounded to the nearest integer.

The optimal solutions of the p-median problems are given by Beasley [Bea85], but those according to the $k_1 + k_2$-trimmed mean problems are not available.

Therefore, using the first type of problems (p-median problems) we estimate the efficiency of our heuristic approaches comparing their results with the optimal solutions. The second type of problems allow pointing out the capability of these approaches to provide solutions for large instances of new facility location problems (such as the $k_1 + k_2$-trimmed mean problem) in a reasonable computing time.

5.2.1 Large p-Median Problems

For solving large p-median problems, as before, the EP is run twice (once initialized with *random* $P(0)$ and once more with *greedy* $P(0)$) and the best solution obtained by both procedures is taken.

Due to the large increase in computing time required by VNS, a stopping condition was necessary. After some preliminary tests, the maximum number of iterations allowed was fixed to 50.

Figure 3 shows the behaviour of the VNS against the required computing time (in seconds) on instance *pmed9* when the p-median problem is solved.

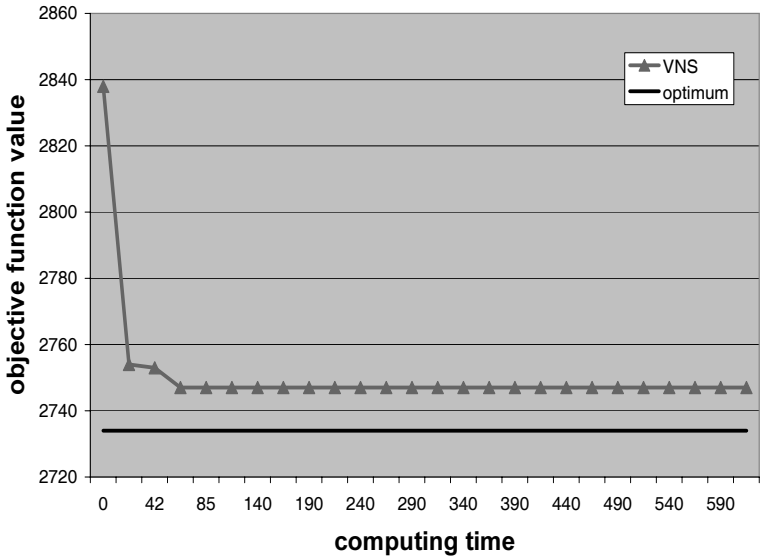

Figure 3: Behaviour of the VNS heuristic when solving the *pmed9* instance of the p-median problem.

From Figure 3 we conclude that convergence of the VNS is very fast. After 18 seconds (in the first iteration) the solution is much improved (from 2838 to 2754, i.e. reducing the

gap from 3.80% to 0.73%). Nevertheless, the VNS stops after 48 iterations, with a solution equal to 2747 (and a gap equal to 0.48%) but requiring almost 615 seconds.

We observed similar behaviour for other instances, which require much more time (some of them have been solved after fifty iterations requiring more than eight hours). For this reason, the maximum number of iterations for solving large p-median problems was fixed to 5 instead of 50.

Computational results for large p-median problems are shown in Table 5.

				Evolution Program			Var. Neigh. Search		
Problem			optimal	best	gap	CPU	best	gap	CPU
Name	m	p	value	found	(%)	(s)	found	(%)	(s)
pmed1	100	5	5819	5819	0.00	25.42	5819	0.00	1.19
pmed2		10	4093	4093	0.00	37.55	4093	0.00	2.97
pmed3		10	4250	4250	0.00	37.88	4250	0.00	3.00
pmed4		20	3034	3046	0.40	61.48	3046	0.40	5.98
pmed5		33	1355	1361	0.44	93.22	1358	0.22	6.81
pmed6	200	5	7824	7824	0.00	36.25	7824	0.00	7.95
pmed7		10	5631	5645	0.25	55.39	5639	0.14	12.72
pmed8		20	4445	4465	0.45	91.81	4457	0.27	21.05
pmed9		40	2734	2762	1.02	170.25	2753	0.69	41.98
pmed10		67	1255	1277	1.75	290.53	1259	0.32	72.22
pmed11	300	5	7696	7696	0.00	47.98	7696	0.00	12.52
pmed12		10	6634	6634	0.00	75.63	6634	0.00	26.02
pmed13		30	4374	4432	1.33	193.22	4374	0.00	87.92
pmed14		60	2968	2997	0.98	359.58	2969	0.03	241.95
pmed15		100	1729	1749	1.16	580.98	1739	0.58	363.39
pmed16	400	5	8162	8183	0.26	56.89	8162	0.00	24.36
pmed17		10	6999	6999	0.00	95.08	6999	0.00	47.30
pmed18		40	4809	4880	1.48	320.38	4811	0.04	275.69
pmed19		80	2845	2891	1.62	604.36	2864	0.67	469.30
pmed20		133	1789	1832	2.40	963.44	1790	0.06	915.17
pmed21	500	5	9138	9138	0.00	70.14	9138	0.00	27.39
pmed22		10	8579	8669	1.05	116.59	8669	1.05	64.25
pmed23		50	4619	4651	0.69	486.08	4619	0.00	443.23
pmed24		100	2961	3009	1.62	924.66	2967	0.20	1382.84
pmed25		167	1828	1890	3.39	1484.13	1841	0.71	2297.25
pmed26	600	5	9917	9919	0.02	84.34	9917	0.00	48.45
pmed27		10	8307	8330	0.28	136.53	8310	0.04	127.63
pmed28		60	4498	4573	1.67	673.30	4508	0.22	965.48
pmed29		120	3033	3099	2.18	1268.89	3036	0.10	2758.56
pmed30		200	1989	2036	2.36	2043.33	2009	1.01	3002.34

cont.

Problem Name	m	p	optimal value	Evolution Program best found	Evolution Program gap (%)	Evolution Program CPU (s)	Var. Neigh. Search best found	Var. Neigh. Search gap (%)	Var. Neigh. Search CPU (s)
pmed31	700	5	10086	10086	0.00	92.67	10086	0.00	56.02
pmed32		10	9297	9319	0.24	156.50	9301	0.04	165.27
pmed33		70	4700	4781	1.72	894.19	4705	0.11	2311.03
pmed34		140	3013	3100	2.89	1762.69	3024	0.37	5384.19
pmed35	800	5	10400	10400	0.00	109.86	10400	0.00	88.50
pmed36		10	9934	9947	0.13	182.06	9934	0.00	200.97
pmed37		80	5057	5126	1.36	1190.25	5066	0.18	2830.30
pmed38	900	5	11060	11060	0.00	120.14	11060	0.00	150.53
pmed39		10	9423	9423	0.00	207.75	9423	0.00	200.73
pmed40		90	5128	5188	1.17	1492.59	5141	0.25	4774.38

Table 5: Computational results for large p-median problems (Beasley [Bea85]).

By analysing this table we conclude that the average gap does not reach 1% for both heuristic procedures, being 0.86% for the EP and only 0.19% for the VNS. Moreover, the optimal objective function value is obtained 12 and 17 times, among the 40 test problems, for the EP and the VNS, respectively. Therefore, both approaches perform well on large p-median problems, the quality of the VNS being better. However, for both methods there is a tradeoff between quality of the solution and computing time required to obtain this solution: the average time is equal to 442.35 seconds for the EP and 747.97 seconds for the VNS. Notice that the maximal computing time required by the EP does not exceed 35 minutes, and that required by the VNS reaches almost 90 minutes.

Observe that the quality of the solutions provided by our VNS (0.19%, in average) is comparable with the one (0.18%, in average) given by the method specifically developed for p-median by Hansen et al. [HMPB01]. Nevertheless, as expected, computing time required by our VNS developed for the DOMP is larger than that provided in [HMPB01].

