Fraunhofer Institut Techno- und Wirtschaftsmathematik

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

Location Software and Interface with GIS and Supply Chain Management

Berichte des Fraunhofer ITWM, Nr. 23 (2001)

© Fraunhofer-Institut für Techno- und
Wirtschaftsmathematik ITWM 2001

ISSN 1434-9973

Bericht 23 (2001)

Alle Rechte vorbehalten. Ohne ausdrückliche, schriftliche Genehmigung des Herausgebers ist es nicht gestattet, das Buch oder Teile daraus in irgendeiner Form durch Fotokopie, Mikrofilm oder andere Verfahren zu reproduzieren oder in eine für Maschinen, insbesondere Datenverarbeitungsanlagen, verwendbare Sprache zu übertragen. Dasselbe gilt für das Recht der öffentlichen Wiedergabe.

Warennamen werden ohne Gewährleistung der freien Verwendbarkeit benutzt.

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.fhg.de
Internet: www.itwm.fhg.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

Location Software and Interface with GIS and Supply Chain Management*

Thorsten Bender[1], Holger Hennes[2], Jörg Kalcsics[3], M. Teresa Melo[3], and Stefan Nickel[3]

[1] SAP AG, Walldorf, Germany
[2] University of Kaiserslautern, Kaiserslautern, Germany
[3] Fraunhofer Institut für Techno- und Wirtschaftsmathematik, Kaiserslautern, Germany

Abstract

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 tools. To address the specific needs of these users, LoLA was linked 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 tool 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.

* To appear in *Location Analysis: Applications and Methods*, Z. Drezner and H.W. Hamacher (ed.), Springer-Verlag (2001).

1 Introduction

A distinguishing peculiarity of location theory is that a very active research community exists (see Chap. 1–3, and the homepages of EWGLA (2000) and SOLA (2000)) consisting of people with a very diverse background. Mathematicians, computer scientists, operation researchers, industrial engineers, regional planners, marketing researchers as well as practitioners from various branches work on location problems, but each group gives emphasis to different aspects. This implies that location problems are typically solved with methods from different disciplines. As a result, research articles do not tend to use a common language, and it is very hard to find out if a certain problem has already been addressed because it may have been solved under a completely different name. To cope with this problem, several classification schemes for location models have been proposed (see Handler and Mirchandani 1979; Brandeau and Chiu 1989; Eiselt et al. 1993; Carrizosa et al. 1995), and in this book also a common notation according to Hamacher and Nickel (1998) is used.

However, there is another even more severe problem (at least from a practical point of view): It is very difficult to establish a continuous knowledge transfer between the various groups dealing with location problems since, for example, mathematicians have a completely different way of describing their solution approaches than regional planners. As a consequence, improved and new solutions to important location problems are not used because the people who are in need of a practical solution do not understand the language in which the solution has been stated.

A possible way of approaching this problem is to develop software for facility location problems and make it available to the location community. This is of considerable assistance since the software focuses only on the required input and gives a specified output. The solution procedure itself does not need to be completely clear to a first–time user. However, especially for researchers, it is absolutely necessary that not only the executables are available, but also that the source code can be used and — if needed — modified. From the argumentation given so far it might seem that this is a one way road just giving the practitioners an easier access to new theoretical results. This is not quite true since also only exhaustive comparisons between different solution methods for a specific problem class allow to choose a good solution. Not too rarely a simply good idea from a practitioner may yield a very good algorithm and give the more theory oriented people some work to do. This occurs because researchers would like to explain why a good algorithm is "good" (and also the meaning of "good").

At this point, the reader should be convinced that developing software for location problems is particularly useful (otherwise, you should either stop reading now or reread this introduction until you are fully convinced). Now the question changes from why develop software for location problems to **how** to develop software for location problems. We have many different potential users with completely different requirements for such software:

Teachers would like to have software with a good user interface and an implementation of all the algorithms that are described in their course text books. The software package should have some graphical *Frontend* that visualizes results and shows illustrative examples. Finally, the software should be cheap.

Students would like to have software which both helps them to understand the theoretical material better and also supports them in doing their assignments. They would like to run the software on their home computers and that software should be free.

Researchers would like to have software with as many implemented algorithms as possible. They need the software to construct examples and counter–examples. Moreover, the software should provide a library which can be used for implementing new algorithms faster than those made from scratch. The software should be free and the source code should be available.

Practitioners would like to have software which can handle the exact problem class they need to solve. It should be able to solve very large problem instances in a reasonable amount of time. Finding an optimal solution is not so important. In most situations a good solution suffices. The software should have good interfaces for data import and export. The price is not so important.

Software companies would like to develop software which can be easily configured to handle several problem classes. It should be possible to use only the parts of the software that apply to the user's specific problems. The software should have a modern design and provide the possibility of linking third party software. Therefore, interfaces on different levels are essential. The price is not so important.

In this paper we will describe the software components that we have developed to fulfill the needs of the different groups listed above and indicate how to obtain that software. Other software packages for solving facility location problems can be found on the EWGLA (2000) and SOLA (2000) homepages. However, these packages either specialize on solving a specific problem or their functionality is already contained in the software described below.

The remainder of the paper is organized as follows. First, LoLA (Library of Location Algorithms) is presented in Sect. 2. This software library was partially funded by the Deutsche Forschungsgemeinschaft (DFG) and has been the starting point of an organized implementation of location algorithms (Hamacher et al. 1999). In the next section, we describe how LoLA and geographical information tools can be linked in order to access the enormous amount of demographic data available. In Sect. 4, we describe how location software supports strategic supply chain management decisions. An actual implementation in the framework of the Advanced Planner and Optimizer business application developed by SAP AG, Walldorf, Germany is presented. The paper ends with some conclusions and an outlook to future activities.

2 LoLA– Library of Location Algorithms

2.1 Motivation and Aim

Our main motivation for developing LoLA was, as the name already hints, to build a powerful collection of algorithms for location theory, encompassing planar, network and discrete problems, and to use these algorithms for solving various problems in a fast and easy way. LoLA can be downloaded free of charge from http://www.mathematik.uni-kl.de/˜lola. Moreover, all source codes are available in the above address.

To detect if a given location problem was already treated in the literature and (in the case of LoLA) to apply the appropriate algorithm we need a uniform language to classify the problem. LoLA utilizes the classification scheme of Hamacher and Nickel (1998) for easy access to the implemented algorithms. The scheme consists of the following five positions

$$pos1/pos2/pos3/pos4/pos5 \ .$$

The meaning of each position along with some examples is given in Table 1. If no special assumptions are made in a position, this is indicated by a •. For example, a • in position 4 means that any distance function is considered.