5.2.2 Large $k_1 + k_2$-Trimmed Mean Problems

As above, to solve large $k_1 + k_2$-trimmed mean problems, the best solution obtained after running twice the EP (once initialized with *random* $P(0)$ and once more with *greedy* $P(0)$) is taken.

Furthermore, as before, a stopping condition based on fixing a maximum number of iterations was considered. Again we could observe that the VNS converged very fast when solving $k_1 + k_2$-trimmed mean problems and again, computing time required for 50 iterations was too long. Therefore, the maximum number of iterations for solving large $k_1 + k_2$-trimmed mean problems was fixed to 5 instead to 50.

Computational results obtained for such problems are shown in Table 6. Since the optimal solutions are not available, this table reports information about the relation between both heuristic approaches (see (5)).

Problem Name	m	p	Evolution Program best found	Evolution Program CPU (s)	Var. Neigh. Search best found	Var. Neigh. Search CPU (s)	Ratio EP/ VNS
pmed1	100	5	4523	25.20	4523	1.27	1.000
pmed2		10	2993	36.98	2987	3.80	1.002
pmed3		10	3067	36.91	3074	2.80	0.998
pmed4		20	2153	60.80	2142	6.98	1.005
pmed5		33	829	92.08	818	8.22	1.013
pmed6	200	5	6064	35.52	6079	7.88	0.998
pmed7		10	4225	54.17	4206	13.41	1.005
pmed8		20	3248	91.95	3182	28.30	1.021
pmed9		40	1831	167.61	1816	66.39	1.008
pmed10		67	849	274.09	829	75.91	1.024
pmed11	300	5	5979	47.75	5979	13.30	1.000
pmed12		10	5021	73.83	5021	25.86	1.000
pmed13		30	3175	183.25	3133	97.80	1.013
pmed14		60	2027	346.42	1957	303.64	1.036
pmed15		100	1181	549.67	1133	415.80	1.042
pmed16	400	5	6341	56.06	6341	24.13	1.000
pmed17		10	5440	89.30	5413	43.83	1.005
pmed18		40	3463	309.50	3443	261.86	1.006
pmed19		80	1973	618.88	1933	779.77	1.021
pmed20		133	1191	1000.41	1152	1108.48	1.034
pmed21	500	5	7245	71.69	7245	24.22	1.000
pmed22		10	6749	117.88	6722	58.58	1.004
pmed23		50	3379	461.50	3306	639.95	1.022
pmed24		100	2068	888.27	2005	1455.81	1.031
pmed25		167	1198	1524.86	1151	2552.02	1.041
pmed26	600	5	7789	87.30	7787	48.11	1.000
pmed27		10	6481	141.97	6444	141.70	1.006
pmed28		60	3304	687.42	3210	1113.89	1.029
pmed29		120	2087	1249.78	2006	3178.69	1.040
pmed30		200	1359	1976.77	1308	4942.75	1.039
pmed31	700	5	8047	90.81	8046	66.16	1.000
pmed32		10	7318	148.77	7280	162.97	1.005
pmed33		70	3463	857.47	3413	2377.72	1.015
pmed34		140	2083	1624.61	2023	5657.56	1.030
pmed35	800	5	8191	102.58	8191	72.58	1.000
pmed36		10	7840	170.38	7820	201.64	1.003
pmed37		80	3684	1086.13	3604	3170.70	1.022
pmed38	900	5	8768	111.61	8720	140.84	1.006
pmed39		10	7398	189.19	7360	313.03	1.005
pmed40		90	3768	1372.98	3718	5422.73	1.013

cont.

Problem Name	m	p	Evolution Program best found	Evolution Program CPU (s)	Var. Neigh. Search best found	Var. Neigh. Search CPU (s)	Ratio EP/ VNS

Table 6: Computational results for large $k_1 + k_2$-trimmed mean problems (Beasley [Bea85]).

From Table 6 we can observe that after a reasonable computing time the EP and the VNS solve large $k_1 + k_2$-trimmed mean problems. The computing time required by the EP (427.81 seconds, in average) is much shorter than that needed by the VNS (875.78 seconds, in average). Nevertheless, in all but two cases the VNS provides a better solution than EP.

6 Conclusions and Further Research

This paper presents the first heuristic techniques, based on genetic algorithms and variable neighborhood search, to solve the DOMP. For problems with a relatively small size, the proposed heuristic approaches perform very well with respect to both computing time and solution quality. For large p-median problems the solution quality is good but the computing time highly increases. However, the results of both procedures are poor for p-center problems, already for small instances and the same may be expected for larger ones.

It should be stressed that our procedures are not intended to compete against approaches specifically developed for p-median and p-center problems. Obviously, such methods take advantage of the structure of these particular problems. Rather, we reach a good performance on more general types of problems.

The presented evolution program performed very well on the majority of the test problems. However, some parameter values such as the size of the population and the number of generations should be adjusted for large problems. Note that Alander [Ala92] suggests that a value between p and $2p$ should be used as population size, therefore a population of 25 individuals can be too small for large examples. Moreover, the *incremental replacement* method produces excessive copies of a solution, especially for large instances. Therefore, we should not allow too many duplicate solutions at each generation, since it limits the ability of the approach to generate new solutions, see Jaramillo et al. [JBB02]. Nevertheless, we could obtain good results at least when solving large p-median problems.

The computing time required by the variable neighborhood search for problems with large size increases considerably. Therefore, we propose to find new stopping conditions (e.g. maximum CPU time allowed, or maximum number of iterations between two improvements) in order to reduce the very large computing time without compromising too much the quality of the solution. Furthermore, different methods to determine the best interchange for a current solution without enumerating all the possibilities could be very interesting for further research.

Finally, it should be remarked that the DOMP generalizes classical facility location problems. Therefore, these heuristic approaches are able to provide solutions for problems not yet investigated in the literature (for example for $k_1 + k_2$-trimmed mean problems). We plan to investigate specialized heuristic procedures for some of these models as future research.

References

[Ala92] J.T. Alander. On optimal population size of genetic algorithms. In *Proceedings of CompEuro*, volume 92, pages 65–70. Silverspring, MD: IEEE Computer Society Press, 1992.

[BDMNP03] N. Boland, P. Domínguez-Marín, S. Nickel, and J. Puerto. Exact procedures for solving the Discrete Ordered Median Problem. ITWM Bericht 47, Fraunhofer Institut für Techno– und Wirtschaftsmathematik (ITWM), Kaiserslautern, Germany, 2003.

[Bea85] J.E. Beasley. Solving large p-median problems. *European Journal of Operational Research*, 21:270–273, 1985.

[Bea90] J.E. Beasley. Or-library: Distributing test problems by electronic mail. *Journal of the Operational Research Society*, 41(11):1069–1072, 1990.

[BHS91] T. Bäck, F. Hoffmeister, and H.P. Schwefel. A survey of evolution strategies. In R.K. Belew and L.B. Booker, editors, *Genetic Algorithms, Proceedings of the Fourth International Conference*, pages 2–9. Morgan Kaufmann, San Mateo, CA, 1991.

[Das95] M.S. Daskin. *Network and Discrete Location: Models, Algorithms, and Applications*. Wiley, 1995.

[Dav87] L. Davis, editor. *Genetic Algorithms and Simulated Annealing*. Morgan Kaufmann Publishers, San Mateo, CA, 1987.

[Dav91] L. Davis, editor. *Handbook of Genetic Algorithms*. Van Nostrand Reinhold, New York, 1991.

[Daw89] R. Dawkins. *The selfish gene*. Oxford University Press, second edition, 1989.

[DE02] Z. Drezner and H.W. Hamacher (Eds.). *Facility Location: Applications and Theory*. Springer Berlin Heidelberg New York, 2002.

[ELP01] S. Elloumi, M. Labbé, and Yves Pochet. New formulation and resolution method for the p-center problem. A preliminary version is available at http://smg.ulb.ac.be/Preprints/Elloumi01_03.html, 2001.

[FLT00] R.L. Francis, T.J. Lowe, and A. Tamir. Aggregation error bounds for a class of location models. *Operations Research*, 48:294–307, 2000.

[Gol89] D.E. Goldberg. *Genetic Algorithms in Search, Optimization and Machine Learning*. Addison Wesley, MA, 1989.

[HG86] C.M. Hosage and M.F. Goodchild. Discrete space location-allocation solutions form genetic algorithms. *Annals of Operations Research*, 6:35–46, 1986.

[HM97] P. Hansen and N. Mladenović. Variable neighborhood search for the p-median. *Location Science*, 5(4):207–226, 1997.

[HM01a] P. Hansen and N. Maldenović. Developments of Variable Neighborhood Search. In C.C. Ribeiro and P. Hansen, editors, *Essays and Surveys in Metaheuristics*, pages 415–439. Kluwer Academic Publishers, 2001.

[HM01b] P. Hansen and N. Mladenović. Variable Neighborhood Search. In F. Glover and G. Kochenberger, editors, *Handbook of Metaheuristics*, 2001. To appear. A preliminary version is available at http://smg.ulb.ac.be/Preprints/Hansen01_02.html.

[HM01c] P. Hansen and N. Mladenović. Variable neighborhood search: Principles and applications. *European Journal of Operations Research*, 130(3):449–467, 2001.

[HMPB01] P. Hansen, N. Maldenović, and D. Pérez-Brito. Variable neighborhood decomposition search. *Journal of Heuristics*, 7:335–350, 2001.

[Hol75] J.H. Holland. *Adaptation in Natural and Artificial Systems*. University of Michigan Press, Ann Arbor, 1975.

[JBB02] J.H. Jaramillo, J. Bhadury, and R. Batta. On the use of genetic algorithms to solve location problems. *Computers and Operations Research*, 29:761–779, 2002.

[KH79] O. Kariv and S.L. Hakimi. An algorithmic approach to network location problems. II: The p-medians. *SIAM Journal on Applied Mathematics*, 37:539–560, 1979.

[KNP03] J. Kalcsics, S. Nickel, and J. Puerto. Multifacility Ordered Median Problems on networks: a further analysis. *Networks*, 41(1):1–12, 2003.

[MF90] P.B. Mirchandani and R.L. Francis. *Discrete Location Theory*. Wiley, New York, NY, 1990.

[MH97] N. Mladenović and P. Hansen. Variable neighbourhood search. *Computers and Operations Research*, 24:1097–1100, 1997.

[Mic96] Z. Michalewicz. *Genetic Algorithms + Data Structures = Evolution Programs*. Springer-Verlag, third edition, 1996.

[Mla95] N. Mladenović. A Variable Neighborhood Algorithm - A New Metaheuristic for Combinatorial Optimization. Abstracts of papers presented at Optimization Days, Montréal, page 112, 1995.

[MLH00] N. Mladenović, M. Labbé, and P. Hansen. Solving the p-center problem by tabu search and variable neighborhood search. A preliminary version is available at http://smg.ulb.ac.be/Preprints/Labbe00_20.html. To appear in Networks, 2000.

[MPRGMV94] J.A. Moreno Pérez, J.L. Roda García, and J.M. Moreno Vega. A parallel genetic algorithm for the discrete p-median problem. In *Studies in Locational Analysis*, volume 7, pages 131–141, 1994.

[MV96] J.M. Moreno Vega. *Metaheurísticas en Localización: Análisis Teórico y Experimental*. PhD thesis, University of La Laguna, 1996. (In Spanish).

[Nic01] S. Nickel. Discrete Ordered Weber problems. In B. Fleischmann, R. Lasch, U. Derigs, W. Domschke, and U. Rieder, editors, *Operations Research Proceedings 2000*, pages 71–76. Springer, 2001.

[NP99] S. Nickel and J. Puerto. A unified approach to network location problems. *Networks*, 34:283–290, 1999.

[PF95] J. Puerto and F.R. Fernández. The symmetrical single facility location problem. Technical Report, Faculty of Mathematics, University of Sevilla, 1995.

[PF00] J. Puerto and F.R. Fernández. Geometrical properties of the symmetrical single facility location problem. *Journal of Nonlinear and Convex Analysis*, 1(3):321–342, 2000.

[RCNPF00] A. Rodríguez-Chía, S. Nickel, J. Puerto, and F.R. Fernández. A flexible approach to location problems. *Mathematical Methods of Operations Research*, 51:69–89, 2000.

[Sch81] H.P. Schwefel. *Numerical Optimization of Computer Models*. Wiley, Chichester, 1981.

[Whi83] R.A. Whitaker. A fast algorithm for the greedy interchange for large-scale clustering and median location problems. *INFOR*, 21:95–108, 1983.

Published reports of the Fraunhofer ITWM

The PDF-files of the following reports are available under:
www.itwm.fraunhofer.de/zentral/berichte.html

1. D. Hietel, K. Steiner, J. Struckmeier

A Finite - Volume Particle Method for Compressible Flows

We derive a new class of particle methods for conservation laws, which are based on numerical flux functions to model the interactions between moving particles. The derivation is similar to that of classical Finite-Volume methods; except that the fixed grid structure in the Finite-Volume method is substituted by so-called mass packets of particles. We give some numerical results on a shock wave solution for Burgers equation as well as the well-known one-dimensional shock tube problem.
(19 pages, 1998)

2. M. Feldmann, S. Seibold

Damage Diagnosis of Rotors: Application of Hilbert Transform and Multi-Hypothesis Testing

In this paper, a combined approach to damage diagnosis of rotors is proposed. The intention is to employ signal-based as well as model-based procedures for an improved detection of size and location of the damage. In a first step, Hilbert transform signal processing techniques allow for a computation of the signal envelope and the instantaneous frequency, so that various types of non-linearities due to a damage may be identified and classified based on measured response data. In a second step, a multi-hypothesis bank of Kalman Filters is employed for the detection of the size and location of the damage based on the information of the type of damage provided by the results of the Hilbert transform.
Keywords: Hilbert transform, damage diagnosis, Kalman filtering, non-linear dynamics
(23 pages, 1998)

3. Y. Ben-Haim, S. Seibold

Robust Reliability of Diagnostic Multi-Hypothesis Algorithms: Application to Rotating Machinery

Damage diagnosis based on a bank of Kalman filters, each one conditioned on a specific hypothesized system condition, is a well recognized and powerful diagnostic tool. This multi-hypothesis approach can be applied to a wide range of damage conditions. In this paper, we will focus on the diagnosis of cracks in rotating machinery. The question we address is: how to optimize the multi-hypothesis algorithm with respect to the uncertainty of the spatial form and location of cracks and their resulting dynamic effects. First, we formulate a measure of the reliability of the diagnostic algorithm, and then we discuss modifications of the diagnostic algorithm for the maximization of the reliability. The reliability of a diagnostic algorithm is measured by the amount of uncertainty consistent with no-failure of the diagnosis. Uncertainty is quantitatively represented with convex models.
Keywords: Robust reliability, convex models, Kalman filtering, multi-hypothesis diagnosis, rotating machinery, crack diagnosis
(24 pages, 1998)

4. F.-Th. Lentes, N. Siedow

Three-dimensional Radiative Heat Transfer in Glass Cooling Processes

For the numerical simulation of 3D radiative heat transfer in glasses and glass melts, practically applicable mathematical methods are needed to handle such problems optimal using workstation class computers. Since the exact solution would require super-computer capabilities we concentrate on approximate solutions with a high degree of accuracy. The following approaches are studied: 3D diffusion approximations and 3D ray-tracing methods.
(23 pages, 1998)