Table 1. Classification scheme for facility location problems

position	meaning	usage (examples)	
1	number of new facilities		
		P	planar problem
2	type of problem	D	discrete problem
		G	problem on a general undirected graph
3	special assumptions and restrictions	$w_m = 1$	all weights are equal
4	type of distance function	γ	a general gauge
5	type of objective function	$\sum$	median problem
		max	center problem

LoLA is designed to address several different user groups. It should accompany lectures in the field of location theory and provide the possibility for students to apply the presented methods to specific problems and "see" location theory in action. Schools make up another application field. A vivid alternative to the classical topics of high school mathematics should be provided in order to bring mathematics and especially location theory closer to pupils. Finally, LoLA enables researchers and software developers to compare their new results with those that already exist or to incorporate some of the available algorithms into their own applications.

LoLA provides a graphical user interface that allows its simple application in industrial projects as well as for demonstrations in high school and university teaching. In addition, a *Text–based* user interface is available to call algorithms of LoLA from other applications. To solve individual facility location problems, a programming interface allows the direct incorporation of specific algorithms of the program library into the implementation of extended routines (*Callable Library*).

To present the different alternatives available in LoLA for solving facility location problems, we will use the following two examples throughout this section.

Example 1. We consider a 1–facility planar minisum problem with the squared Euclidean distance and a convex polyhedron as a forbidden region inside. According to the classification scheme of Hamacher and Nickel (1998), this problem is denoted by $1/P/\mathcal{R} = convpoly/l_2^2/\sum$.

Example 2. The second example we will focus on, is a 4–facility undirected network minisum problem where the solutions are searched on the nodes. In the classification scheme of Hamacher and Nickel (1998) this problem is described by $4/G/\bullet/d(V,V)/\sum$. We will solve the problem with the interchange heuristic of Teitz and Bart (1968).

2.2 System Design

The main component of LoLA is the open source C++ Routine Library which uses the software packages LEDA (Library of Efficient Data types and Algorithms, Mehlhorn et al. 2000) and LP–Solve (Berkelaar 1995) as shown in Fig. 1.

In order to be applicable for teaching, LoLA is available as a stand–alone program. For this purpose there exist two different ways of executing algorithms. One is with the help of the graphical user interface (GUI) of LoLA, called *Frontend*, which is directly attached to the library to load data files, select algorithms and view the results of the optimization. In order to guide the user to the appropriate solution of his/her problem, the *Frontend*, which is implemented in Tcl/Tk (Ousterhout 2000), reflects the classification scheme of Hamacher and Nickel (1998) and provides a detailed help manual. The second possibility is to run algorithms *Text–based* from a console (e.g. from MS-DOS[1], a Unix[2] or Linux Shell). Therefore, specific algorithms can be called from other applications, e.g. a geographical information system (see Sect. 3), providing input for the chosen method and showing the results of the optimization. Data is transferred between the invoking application and LoLA using text files based on a descriptive language.

To develop new applications using algorithms available in LoLA, a *Programming Interface* was devised for researchers and software developers. The *Callable*

[1] MS®-DOS is a registered trademark of Microsoft Corporation, USA.

[2] Unix® is a registered trademark of The Open Group, USA.

Library enables the incorporation of LoLA algorithms into other applications at an implementational level. This design allows fast access to the available functions and the direct transfer of data to and from these functions thus avoiding the detour via ASCII files.

The above mentioned software packages LEDA, Tcl/Tk and LP–Solve are available for the platforms Windows[3] 95/98/NT as well as for Unix and Linux systems. Hence, LoLA is independent of these platforms.

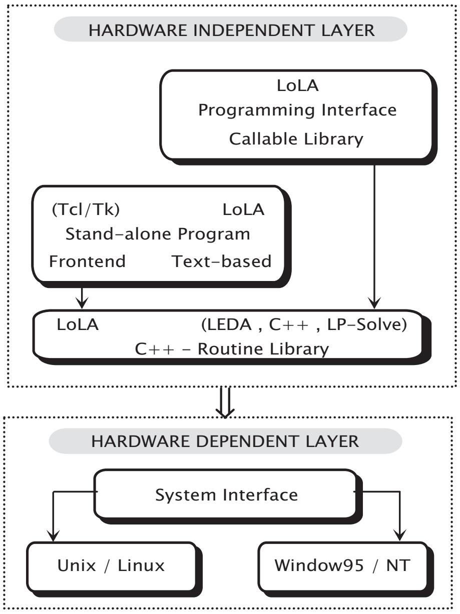

Fig. 1. System Design of LoLA

2.3 The Components of LoLA

Graphical User Interface. The GUI of LoLA, called *Frontend*, is based on the 5–position classification scheme which was introduced by Hamacher and Nickel (1998) and briefly described in Sect. 2.1. If Tcl/Tk is available on your system, calling LoLA creates the window shown in Fig. 2.

Following the classification scheme, the menu of the *Frontend* contains the buttons listed below:

Number has the options 1–facility, N–facility and 1–line.

[3] Windows® is a registered trademark of Microsoft Corporation, USA.

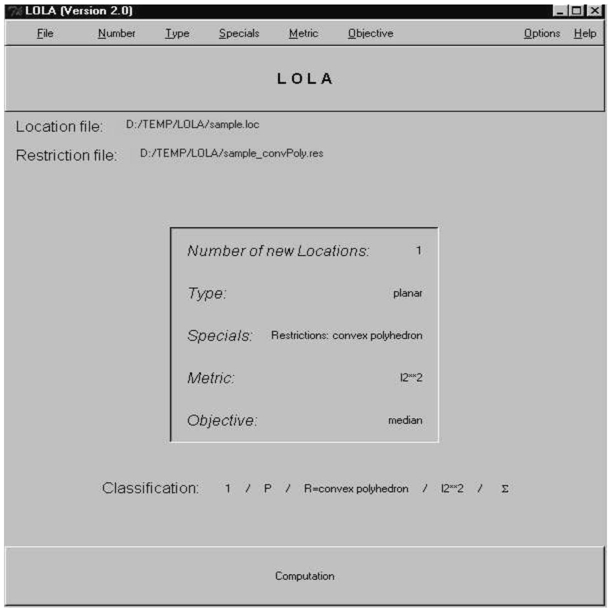

Fig. 2. LoLA *Frontend*

Type is used to choose among the problem types, P: planar, G: graph, D: discrete and T: tree.

Specials is used to select extra assumptions for the problem to be solved which may include:

equal weights If the weights of the existing locations are all equal, fast procedures are available for some types of location problems, e.g. $1/T/w_i = 1/d(V,T)/\Sigma$.

restrictions With the option outside we can choose whether the forbidden region is inside or outside the given restrictions. Possible restriction types are: polyhedron, convex polyhedron, circle, rectangle and all (the restriction file must describe one or more of the above four possibilities in arbitrary order and number).

none (default option)

Metric is used to select the distance function. For planar problems the following norms are available: l_1, l_2, l_2^2, l_∞ and l_p, which are denoted in LoLA by l1, l2, l2**2, linf and lp, respectively. For the l_p–norm, the value of p can be selected in Options under Preferences. Self–created gauges (gauge) and block–norms (block) that is, symmetrical gauges, are also available.