5. A. Klar, R. Wegener

A hierarchy of models for multilane vehicular traffic
Part I: Modeling

In the present paper multilane models for vehicular traffic are considered. A microscopic multilane model based on reaction thresholds is developed. Based on this model an Enskog like kinetic model is developed. In particular, care is taken to incorporate the correlations between the vehicles. From the kinetic model a fluid dynamic model is derived. The macroscopic coefficients are deduced from the underlying kinetic model. Numerical simulations are presented for all three levels of description in [10]. Moreover, a comparison of the results is given there.
(23 pages, 1998)

Part II: Numerical and stochastic investigations

In this paper the work presented in [6] is continued. The present paper contains detailed numerical investigations of the models developed there. A numerical method to treat the kinetic equations obtained in [6] are presented and results of the simulations are shown. Moreover, the stochastic correlation model used in [6] is described and investigated in more detail.
(17 pages, 1998)

6. A. Klar, N. Siedow

Boundary Layers and Domain Decomposition for Radiative Heat Transfer and Diffusion Equations: Applications to Glass Manufacturing Processes

In this paper domain decomposition methods for radiative transfer problems including conductive heat transfer are treated. The paper focuses on semi-transparent materials, like glass, and the associated conditions at the interface between the materials. Using asymptotic analysis we derive conditions for the coupling of the radiative transfer equations and a diffusion approximation. Several test cases are treated and a problem appearing in glass manufacturing processes is computed. The results clearly show the advantages of a domain decomposition approach. Accuracy equivalent to the solution of the global radiative transfer solution is achieved, whereas computation time is strongly reduced.
(24 pages, 1998)

7. I. Choquet

Heterogeneous catalysis modelling and numerical simulation in rarified gas flows Part I: Coverage locally at equilibrium

A new approach is proposed to model and simulate numerically heterogeneous catalysis in rarefied gas flows. It is developed to satisfy all together the following points:
1) describe the gas phase at the microscopic scale, as required in rarefied flows,
2) describe the wall at the macroscopic scale, to avoid prohibitive computational costs and consider not only crystalline but also amorphous surfaces,
3) reproduce on average macroscopic laws correlated with experimental results and
4) derive analytic models in a systematic and exact way. The problem is stated in the general framework of a non static flow in the vicinity of a catalytic and non porous surface (without aging). It is shown that the exact and systematic resolution method based on the Laplace transform, introduced previously by the author to model collisions in the gas phase, can be extended to the present problem. The proposed approach is applied to the modelling of the EleyRideal and LangmuirHinshelwood recombinations, assuming that the coverage is locally at equilibrium. The models are developed considering one atomic species and extended to the general case of several atomic species. Numerical calculations show that the models derived in this way reproduce with accuracy behaviors observed experimentally.
(24 pages, 1998)

8. J. Ohser, B. Steinbach, C. Lang

Efficient Texture Analysis of Binary Images

A new method of determining some characteristics of binary images is proposed based on a special linear filtering. This technique enables the estimation of the area fraction, the specific line length, and the specific integral of curvature. Furthermore, the specific length of the total projection is obtained, which gives detailed information about the texture of the image. The influence of lateral and directional resolution depending on the size of the applied filter mask is discussed in detail. The technique includes a method of increasing directional resolution for texture analysis while keeping lateral resolution as high as possible.
(17 pages, 1998)

9. J. Orlik

Homogenization for viscoelasticity of the integral type with aging and shrinkage

A multiphase composite with periodic distributed inclusions with a smooth boundary is considered in this contribution. The composite component materials are supposed to be linear viscoelastic and aging (of the nonconvolution integral type, for which the Laplace transform with respect to time is not effectively applicable) and are subjected to isotropic shrinkage. The free shrinkage deformation can be considered as a fictitious temperature deformation in the behavior law. The procedure presented in this paper proposes a way to determine average (effective homogenized) viscoelastic and shrinkage (temperature) composite properties and the homogenized stressfield from known properties of the components. This is done by the extension of the asymptotic homogenization technique known for pure elastic nonhomogeneous bodies to the nonhomogeneous thermoviscoelasticity of the integral noncon-

volution type. Up to now, the homogenization theory has not covered viscoelasticity of the integral type. SanchezPalencia (1980), Francfort & Suquet (1987) (see [2], [9]) have considered homogenization for viscoelasticity of the differential form and only up to the first derivative order. The integralmodeled viscoelasticity is more general then the differential one and includes almost all known differential models. The homogenization procedure is based on the construction of an asymptotic solution with respect to a period of the composite structure. This reduces the original problem to some auxiliary boundary value problems of elasticity and viscoelasticity on the unit periodic cell, of the same type as the original non-homogeneous problem. The existence and uniqueness results for such problems were obtained for kernels satisfying some constrain conditions. This is done by the extension of the Volterra integral operator theory to the Volterra operators with respect to the time, whose 1 kernels are space linear operators for any fixed time variables. Some ideas of such approach were proposed in [11] and [12], where the Volterra operators with kernels depending additionally on parameter were considered. This manuscript delivers results of the same nature for the case of the spaceoperator kernels.
(20 pages, 1998)

10. J. Mohring

Helmholtz Resonators with Large Aperture

The lowest resonant frequency of a cavity resonator is usually approximated by the classical Helmholtz formula. However, if the opening is rather large and the front wall is narrow this formula is no longer valid. Here we present a correction which is of third order in the ratio of the diameters of aperture and cavity. In addition to the high accuracy it allows to estimate the damping due to radiation. The result is found by applying the method of matched asymptotic expansions. The correction contains form factors describing the shapes of opening and cavity. They are computed for a number of standard geometries. Results are compared with numerical computations.
(21 pages, 1998)

11. H. W. Hamacher, A. Schöbel

On Center Cycles in Grid Graphs

Finding "good" cycles in graphs is a problem of great interest in graph theory as well as in locational analysis. We show that the center and median problems are NP hard in general graphs. This result holds both for the variable cardinality case (i.e. all cycles of the graph are considered) and the fixed cardinality case (i.e. only cycles with a given cardinality p are feasible). Hence it is of interest to investigate special cases where the problem is solvable in polynomial time. In grid graphs, the variable cardinality case is, for instance, trivially solvable if the shape of the cycle can be chosen freely. If the shape is fixed to be a rectangle one can analyze rectangles in grid graphs with, in sequence, fixed dimension, fixed cardinality, and variable cardinality. In all cases a complete characterization of the optimal cycles and closed form expressions of the optimal objective values are given, yielding polynomial time algorithms for all cases of center rectangle problems. Finally, it is shown that center cycles can be chosen as rectangles for small cardinalities such that the center cycle problem in grid graphs is in these cases completely solved.
(15 pages, 1998)

12. H. W. Hamacher, K.-H. Küfer

Inverse radiation therapy planning - a multiple objective optimisation approach

For some decades radiation therapy has been proved successful in cancer treatment. It is the major task of clinical radiation treatment planning to realize on the one hand a high level dose of radiation in the cancer tissue in order to obtain maximum tumor control. On the other hand it is obvious that it is absolutely necessary to keep in the tissue outside the tumor, particularly in organs at risk, the unavoidable radiation as low as possible.
No doubt, these two objectives of treatment planning - high level dose in the tumor, low radiation outside the tumor - have a basically contradictory nature. Therefore, it is no surprise that inverse mathematical models with dose distribution bounds tend to be infeasible in most cases. Thus, there is need for approximations compromising between overdosing the organs at risk and underdosing the target volume.
Differing from the currently used time consuming iterative approach, which measures deviation from an ideal (non-achievable) treatment plan using recursively trial-and-error weights for the organs of interest, we go a new way trying to avoid a priori weight choices and consider the treatment planning problem as a multiple objective linear programming problem: with each organ of interest, target tissue as well as organs at risk, we associate an objective function measuring the maximal deviation from the prescribed doses.
We build up a data base of relatively few efficient solutions representing and approximating the variety of Pareto solutions of the multiple objective linear programming problem. This data base can be easily scanned by physicians looking for an adequate treatment plan with the aid of an appropriate online tool.
(14 pages, 1999)

13. C. Lang, J. Ohser, R. Hilfer

On the Analysis of Spatial Binary Images

This paper deals with the characterization of microscopically heterogeneous, but macroscopically homogeneous spatial structures. A new method is presented which is strictly based on integral-geometric formulae such as Crofton's intersection formulae and Hadwiger's recursive definition of the Euler number. The corresponding algorithms have clear advantages over other techniques. As an example of application we consider the analysis of spatial digital images produced by means of Computer Assisted Tomography.
(20 pages, 1999)

14. M. Junk

On the Construction of Discrete Equilibrium Distributions for Kinetic Schemes

A general approach to the construction of discrete equilibrium distributions is presented. Such distribution functions can be used to set up Kinetic Schemes as well as Lattice Boltzmann methods. The general principles are also applied to the construction of Chapman Enskog distributions which are used in Kinetic Schemes for compressible Navier-Stokes equations.
(24 pages, 1999)

15. M. Junk, S. V. Raghurame Rao

A new discrete velocity method for Navier-Stokes equations

The relation between the Lattice Boltzmann Method, which has recently become popular, and the Kinetic Schemes, which are routinely used in Computational Fluid Dynamics, is explored. A new discrete velocity model for the numerical solution of Navier-Stokes equations for incompressible fluid flow is presented by combining both the approaches. The new scheme can be interpreted as a pseudo-compressibility method and, for a particular choice of parameters, this interpretation carries over to the Lattice Boltzmann Method.
(20 pages, 1999)

16. H. Neunzert

Mathematics as a Key to Key Technologies

The main part of this paper will consist of examples, how mathematics really helps to solve industrial problems; these examples are taken from our Institute for Industrial Mathematics, from research in the Technomathematics group at my university, but also from ECMI groups and a company called TecMath, which originated 10 years ago from my university group and has already a very successful history.
(39 pages (4 PDF-Files), 1999)

17. J. Ohser, K. Sandau

Considerations about the Estimation of the Size Distribution in Wicksell's Corpuscle Problem

Wicksell's corpuscle problem deals with the estimation of the size distribution of a population of particles, all having the same shape, using a lower dimensional sampling probe. This problem was originary formulated for particle systems occurring in life sciences but its solution is of actual and increasing interest in materials science. From a mathematical point of view, Wicksell's problem is an inverse problem where the interesting size distribution is the unknown part of a Volterra equation. The problem is often regarded ill-posed, because the structure of the integrand implies unstable numerical solutions. The accuracy of the numerical solutions is considered here using the condition number, which allows to compare different numerical methods with different (equidistant) class sizes and which indicates, as one result, that a finite section thickness of the probe reduces the numerical problems. Furthermore, the relative error of estimation is computed which can be split into two parts. One part consists of the relative discretization error that increases for increasing class size, and the second part is related to the relative statistical error which increases with decreasing class size. For both parts, upper bounds can be given and the sum of them indicates an optimal class width depending on some specific constants.
(18 pages, 1999)

18. E. Carrizosa, H. W. Hamacher, R. Klein, S. Nickel

Solving nonconvex planar location problems by finite dominating sets

It is well-known that some of the classical location problems with polyhedral gauges can be solved in polynomial time by finding a finite dominating set, i.e. a finite set of candidates guaranteed to contain at least one optimal location.
In this paper it is first established that this result holds

for a much larger class of problems than currently considered in the literature. The model for which this result can be proven includes, for instance, location problems with attraction and repulsion, and location-allocation problems.
Next, it is shown that the approximation of general gauges by polyhedral ones in the objective function of our general model can be analyzed with regard to the subsequent error in the optimal objective value. For the approximation problem two different approaches are described, the sandwich procedure and the greedy algorithm. Both of these approaches lead - for fixed epsilon - to polynomial approximation algorithms with accuracy epsilon for solving the general model considered in this paper.
Keywords: Continuous Location, Polyhedral Gauges, Finite Dominating Sets, Approximation, Sandwich Algorithm, Greedy Algorithm
(19 pages, 2000)

19. A. Becker

A Review on Image Distortion Measures

Within this paper we review image distortion measures. A distortion measure is a criterion that assigns a "quality number" to an image. We distinguish between mathematical distortion measures and those distortion measures in-cooperating a priori knowledge about the imaging devices (e. g. satellite images), image processing algorithms or the human physiology. We will consider representative examples of different kinds of distortion measures and are going to discuss them.
Keywords: Distortion measure, human visual system
(26 pages, 2000)

20. H. W. Hamacher, M. Labbé, S. Nickel, T. Sonneborn

Polyhedral Properties of the Uncapacitated Multiple Allocation Hub Location Problem

We examine the feasibility polyhedron of the uncapacitated hub location problem (UHL) with multiple allocation, which has applications in the fields of air passenger and cargo transportation, telecommunication and postal delivery services. In particular we determine the dimension and derive some classes of facets of this polyhedron. We develop some general rules about lifting facets from the uncapacitated facility location (UFL) for UHL and projecting facets from UHL to UFL. By applying these rules we get a new class of facets for UHL which dominates the inequalities in the original formulation. Thus we get a new formulation of UHL whose constraints are all facet–defining. We show its superior computational performance by benchmarking it on a well known data set.
Keywords: integer programming, hub location, facility location, valid inequalities, facets, branch and cut
(21 pages, 2000)

21. H. W. Hamacher, A. Schöbel

Design of Zone Tariff Systems in Public Transportation

Given a public transportation system represented by its stops and direct connections between stops, we consider two problems dealing with the prices for the customers: The fare problem in which subsets of stops are already aggregated to zones and "good" tariffs have to be found in the existing zone system. Closed form solutions for the fare problem are presented for three objective functions. In the zone problem the design of the zones is part of the problem. This problem is NP hard and we therefore propose three heuristics which prove to be very successful in the redesign of one of Germany's transportation systems.
(30 pages, 2001)

22. D. Hietel, M. Junk, R. Keck, D. Teleaga:

The Finite-Volume-Particle Method for Conservation Laws

In the Finite-Volume-Particle Method (FVPM), the weak formulation of a hyperbolic conservation law is discretized by restricting it to a discrete set of test functions. In contrast to the usual Finite-Volume approach, the test functions are not taken as characteristic functions of the control volumes in a spatial grid, but are chosen from a partition of unity with smooth and overlapping partition functions (the particles), which can even move along pre- scribed velocity fields. The information exchange between particles is based on standard numerical flux functions. Geometrical information, similar to the surface area of the cell faces in the Finite-Volume Method and the corresponding normal directions are given as integral quantities of the partition functions. After a brief derivation of the Finite-Volume-Particle Method, this work focuses on the role of the geometric coefficients in the scheme.
(16 pages, 2001)