For Graph and Tree algorithms, we have the options d(V,V) and d(V,G) (d(V,T)), respectively, where the optimal points are searched only on the node set V or on the entire graph G (tree graph, T).

Objective to select the type of objective function to be minimized. median and center functions as well as their multicriteria extensions Q–median and Q–center are available.

In addition to the above options, the *Frontend* of LoLA contains the following menus:

File is used to select files providing data for a specific problem, e.g. Load Location, Load Restriction among others. The buttons View Location and View Restriction allow the display of the input data in a separate window (see Fig. 5). Graphical Edit and Create Example provide tools to create new data files.

Help is used to call the online help browser.

Options is used to set general preferences on the maximum number of iterations, the default metric for l_p, the choice of a heuristic for p–facility network problems or other problem dependent settings.

Computation is used to start the optimization.

Example 1 *and* 2 *(cont.). The input data for the planar problem (Example 1) regarding existing locations and their weights are contained in the file "sample.loc", while "sample_convPoly.res" is the restriction file (available in every standard distribution of* LoLA*). The input data for the network problem (Example 2) are contained in the file "sample.gra". Input files can be generated with any ASCII–editor (see Sect. 2.4 for a description of the data format) or with the graphical editor that is available under* Graphical Edit *in the* File *menu. In this editor, the location data can be entered using the mouse. To solve our two problems, three steps must be performed.*

a) *First, the problem type is specified by selecting the following options in the menu bar.*

Options	*Example 1*	*Example 2*
Number	1–facility	N–facility *and then enter* 4
Type	planar	graph
Specials	convex polyhedron	–
Metric	$l2**2$	d(V,V)
Objective	median	median
Options	–	N/G/•/d(V,V)/• *and then select* Exchange–heuristic

Depending on the problem type, the classification string on the LoLA *screen now reads*

$$1/P/R = convex\,polyhedron/l2**2/\Sigma \qquad or \qquad 4/G/\bullet/d(V,V)/\Sigma\ .$$

b) *Next, the input files are loaded using the following menu entries that are available under* File.

Example 1	*Example 2*
Load Location	Load Graph
Load Restriction	

c) *Clicking on* Computation, LoLA *will provide an output window with the solution (see Fig. 3 for the planar problem).*

Fig. 3. Windows with the optimal solution and the numerical result for the planar problem (Example 1)

A solution window, as the one shown in Fig. 3, contains buttons to Save Results of the current solution in a file, to Refresh the solution window and to Close the window. Pressing the button View Results will show the coordinates and the objective function value of the optimal solution (see Fig. 3).

Solution windows of planar problems additionally contain the buttons for showing or hiding, respectively, the Convex Hull of the set of existing facilities, for showing/removing the Weights of the existing facilities and to view the unit ball of a special gauge: View Gauge.

In solution windows of network problems the buttons Node_Weights and Edge_Weights are available for showing or hiding the weights of the nodes and the edges, respectively, of the network (see Fig. 4).

For discrete location problems, buttons for showing or hiding the Fixed Costs of the supply points, the Weights of the demand points or to View the Cost Matrix are additionally available in the solution window.

Text–based User Interface. All algorithms of LoLA can be executed using command–line options in the *text–based mode* for which the *Frontend*, and therefore Tcl/Tk, is not needed. Upon invocation, LoLA returns the solution in text or graphical format. However, the latter can be completely suppressed, rendering LoLA capable of operating text–only. This mode of operation is well suited to perform automated or repeated tasks or calling algorithms from other applications.

The *text–based mode* is automatically accessed if command–line options are detected, and it is the only available mode if the LoLA executable has been built with the configuration `--withtcltk=no`. The available options are

```
lola -a <algorithm> [-l|-r|-g|-d|-m <file>] [-p <n>] [-n <n>]
[--output=<arguments>]

Options:

  -a : the algorithm to run with the data (see below)
  -l : <file> containing data describing the existing facilities
  -r : <file> containing data describing the restriction(s)
  -g : <file> containing data describing  a (directed or
       undirected) graph
  -d : <file> containing data describing polygonal gauge
       definitions
  -m : <file> containing data describing a cost matrix for
       n-facility problems
  -p : <n> is the value of p for an l_p-norm
  -n : <n> is the number of new facilities for problems on graphs
  --output:
    this option takes a comma-separated list of arguments:
      (no)windowed : (don't) present the solution graphically;
      [file=]<file>: write solution as text into <file>;
    only the right-most argument of each type takes effect
```

Example 1 *(cont.). To solve our planar problem* $1/P/\mathcal{R} = convpoly/l_2^2/\sum$ *using the input files "sample.loc" and "sample_convPoly.res", we need to type*

```
lola -a in_l2sqr_sum -l sample.loc -r sample_convPoly.res
     --output:file=results.txt
```

The results of the optimization will be saved in the file "results.txt".

Example 2 *(cont.). Typing*

```
lola -a N_median_exchange -g sample.gra -n 4
```

will yield the solution for our multi-facility network problem (see Fig. 4).

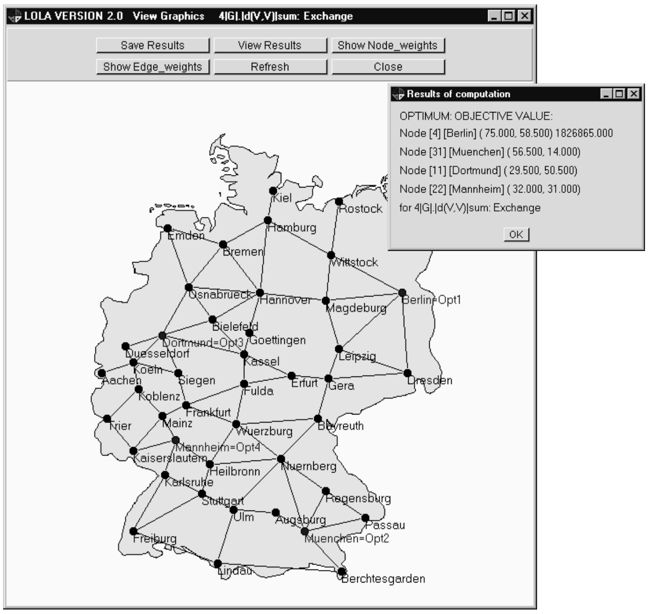

Fig. 4. Windows with the solutions Opt1–Opt4 and the numerical result for the network problem (Example 2)

Programming Interface. All algorithms implemented in LoLA can be accessed via the *Callable Library* interface. Again using our examples, we will briefly explain how the LoLA–libraries can be used directly in a C++ program without making use of the LoLA *Frontend*.