23. T. Bender, H. Hennes, J. Kalcsics, M. T. Melo, S. Nickel

Location Software and Interface with GIS and Supply Chain Management

The objective of this paper is to bridge the gap between location theory and practice. To meet this objective focus is given to the development of software capable of addressing the different needs of a wide group of users. There is a very active community on location theory encompassing many research fields such as operations research, computer science, mathematics, engineering, geography, economics and marketing. As a result, people working on facility location problems have a very diverse background and also different needs regarding the software to solve these problems. For those interested in non-commercial applications (e. g. students and researchers), the library of location algorithms (LoLA can be of considerable assistance. LoLA contains a collection of efficient algorithms for solving planar, network and discrete facility location problems. In this paper, a detailed description of the functionality of LoLA is presented. In the fields of geography and marketing, for instance, solving facility location problems requires using large amounts of demographic data. Hence, members of these groups (e. g. urban planners and sales managers) often work with geographical information too s. To address the specific needs of these users, LoLA was inked to a geographical information system (GIS) and the details of the combined functionality are described in the paper. Finally, there is a wide group of practitioners who need to solve large problems and require special purpose software with a good data interface. Many of such users can be found, for example, in the area of supply chain management (SCM). Logistics activities involved in strategic SCM include, among others, facility location planning. In this paper, the development of a commercial location software tool is also described. The too is embedded in the Advanced Planner and Optimizer SCM software developed by SAP AG, Walldorf, Germany. The paper ends with some conclusions and an outlook to future activities.
Keywords: facility location, software development, geographical information systems, supply chain management.
(48 pages, 2001)

24. H. W. Hamacher, S. A. Tjandra

Mathematical Modelling of Evacuation Problems: A State of Art

This paper details models and algorithms which can be applied to evacuation problems. While it concentrates on building evacuation many of the results are applicable also to regional evacuation. All models consider the time as main parameter, where the travel time between components of the building is part of the input and the overall evacuation time is the output. The paper distinguishes between macroscopic and microscopic evacuation models both of which are able to capture the evacuees' movement over time.
Macroscopic models are mainly used to produce good lower bounds for the evacuation time and do not consider any individual behavior during the emergency situation. These bounds can be used to analyze existing buildings or help in the design phase of planning a building. Macroscopic approaches which are based on dynamic network flow models (minimum cost dynamic flow, maximum dynamic flow, universal maximum flow, quickest path and quickest flow) are described. A special feature of the presented approach is the fact, that travel times of evacuees are not restricted to be constant, but may be density dependent. Using multicriteria optimization priority regions and blockage due to fire or smoke may be considered. It is shown how the modelling can be done using time parameter either as discrete or continuous parameter.
Microscopic models are able to model the individual evacuee's characteristics and the interaction among evacuees which influence their movement. Due to the corresponding huge amount of data one uses simulation approaches. Some probabilistic laws for individual evacuee's movement are presented. Moreover ideas to model the evacuee's movement using cellular automata (CA) and resulting software are presented. In this paper we will focus on macroscopic models and only summarize some of the results of the microscopic approach. While most of the results are applicable to general evacuation situations, we concentrate on building evacuation.
(44 pages, 2001)

25. J. Kuhnert, S. Tiwari

Grid free method for solving the Poisson equation

A Grid free method for solving the Poisson equation is presented. This is an iterative method. The method is based on the weighted least squares approximation in which the Poisson equation is enforced to be satisfied in every iterations. The boundary conditions can also be enforced in the iteration process. This is a local approximation procedure. The Dirichlet, Neumann and mixed boundary value problems on a unit square are presented and the analytical solutions are compared with the exact solutions. Both solutions matched perfectly.
Keywords: Poisson equation, Least squares method, Grid free method
(19 pages, 2001)

26. T. Götz, H. Rave, D. Reinel-Bitzer, K. Steiner, H. Tiemeier

Simulation of the fiber spinning process

To simulate the influence of process parameters to the melt spinning process a fiber model is used and coupled with CFD calculations of the quench air flow. In the fiber model energy, momentum and mass balance are solved for the polymer mass flow. To calculate the quench air the Lattice Boltzmann method is used. Simulations and experiments for different process parameters and hole configurations are compared and show a good agreement.

Keywords: Melt spinning, fiber model, Lattice Boltzmann, CFD

(19 pages, 2001)

27. A. Zemitis

On interaction of a liquid film with an obstacle

In this paper mathematical models for liquid films generated by impinging jets are discussed. Attention is stressed to the interaction of the liquid film with some obstacle. S. G. Taylor [Proc. R. Soc. London Ser. A 253, 313 (1959)] found that the liquid film generated by impinging jets is very sensitive to properties of the wire which was used as an obstacle. The aim of this presentation is to propose a modification of the Taylor's model, which allows to simulate the film shape in cases, when the angle between jets is different from 180°. Numerical results obtained by discussed models give two different shapes of the liquid film similar as in Taylors experiments. These two shapes depend on the regime: either droplets are produced close to the obstacle or not. The difference between two regimes becomes larger if the angle between jets decreases. Existence of such two regimes can be very essential for some applications of impinging jets, if the generated liquid film can have a contact with obstacles.

Keywords: impinging jets, liquid film, models, numerical solution, shape

(22 pages, 2001)

28. I. Ginzburg, K. Steiner

Free surface lattice-Boltzmann method to model the filling of expanding cavities by Bingham Fluids

The filling process of viscoplastic metal alloys and plastics in expanding cavities is modelled using the lattice Boltzmann method in two and three dimensions. These models combine the regularized Bingham model for viscoplastic with a free-interface algorithm. The latter is based on a modified immiscible lattice Boltzmann model in which one species is the fluid and the other one is considered as vacuum. The boundary conditions at the curved liquid-vacuum interface are met without any geometrical front reconstruction from a first-order Chapman-Enskog expansion. The numerical results obtained with these models are found in good agreement with available theoretical and numerical analysis.

Keywords: Generalized LBE, free-surface phenomena, interface boundary conditions, filling processes, Bingham viscoplastic model, regularized models

(22 pages, 2001)

29. H. Neunzert

»Denn nichts ist für den Menschen als Menschen etwas wert, was er nicht mit Leidenschaft tun kann«

Vortrag anlässlich der Verleihung des Akademiepreises des Landes Rheinland-Pfalz am 21.11.2001

Was macht einen guten Hochschullehrer aus? Auf diese Frage gibt es sicher viele verschiedene, fachbezogene Antworten, aber auch ein paar allgemeine Gesichtspunkte: es bedarf der »Leidenschaft« für die Forschung (Max Weber), aus der dann auch die Begeisterung für die Lehre erwächst. Forschung und Lehre gehören zusammen, um die Wissenschaft als lebendiges Tun vermitteln zu können. Der Vortrag gibt Beispiele dafür, wie in angewandter Mathematik Forschungsaufgaben aus praktischen Alltagsproblemstellungen erwachsen, die in die Lehre auf verschiedenen Stufen (Gymnasium bis Graduiertenkolleg) einfließen; er leitet damit auch zu einem aktuellen Forschungsgebiet, der Mehrskalenanalyse mit ihren vielfältigen Anwendungen in Bildverarbeitung, Materialentwicklung und Strömungsmechanik über, was aber nur kurz gestreift wird. Mathematik erscheint hier als eine moderne Schlüsseltechnologie, die aber auch enge Beziehungen zu den Geistes- und Sozialwissenschaften hat.

Keywords: Lehre, Forschung, angewandte Mathematik, Mehrskalenanalyse, Strömungsmechanik

(18 pages, 2001)

30. J. Kuhnert, S. Tiwari

Finite pointset method based on the projection method for simulations of the incompressible Navier-Stokes equations

A Lagrangian particle scheme is applied to the projection method for the incompressible Navier-Stokes equations. The approximation of spatial derivatives is obtained by the weighted least squares method. The pressure Poisson equation is solved by a local iterative procedure with the help of the least squares method. Numerical tests are performed for two dimensional cases. The Couette flow, Poiseuelle flow, decaying shear flow and the driven cavity flow are presented. The numerical solutions are obtained for stationary as well as instationary cases and are compared with the analytical solutions for channel flows. Finally, the driven cavity in a unit square is considered and the stationary solution obtained from this scheme is compared with that from the finite element method.

Keywords: Incompressible Navier-Stokes equations, Meshfree method, Projection method, Particle scheme, Least squares approximation

AMS subject classification: 76D05, 76M28

(25 pages, 2001)

31. R. Korn, M. Krekel

Optimal Portfolios with Fixed Consumption or Income Streams

We consider some portfolio optimisation problems where either the investor has a desire for an a priori specified consumption stream or/and follows a deterministic pay in scheme while also trying to maximize expected utility from final wealth. We derive explicit closed form solutions for continuous and discrete monetary streams. The mathematical method used is classical stochastic control theory.

Keywords: Portfolio optimisation, stochastic control, HJB equation, discretisation of control problems.

(23 pages, 2002)

32. M. Krekel

Optimal portfolios with a loan dependent credit spread

If an investor borrows money he generally has to pay higher interest rates than he would have received, if he had put his funds on a savings account. The classical model of continuous time portfolio optimisation ignores this effect. Since there is obviously a connection between the default probability and the total percentage of wealth, which the investor is in debt, we study portfolio optimisation with a control dependent interest rate. Assuming a logarithmic and a power utility function, respectively, we prove explicit formulae of the optimal control.

Keywords: Portfolio optimisation, stochastic control, HJB equation, credit spread, log utility, power utility, non-linear wealth dynamics

(25 pages, 2002)

33. J. Ohser, W. Nagel, K. Schladitz

The Euler number of discretized sets - on the choice of adjacency in homogeneous lattices

Two approaches for determining the Euler-Poincaré characteristic of a set observed on lattice points are considered in the context of image analysis { the integral geometric and the polyhedral approach. Information about the set is assumed to be available on lattice points only. In order to retain properties of the Euler number and to provide a good approximation of the true Euler number of the original set in the Euclidean space, the appropriate choice of adjacency in the lattice for the set and its background is crucial. Adjacencies are defined using tessellations of the whole space into polyhedrons. In R 3 , two new 14 adjacencies are introduced additionally to the well known 6 and 26 adjacencies. For the Euler number of a set and its complement, a consistency relation holds. Each of the pairs of adjacencies (14:1; 14:1), (14:2; 14:2), (6; 26), and (26; 6) is shown to be a pair of complementary adjacencies with respect to this relation. That is, the approximations of the Euler numbers are consistent if the set and its background (complement) are equipped with this pair of adjacencies. Furthermore, sufficient conditions for the correctness of the approximations of the Euler number are given. The analysis of selected microstructures and a simulation study illustrate how the estimated Euler number depends on the chosen adjacency. It also shows that there is not a uniquely best pair of adjacencies with respect to the estimation of the Euler number of a set in Euclidean space.

Keywords: image analysis, Euler number, neighborhod relationships, cuboidal lattice

(32 pages, 2002)

34. I. Ginzburg, K. Steiner

Lattice Boltzmann Model for Free-Surface flow and Its Application to Filling Process in Casting

A generalized lattice Boltzmann model to simulate free-surface is constructed in both two and three dimensions. The proposed model satisfies the interfacial boundary conditions accurately. A distinctive feature of the model is that the collision processes is carried out only on the points occupied partially or fully by the fluid. To maintain a sharp interfacial front, the method includes an anti-diffusion algorithm. The unknown distribution functions at the interfacial region are constructed according to the first order Chapman-Enskog analysis. The interfacial boundary conditions are satis-

fied exactly by the coefficients in the Chapman-Enskog expansion. The distribution functions are naturally expressed in the local interfacial coordinates. The macroscopic quantities at the interface are extracted from the least-square solutions of a locally linearized system obtained from the known distribution functions. The proposed method does not require any geometric front construction and is robust for any interfacial topology. Simulation results of realistic filling process are presented: rectangular cavity in two dimensions and Hammer box, Campbell box, Sheffield box, and Motorblock in three dimensions. To enhance the stability at high Reynolds numbers, various upwind-type schemes are developed. Free-slip and no-slip boundary conditions are also discussed.
Keywords: Lattice Boltzmann models; free-surface phenomena; interface boundary conditions; filling processes; injection molding; volume of fluid method; interface boundary conditions; advection-schemes; upwind-schemes
(54 pages, 2002)

35. M. Günther, A. Klar, T. Materne, R. Wegener

Multivalued fundamental diagrams and stop and go waves for continuum traffic equations

In the present paper a kinetic model for vehicular traffic leading to multivalued fundamental diagrams is developed and investigated in detail. For this model phase transitions can appear depending on the local density and velocity of the flow. A derivation of associated macroscopic traffic equations from the kinetic equation is given. Moreover, numerical experiments show the appearance of stop and go waves for highway traffic with a bottleneck.
Keywords: traffic flow, macroscopic equations, kinetic derivation, multivalued fundamental diagram, stop and go waves, phase transitions
(25 pages, 2002)

36. S. Feldmann, P. Lang, D. Prätzel-Wolters

Parameter influence on the zeros of network determinants

To a network N(q) with determinant D(s;q) depending on a parameter vector $q \hat{I} R^r$ via identification of some of its vertices, a network N^ (q) is assigned. The paper deals with procedures to find N^ (q), such that its determinant D^ (s;q) admits a factorization in the determinants of appropriate subnetworks, and with the estimation of the deviation of the zeros of D^ from the zeros of D. To solve the estimation problem state space methods are applied.
Keywords: Networks, Equicofactor matrix polynomials, Realization theory, Matrix perturbation theory
(30 pages, 2002)

37. K. Koch, J. Ohser, K. Schladitz

Spectral theory for random closed sets and estimating the covariance via frequency space

A spectral theory for stationary random closed sets is developed and provided with a sound mathematical basis. Definition and proof of existence of the Bartlett spectrum of a stationary random closed set as well as the proof of a Wiener-Khintchine theorem for the power spectrum are used to two ends: First, well known second order characteristics like the covariance can be estimated faster than usual via frequency space. Second, the Bartlett spectrum and the power spectrum can be used as second order characteristics in frequency space. Examples show, that in some cases information about the random closed set is easier to obtain from these characteristics in frequency space than from their real world counterparts.
Keywords: Random set, Bartlett spectrum, fast Fourier transform, power spectrum
(28 pages, 2002)

38. D. d'Humières, I. Ginzburg

Multi-reflection boundary conditions for lattice Boltzmann models

We present a unified approach of several boundary conditions for lattice Boltzmann models. Its general framework is a generalization of previously introduced schemes such as the bounce-back rule, linear or quadratic interpolations, etc. The objectives are two fold: first to give theoretical tools to study the existing boundary conditions and their corresponding accuracy; secondly to design formally third- order accurate boundary conditions for general flows. Using these boundary conditions, Couette and Poiseuille flows are exact solution of the lattice Boltzmann models for a Reynolds number $Re = 0$ (Stokes limit).
Numerical comparisons are given for Stokes flows in periodic arrays of spheres and cylinders, linear periodic array of cylinders between moving plates and for Navier-Stokes flows in periodic arrays of cylinders for $Re < 200$. These results show a significant improvement of the overall accuracy when using the linear interpolations instead of the bounce-back reflection (up to an order of magnitude on the hydrodynamics fields). Further improvement is achieved with the new multi-reflection boundary conditions, reaching a level of accuracy close to the quasi-analytical reference solutions, even for rather modest grid resolutions and few points in the narrowest channels. More important, the pressure and velocity fields in the vicinity of the obstacles are much smoother with multi-reflection than with the other boundary conditions.
Finally the good stability of these schemes is highlighted by some simulations of moving obstacles: a cylinder between flat walls and a sphere in a cylinder.
Keywords: lattice Boltzmann equation, boudary condistions, bounce-back rule, Navier-Stokes equation
(72 pages, 2002)