Example 1 *(cont.). First, we need to include the definitions of the routines required to handle location and restriction files (read, write) and the algorithms to solve the planar problem.*

```
#include <LOLA/facs_util.h>
#include <LOLA/planealg.h>

void main() {
  // Variables to store the objective value, the
  // classification string and the names of the input files.
  double objval;
  string class_string;
  string loc_file="sample.loc";
  string restr_file="sample_convPoly.res";

  // List to save the solution points.
  list<location> PlanarSol;
```

```
  // Data structures of LoLA to handle locations,
  // restrictions and the available planar algorithms.
  facs_util FacsUtil;
  planealg PlaneAlg;
  restrictions Restr;

  // Read the input data of the planar problem describing
  // locations, i.e. x-coord, y-coord and their weight.
  ifstream locfile (loc_file);
  FacsUtil.ReadLoc(locfile);

  // Read a restriction file.
  ifstream restrfile (restr_file);
  Restr = ReadRestr(restrfile);

  // Solve the planar problem.
  objval = PlaneAlg.l2sqr_sum(FacsUtil,Restr);

  // The input data including locations, the restrictions
  // and the solution are graphically shown in a window.
  PlanarSol = PlaneAlg.alg_solution();
  class_string = "1/P/R = convex polyhedron/l2sqr/sum";
  FacsUtil.View(objval,class_string,PlanarSol,Restr);
}
```

Example 2 *(cont.). For the network example, the following commands are required.*

```
#include <LOLA/graph_util.h>
#include <LOLA/lgraphalg.h>

void main() {
  // Variables to store the objective value, the
  // classification string and the name of the input file.
  double objval;
  string class_string;
  string graph_file="sample.gra";

  // List to save the solution points.
  list<sol_typ> NetworkSol;

  // Data structures of LoLA to handle network problems;
  // the coordinates and names of the nodes are stored
  // separately in two data structures;
  // for N-facility problems additionally the assignments
```

```
  // of nodes to the solution points are computed.
  facilities GraphFacilities;
  list<string> LocTxt;
  graph_util GraphUtil;
  lgraphalg GraphAlg;
  list<int> BelongTo;

  // Load the input data for the network problem.
  ifstream graphfile (graph_file);
  GraphUtil.ReadGraph(graphfile,GraphFacilities,LocTxt);

  // Solve the network problem and show the solution.
  lolaundirected UGraph = GraphUtil;
  objval = GraphAlg.N_median_exchange(UGraph,4,BelongTo);

  NetworkSol = GraphAlg.alg_solution();
  class_string = "4/G/./d(V,V)/sum";
  GraphUtil.LGraphView(objval,class_string,NetworkSol,
                       GraphFacilities,LocTxt,BelongTo);
}
```

2.4 Format of the Input and Output Data

To solve facility location problems independent of a specific platform, LoLA reads input data and writes the results to ASCII files. LoLA interprets problem data for location problems using a descriptive language specifically designed for that task. There are two possibilities to generate input data files. One is to use any standard ASCII editor. The second is to use the graphical editor attached to LoLA. This tool allows a direct graphical input of data which can be converted into the LoLA data language afterwards.

In the following we describe the input and output formats for planar, restriction and network data files required for solving our example problems. For a complete description we refer to Hamacher et al. (1999). Note that the blanks in the environment specifications as given below cannot be omitted!

Input File Type: planar location data *loc*

`begin {location} [d,Q]`
$x_{11} \cdots x_{1d}$ $w_{11} \cdots w_{1Q}$ [*symbolic name of facility* 1]
$\vdots$ $\vdots$ $\vdots$
$x_{M1} \cdots x_{Md}$ $w_{M1} \cdots w_{MQ}$ [*symbolic name of facility* M]
`end {location}`

d: Dimension of the facilities.
Q: For N–Facilities problems: Q is the dimension of the weights, which in this case is equal to the number N of new locations.

For Q–median or Q–center problems: The number of criteria (objectives) according to which the problem is to be solved.

x_{ij}**:** j–th coordinate of the i–th facility, $i = 1, \ldots, M$, $j = 1, \ldots, d$.

w_{ij}**:** For N–Facilities problems: For each new facility, there must be a weight representing the importance (demand) of this new facility with respect to the existing facilities. Hence, the value w_{ij} represents the weight of the j–th new facility with respect to the i–th existing facility. Note that in the case that $N = 1$, i.e. for 1–facility problems, only one weight w_i has to be specified for each existing facility.

For Q–median or Q–center problems: w_{ij} represents the weight (demand, importance) of the existing facility i with respect to the j–th criterion (objective).

Input File Type: restriction data *res*

In case of planar location problems, restrictions (forbidden regions or barriers for the new locations) can be specified using files of type *res*. Next, only the description of 2-dimensional convex polyhedra is given.

```
begin {restriction} [2]
begin {conpolyhedron} [m]
x1 y1   This environment should be alternatively used in case
x2 y2   of convex polyhedral restrictions.
  ⋮
xn yn
end {conpolyhedron}
end {restriction}
```

m: This number indicates the type of restriction:
0: restriction is a forbidden region (default argument, can be omitted),
inf: restriction is a barrier.

Example 1 *(cont.). The two files "sample.loc" and "sample_konvPoly.res" of the planar example problem are (partly) depicted in Fig. 5 using the buttons* View Location *and* View Restriction*.*

Input File Type: network data *gra*

The input format for network location problems consists of a *location* file and a file representing the adjacency matrix of the network (graph) in an *adjacencylist*. For this purpose, the format *adjlist_byname* is provided where the edges of the corresponding network can be specified using the symbolic names of their starting node (source node) and their end node (target node). The information of the existing facilities and the other nodes of the network is stored using the *location* format. Here the nodes can be specified by their (d–dimensional) coordinates which allows the use of location data from problems of planar type.

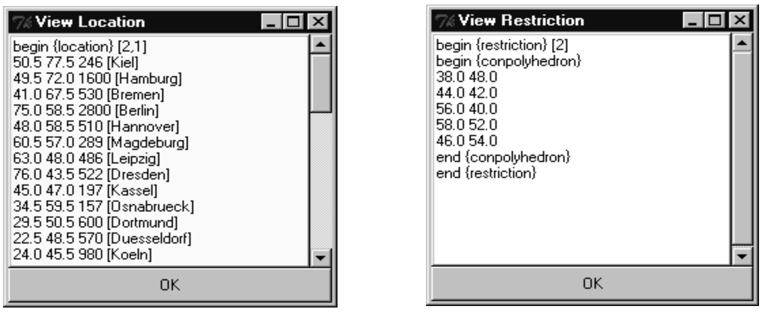

Fig. 5. Location and restriction data files of the planar example

They can be alternatively assigned the attribute NC, i.e. "no coordinates" have to be specified. If desired, the symbolic names of nodes in the adjacency list can be replaced by numbers corresponding to the sequence of the facilities in the location part of the network data file. Nodes of the network not representing an existing facility can be included in this list by setting their weights w_{ij} equal to zero.

`begin {lolagraph}`
`begin {location} [d,Q]`
$x_{11} \cdots x_{1d}$ $w_{11} \cdots w_{1Q}$ [symbolic name of facility 1]
$\vdots$ $\vdots$ $\vdots$
$x_{M1} \cdots x_{Md}$ $w_{M1} \cdots w_{MQ}$ [symbolic name of facility M]
`end {location}`
`begin {adjlist_byname}`
$sourcename_1$ $targetname_1$ ew_1
$\vdots$ $\vdots$
$sourcename_n$ $targetname_n$ ew_n
`end {adjlist_byname}`
`end {lolagraph}`

ew_i: Length of the i–th edge.
$sourcename_i$: Symbolic name of the starting node of the i–th edge in the location file.
$target_i$: Symbolic name of the end node of the i–th edge in the location file.

The information about the solution of a location problem is saved in files of type *sol*. Depending on the type of problem solved, this file may contain different information. In the first two environments, `classification` and `objective value` are self–explaining. In the environments `polygonlist` and `graphpointset`, one or several (as e.g. in the case of multicriteria problems) sets of optimal points/polyhedra can be given.

Output File Type: planar and network results *sol*

```
begin {result}
begin {classification}
classification
end {classification}
begin {objective value}
z z z
end {objective value}
```

For planar problems, the `polygonlist` contains the solution points/polyhedra:

```
begin {polygonlist}
begin {polygon}
x_1 y_1
⋮
x_n y_n
end {polygon}
⋮
begin {polygon}
x_1 y_1
⋮
x_n y_n
end {polygon}
end {polygonlist}
end {result}
```

whereas for the result of a network problem the solution points are given in the list `graphpointset`:

```
begin {graphpointset}
SolutionPoint 1
⋮
SolutionPoint n
end {graphpointset}
end {result}
```

The *SolutionPoints* are described by

Node [i]**:** If the solution point is a node of the network.
Edge [i][j] t**:** If the solution point is on an edge of the underlying graph. i and j denote the source and the target node, respectively, of the edge and t, $0 \leq t \leq 1$, the relative distance of the solution point on the edge from i to j.

Example 2 *(cont.). For our network problem the result file contains the following information:*

begin {result}
begin {objective value}
1.99011e+006
end {objective value}
begin {graphpointset}
Node [13]
Node [39]
Node [33]
Node [23]
end {graphpointset}
end {result}

2.5 Function Reference

The algorithms available in LoLA for solving planar and network facility location problems are listed in Tables 2 and 3. A comma separated list at positions 2 or 3 of the classification scheme of Hamacher and Nickel (1998) indicates the available restrictions for a specific planar problem or on which type of networks the problem can be solved, respectively. Note that γ at position 4 denotes a general gauge. Finally, LoLA can solve discrete problems of the type $\#/D/\bullet/\bullet/\bullet$.

Table 2. Algorithms which are available in LoLA for planar problems

Planar Problems
$1/P/\bullet, \mathcal{R}, \mathcal{R}^c/l_1/\sum$
$N/P/\bullet/l_1/\sum$
$1/P/\bullet/l_1/2-\sum_{par}$
$1L/P/\bullet, \mathcal{R} = convpoly/l_1/\sum$
$1/P/\bullet, B, \mathcal{R} = convpoly/l_2/\sum$
$N/P/\bullet/l_2/\sum$
$1L/P/\bullet, w_i = 1, \mathcal{R} = convpoly/l_2/\sum$
$1/P/\bullet, \mathcal{R}, \mathcal{R}^c/l_2^2/\sum$
$N/P/\bullet/l_2^2/\sum$
$1/P/\bullet/l_2^2/Q-\sum_{par}$
$1/P/\bullet, \mathcal{R} = convpoly/l_p/\sum$
$N/P/\bullet/l_p/\sum$
$1L/P/\bullet, \mathcal{R} = convpoly/l_p/\sum$

$1/P/\bullet,\mathcal{R},\mathcal{R}^c/l_\infty/\sum$
$N/P/\bullet/l_\infty/\sum$
$1/P/\bullet/l_\infty/2-\sum_{par}$
$1L/P/\bullet,\mathcal{R}=convpoly/l_\infty/\sum$
$1/P/\bullet/\gamma/\sum$
$1/P/\bullet/\gamma/2-\sum_{par}$
$1L/P/\bullet,\mathcal{R}=convpoly/\gamma_B/\sum$
$1/P/\bullet,w_i=1,\mathcal{R}=convex/l_1/max$
$N/P/\bullet/l_1/max$
$1L/P/\bullet/l_1/max$
$1/P/\bullet,\mathcal{R}/l_2/max$
$1L/P/\bullet/l_2/max$
$1L/P/\bullet/l_p/max$
$1/P/\bullet,w_i=1,\mathcal{R}=convex/l_\infty/max$
$N/P/\bullet/l_\infty/max$
$1L/P/\bullet/l_\infty/max$
$1/P/\bullet,\mathcal{R}/\gamma/\max$
$1L/P/\bullet/\gamma_B/\max$

Table 3. Algorithms which are available in LoLA for network problems

Network Problems
$1/G_D,G,T/\bullet/d(V,V)/\sum$
$1/G/\bullet/d(V,G)/\sum$
$1/T/\bullet/d(V,T)/\sum$
$1/G/\bullet/d(V,G)/2-\sum_{par}$
$1/G_D,G/\bullet/d(V,V)/Q-\sum_{par}$
$1/G_D,G/\bullet/d(V,G)/Q-\sum_{par}$
$1/G_D,G/\bullet/d(V,V)/Q-\sum_{lex}$
$1/G_D,G/\bullet/d(V,G)/Q-\sum_{lex}$
$1/G_D,G,T/\bullet/d(V,V)/\max$
$1/G_D,G/\bullet/d(V,G)/\max$
$1/T/\bullet/d(V,T)/\max$
$1/G_D/\bullet/d(V,V)/Q-\max_{par}$
$1/G_D/\bullet/d(V,G)/Q-\max_{par}$
$1/G/\bullet/d(V,V)/Q-\max_{lex}$
$1/G_D/\bullet/d(V,V)/Q-\max_{lex}$
$N/G/\bullet/d(V,V)/\sum$
$N/G/\bullet/d(V,V)/\max$

3 LoLA goes GIS

3.1 Introduction

In the preceding section, we introduced LoLA and explained how its input and output functionality is designed. An open question is how to get real world data easily into LoLA to solve real world problems.