39. R. Korn

Elementare Finanzmathematik

Im Rahmen dieser Arbeit soll eine elementar gehaltene Einführung in die Aufgabenstellungen und Prinzipien der modernen Finanzmathematik gegeben werden. Insbesondere werden die Grundlagen der Modellierung von Aktienkursen, der Bewertung von Optionen und der Portfolio-Optimierung vorgestellt. Natürlich können die verwendeten Methoden und die entwickelte Theorie nicht in voller Allgemeinheit für den Schulunterricht verwendet werden, doch sollen einzelne Prinzipien so heraus gearbeitet werden, dass sie auch an einfachen Beispielen verstanden werden können.
Keywords: Finanzmathematik, Aktien, Optionen, Portfolio-Optimierung, Börse, Lehrerweiterbildung, Mathematikunterricht
(98 pages, 2002)

40. J. Kallrath, M. C. Müller, S. Nickel

Batch Presorting Problems: Models and Complexity Results

In this paper we consider short term storage systems. We analyze presorting strategies to improve the effiency of these storage systems. The presorting task is called Batch PreSorting Problem (BPSP). The BPSP is a variation of an assigment problem, i. e., it has an assigment problem kernel and some additional constraints. We present different types of these presorting problems, introduce mathematical programming formulations and prove the NP-completeness for one type of the BPSP. Experiments are carried out in order to compare the different model formulations and to investigate the behavior of these models.
Keywords: Complexity theory, Integer programming, Assigment, Logistics
(19 pages, 2002)

41. J. Linn

On the frame-invariant description of the phase space of the Folgar-Tucker equation

The Folgar-Tucker equation is used in flow simulations of fiber suspensions to predict fiber orientation depending on the local flow. In this paper, a complete, frame-invariant description of the phase space of this differential equation is presented for the first time.
Key words: fiber orientation, Folgar-Tucker equation, injection molding
(5 pages, 2003)

42. T. Hanne, S. Nickel

A Multi-Objective Evolutionary Algorithm for Scheduling and Inspection Planning in Software Development Projects

In this article, we consider the problem of planning inspections and other tasks within a software development (SD) project with respect to the objectives quality (no. of defects), project duration, and costs. Based on a discrete-event simulation model of SD processes comprising the phases coding, inspection, test, and rework, we present a simplified formulation of the problem as a multiobjective optimization problem. For solving the problem (i. e. finding an approximation of the efficient set) we develop a multiobjective evolutionary algorithm. Details of the algorithm are discussed as well as results of its application to sample problems.
Key words: multiple objective programming, project management and scheduling, software development, evolutionary algorithms, efficient set
(29 pages, 2003)

43. T. Bortfeld , K.-H. Küfer, M. Monz, A. Scherrer, C. Thieke, H. Trinkaus

Intensity-Modulated Radiotherapy - A Large Scale Multi-Criteria Programming Problem -

Radiation therapy planning is always a tight rope walk between dangerous insufficient dose in the target volume and life threatening overdosing of organs at risk. Finding ideal balances between these inherently contradictory goals challenges dosimetrists and physicians in their daily practice. Today's planning systems are typically based on a single evaluation function that measures the quality of a radiation treatment plan. Unfortunately, such a one dimensional approach can-

not satisfactorily map the different backgrounds of physicians and the patient dependent necessities. So, too often a time consuming iteration process between evaluation of dose distribution and redefinition of the evaluation function is needed.
In this paper we propose a generic multi-criteria approach based on Pareto's solution concept. For each entity of interest - target volume or organ at risk a structure dependent evaluation function is defined measuring deviations from ideal doses that are calculated from statistical functions. A reasonable bunch of clinically meaningful Pareto optimal solutions are stored in a data base, which can be interactively searched by physicians. The system guarantees dynamical planning as well as the discussion of tradeoffs between different entities.
Mathematically, we model the upcoming inverse problem as a multi-criteria linear programming problem. Because of the large scale nature of the problem it is not possible to solve the problem in a 3D-setting without adaptive reduction by appropriate approximation schemes.
Our approach is twofold: First, the discretization of the continuous problem is based on an adaptive hierarchical clustering process which is used for a local refinement of constraints during the optimization procedure. Second, the set of Pareto optimal solutions is approximated by an adaptive grid of representatives that are found by a hybrid process of calculating extreme compromises and interpolation methods.
Keywords: multiple criteria optimization, representative systems of Pareto solutions, adaptive triangulation, clustering and disaggregation techniques, visualization of Pareto solutions, medical physics, external beam radiotherapy planning, intensity modulated radiotherapy
(31 pages, 2003)

44. T. Halfmann, T. Wichmann

Overview of Symbolic Methods in Industrial Analog Circuit Design

Industrial analog circuits are usually designed using numerical simulation tools. To obtain a deeper circuit understanding, symbolic analysis techniques can additionally be applied. Approximation methods which reduce the complexity of symbolic expressions are needed in order to handle industrial-sized problems. This paper will give an overview to the field of symbolic analog circuit analysis. Starting with a motivation, the state-of-the-art simplification algorithms for linear as well as for nonlinear circuits are presented. The basic ideas behind the different techniques are described, whereas the technical details can be found in the cited references. Finally, the application of linear and nonlinear symbolic analysis will be shown on two example circuits.
Keywords: CAD, automated analog circuit design, symbolic analysis, computer algebra, behavioral modeling, system simulation, circuit sizing, macro modeling, differential-algebraic equations, index
(17 pages, 2003)

45. S. E. Mikhailov, J. Orlik

Asymptotic Homogenisation in Strength and Fatigue Durability Analysis of Composites

Asymptotic homogenisation technique and two-scale convergence is used for analysis of macro-strength and fatigue durability of composites with a periodic structure under cyclic loading. The linear damage accumulation rule is employed in the phenomenological micro-durability conditions (for each component of the composite) under varying cyclic loading. Both local and non-local strength and durability conditions are analysed. The strong convergence of the strength and fatigue damage measure as the structure period tends to zero is proved and their limiting values are estimated.
Keywords: multiscale structures, asymptotic homogenization, strength, fatigue, singularity, non-local conditions
(14 pages, 2003)

46. P. Domínguez-Marín, P. Hansen, N. Mladenović, S. Nickel

Heuristic Procedures for Solving the Discrete Ordered Median Problem

We present two heuristic methods for solving the Discrete Ordered Median Problem (DOMP), for which no such approaches have been developed so far. The DOMP generalizes classical discrete facility location problems, such as the p-median, p-center and Uncapacitated Facility Location problems. The first procedure proposed in this paper is based on a genetic algorithm developed by Moreno Vega [MV96] for p-median and p-center problems. Additionally, a second heuristic approach based on the Variable Neighborhood Search metaheuristic (VNS) proposed by Hansen & Mladenovic [HM97] for the p-median problem is described. An extensive numerical study is presented to show the efficiency of both heuristics and compare them.
Keywords: genetic algorithms, variable neighborhood search, discrete facility location
(31 pages, 2003)

47. N. Boland, P. Domínguez-Marín, S. Nickel, J. Puerto

Exact Procedures for Solving the Discrete Ordered Median Problem

The Discrete Ordered Median Problem (DOMP) generalizes classical discrete location problems, such as the N-median, N-center and Uncapacitated Facility Location problems. It was introduced by Nickel [16], who formulated it as both a nonlinear and a linear integer program. We propose an alternative integer linear programming formulation for the DOMP, discuss relationships between both integer linear programming formulations, and show how properties of optimal solutions can be used to strengthen these formulations. Moreover, we present a specific branch and bound procedure to solve the DOMP more efficiently. We test the integer linear programming formulations and this branch and bound method computationally on randomly generated test problems.
Keywords: discrete location, Integer programming
(41 pages, 2003)

Status quo: January 2003