Suppose we have a large set of facilities whose locations are saved in a certain database table or something similar. Although LoLA provides different ways of using these data, it cannot read any arbitrary input data file. On one hand, we can create the input file by hand, i.e. copy and paste; and on the other hand we can use the graphical editor of LoLA to create the file. Since both ways are very time consuming and ineffective, we developed a third option which consists of implementing a converter to create the location file from a database table.

This is the point where Geographical Information Systems (GIS) come into play. These systems are designed for visualizing real world data of countries, states, cities, etc. with map data and additional information that allow decision makers to find solutions to problems such as locating facilities, routing, etc. An example of a *Frontend* of such a GIS is given in Fig. 6.

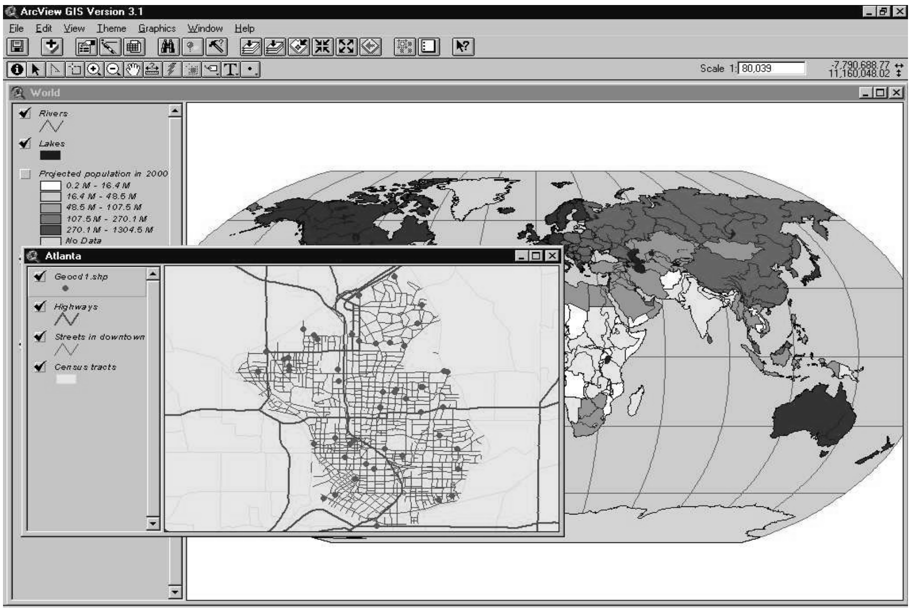

Fig. 6. ArcView GIS *Frontend*

It seems to be a natural idea to link both systems, GIS and LoLA. LoLA needs real world data and an easy way to quickly input large data sets, while the GIS needs more sophisticated algorithms to be able to deal with locational decisions. One of the basic tasks of such a link is to transform data from GIS to LoLA. Additionally, the GIS should be able to call routines from LoLA, extract the solution of the location problem and visualize it in a map with the given data. Figure 7 depicts the scheme of the functionality the link should have.

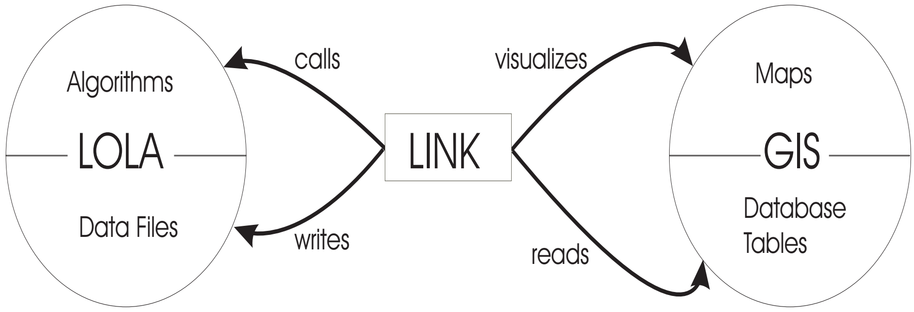

Fig. 7. LoLA–GIS link

An example of a Geographical Information System is ArcView GIS[4]. This GIS provides a large amount of location–relevant data, and new data is easy to add in the form of database tables, text files, etc. The large amount of data in ArcView GIS gives LoLA a large source of real world data. Second, new data can be easily added. This gives us a larger area of acting and solving problems.

A very useful tool of ArcView GIS is the script language which is called Avenue. It is easy to learn, easy to handle and has the ability to control every process of ArcView GIS. This language is what we need to combine ArcView GIS with LoLA.

We have implemented scripts that read data out of the ArcView GIS database, where all the locational data is stored, and convert the data into a LoLA input file. From another Avenue script, we can call LoLA and its routines as a console application, i.e. in the text–based mode; or we can call other executables with our own implementations which are based on LoLA. Since LoLA gives us the possibility to save the solution into a file we can get the solution back into ArcView GIS with another script, where we can visualize it. Hence, ArcView GIS together with Avenue fulfills the necessary conditions for linking it to LoLA; and, additionally, ArcView GIS meets the desired conditions that we mentioned above.

With the help of Avenue, we were also able to create and control menu options so that the user has an interface that allows him to communicate with ArcView GIS to select the algorithms of LoLA.

4 ArcView® GIS is a registered trademark of Environmental Systems Research Institute, Inc. (ESRI), USA.

3.2 Implementation

First of all, we would like to export data from a GIS to LoLA in order to have real world data for our software. Since ArcView GIS was the selected GIS, we have to accept its conventions. The locational data concerning the facilities are stored in tables which we would like either to read from the database of ArcView GIS directly into LoLA or to write into an ASCII file which could be read by LoLA. Avenue gives us the means to accomplish this. Another advantage of Avenue is that it can directly access the coordinate data of the facilities which are hidden by ArcView GIS. With Avenue we can easily access the data base of ArcView GIS and write our own ASCII data files in the input format that LoLA uses. Below you will find the procedure for getting data of one record, for one facility, and writing the corresponding ASCII file:

```
lolaPoint = lolathemetab.GetLabelPoint(rec)
lolafile.WriteElt(lolaPoint.GetX.AsString + " " ++
      lolaPoint.GetY.AsString + " " ++
      lolathemetab.ReturnValue(lola3field, rec).AsString +
      " ["++ lolathemetab.returnValue(lolafield, rec) + "]").
```

The coordinates are written first, then a chosen value as weight and the name of the location. Running this procedure over each record we get the location data written in an ASCII file where only the head and the tail must be added in order to use it in LoLA.

Based on this script that we extended, the scripts for ArcView GIS increase the functionality of the ArcView–LoLA–link (ALL). We added our own control functions to have more control of ALL. In Fig. 8, the added LoLA button and the menu part for LoLA are shown.

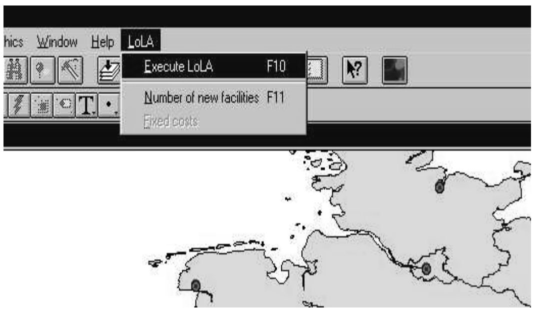

Fig. 8. ArcView GIS–LoLA menu

The menu option *number of new facilities* in Fig. 8 is used to specify the number of new facilities to build, i.e. a single facility, a specified number p for the

p–median or center problem respectively, or to solve the Uncapacitated Facility Location Problem (UFLP) with an unspecified number. The dialog is shown in Fig. 9.

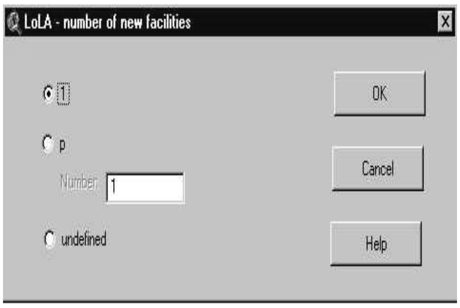

Fig. 9. ArcView GIS–LoLA number of new facilities dialog

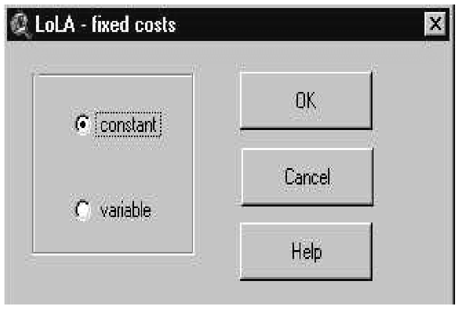

Fig. 10. ArcView GIS–LoLA fixed costs dialog

In the last option *fixed costs* we can define fixed costs for setting up new facilities (see Fig. 10). The option *Call* LoLA will start the process of writing the data file and calling LoLA. The LoLA button runs the same process. Upon starting the process, the dialog shown in Fig. 11 appears. Here the user has three fields to choose from. The first field is used to select a theme as a base for the locations. A theme is a visual representation of some geographic data. Within a theme the complete information of ArcView GIS with respect to a special data field, i.e. cities, countries, streets, etc., is stored. The data for the weights can be chosen in the second field. The dialog takes all numerical data that can can be found in the theme table and displays it. Then the user can choose one field from the list box which uses the data as weights. The last option allows the user to decide on the algorithm that runs with LoLA. This action depends on the number of facilities chosen in the *number* menu option.

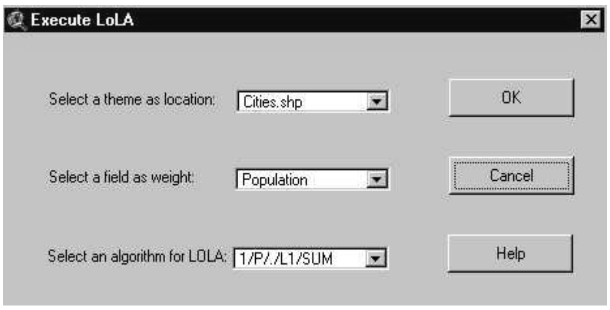

Fig. 11. ArcView GIS–LoLA dialog

For single facility location problems there are in total six algorithms available for solving median and center problems with the distance functions l_1, l_2 and l_∞. For p–median/center problems and for the UFLP, there are three heuristics and an exact algorithm available.

In Fig. 12 the visualization of the result of a p–median problem with 4 new facilities is shown. The new facilities are depicted with large points and the allocation of the existing facilities to the new ones is displayed with straight lines.

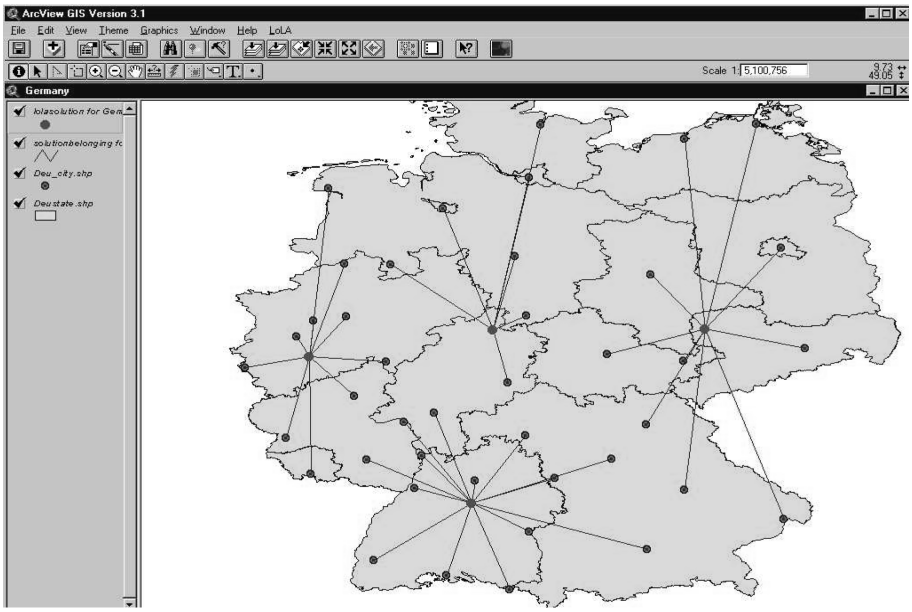

Fig. 12. ArcView GIS–LoLA solution

4 Supply Chain Management

4.1 The Role of Facility Location Planning in Supply Chain Management

A supply chain is a network of facilities (e.g. plants, distribution centers, warehouses) that performs a set of operations ranging from the acquisition of raw materials, the transformation of these materials into intermediate and finished products, to the distribution of the finished goods to the customers. The optimization of the complete supply chain is accomplished through efficient planning decisions. Three levels of planning can be distinguished depending on the time horizon. Strategic planning has a long–range scope and focus on the development of objectives and policies for the supply chain. Tactical planning is a medium–term activity concerned with the means by which the strategic objectives can be realized such as the effective use of existing resources. Finally, operational planning involves short–term decisions that deal with the efficient operation of daily activities.

The strategic design of a supply chain involves determining

- the number, location and capacity of manufacturing plants, distribution centers, warehouses, etc. that should be used;
- the number and location of existing service facilities that should be maintained;
- the set of suppliers that should be selected and the amount of raw materials that should be procured;
- the amount of intermediate and finished products that should be manufactured at each plant;
- the transportation channels that should be used to ship the products between the facilities.

Economy globalization along with rapid changes in technology and increasing competition in the business environment are forcing organizations to focus on and invest in their supply chains to quickly respond to customer needs. Well–planned strategic decisions enable the efficient flow of materials through the logistics system and lead to decreased costs and improved customer service (Bramel and Simchi–Levi 1997).

Among the strategic issues listed above, facility location decisions have a key role in strategic planning for a wide range of organizations. The set up of a new facility is typically a long–term project involving high costs (e.g. property acquisition, facility construction) with a strong impact on various tactical and operational decisions. Therefore, facilities which are located today are expected to remain in operation for an extended period of time and perform well even as market trends and system conditions change.

4.2 Facility Location Models for Strategic Supply Chain Planning

As mentioned in the previous section, decisions concerning facility location have a major impact on the strategic design of supply chains. Although the term *supply chain management* is relatively new, the development of facility location models for the strategic design of production–distribution networks has received considerable attention in the last decades. In particular, emphasis has been put on Mixed Integer Programming (MIP) formulations. Aikens (1985) reviews some relevant MIP formulations for production–distribution systems. One of the earliest contributions in this area is dedicated to a multi–commodity problem comprising several plants with known capacities, distribution centers (DCs), and a number of customer zones (Geoffrion and Graves 1974). Potential site locations for new DCs are known, but the particular sites to be used are selected based on minimizing total network costs. These costs consist of fixed charges for setting up a DC, variable operating costs (based on the amount shipped through a DC), and production and transportation costs for shipping products from a plant to a customer zone via a DC. The constraints in the model are related to capacity limits at the plants, customer demand satisfaction, single sourcing by customer zones and bounds on the throughput at each DC. A solution procedure based on Benders decomposition is proposed. A refined version of the model addressing real world problems was later developed by Geoffrion et al. (1978). The extended model considers single sourcing of customer zone by product type, nonlinear facility throughput constraints and trade–offs between distribution and customer service. The optimization procedure for solving this large scale MIP problem is again based on Benders decomposition.

In many practical situations, the selection of a potentially new facility is linked with the decision about the type of equipment that facility should use to handle a particular commodity. Hence, apart from the fixed charge for opening a facility, an additional fixed cost is incurred if an open facility is equipped to handle a given product or has a certain level of capacity installed (e.g. small, medium and large). Brown et al. (1987) considered such a generalization of the multi–product capacitated facility location problem. Their MIP formulation addresses the opening and closing of plants, the assignment of equipment to plants, and the commodities produced at each plant and delivered directly to customer zones. Variable production and transportation costs, fixed costs of equipment assignment, and fixed costs of plant operations are included in the objective function. The constraints considered in the model are comprised of customer demand satisfaction, maximum number of equipment assigned to each plant, single sourcing of equipment to plants, and upper bounds on commodities produced on each equipment at each plant. The problem is solved by applying a decomposition principle similar to that developed by Geoffrion and Graves (1974). Recently, Lee (1993), and Mazzola and Neebe (1999) addressed a similar problem in a 2–echelon context in which plant and customer locations were fixed and facility location decisions were restricted to a single echelon of DCs. Solution procedures developed for this problem use cross decomposition (Lee 1993) and Lagrangian relaxation (Mazzola and Neebe 1999).

Another type of problem that has practical relevance in industrial settings refers to the simultaneous selection of site locations for plants and DCs in a multi–commodity capacitated production–distribution system. Pirkul and Jayaraman (1996) apply Lagrangian relaxation to an MIP formulation of the problem and present a heuristic procedure for solving it. The objective function of the model minimizes the sum of the fixed cost of establishing and operating the plants and the DCs along with the variable cost of transporting products from the plants to the DCs and distributing the commodities from the DCs to the customers in order to satisfy the demands of the latter. Both plants and DCs have limited capacity.

The long–term nature of facility location decisions involves planning the operation of facilities in such a way that they can cope well with an uncertain future environment. Therefore, robust location decisions are of particular importance. For the case of incomplete information regarding future customer demands, Carrizosa and Nickel (1998) propose a model and a solution technique for solving a single facility continuous location problem. Robust decisions should also consider the dynamic aspect of facility location with respect to the timing of facility expansions and relocations during the whole planning horizon. While the literature on static facility location in production–distribution systems is quite extensive, the dynamic version of the problem has received considerably less attention. Owen and Daskin (1998) present a review on relevant MIP formulations in this area. A recent contribution dedicated to a 2–echelon multi–commodity capacitated facility location problem considers opening and closing both plants and DCs over a given number of time periods (Hinojosa et al. 2000). A Lagrangean relaxation scheme incorporating a heuristic procedure is proposed for solving the problem.

Due to the globalization of the economy and the emergence of global logistics, the development of models for the strategic design of international production–distribution systems has gained increasing importance. Such models address global features common to an international scenario in which the business activities of a firm are diversified among multiple countries. Critical issues in the strategic design of a global supply chain concern, for example, taxes and duties, exchange rates, trade barriers and government stability. Verter and Dincer (1995) present a literature review on analytical models for facility location in global supply chains. The authors observe that most existing models focus on the optimization of location and allocation decisions and neglect the interaction between financial and location decisions.

A number of case studies describing the application of facility location models to the strategic design of real life supply chains have been reported in the past decade, showing the growing awareness and importance that practitioners are devoting to this area. The rapid evolution of computer and communications technology has made possible the optimization of facility location decisions in real–world production–distribution systems. General–purpose mathematical programming software has become available to solve problems of realistic size. Examples have been reported by Breitman and Lucas (1987), Van Roy (1989), Martin et al. (1993), Pooley (1994), Camm et al. (1997), and Köksalan and Süral (1999) for various industry branches. In some cases, however, the problem size and complexity along with the management's wish to obtain "good" solutions in reasonable time have driven researchers to develop heuristic solution procedures. For example, this was the case for the p–median model proposed by Erkut et al. (2000) for an energy company. The model was solved using the greedy heuristic of Teitz and Bart (1968). To display the coverage areas of each selected facility, the solution generated by the heuristic is displayed in a geographical information system.

In practical situations, the analysis of a supply chain usually starts by examining how well the existing operations are being run with a view to optimizing the number and location of facilities under a given set of constraints, see e.g.

Pooley (1994). Since data are usually not accurate enough, one may be interested in getting an indication of the approximate region where it would be sensible to site new facilities. Hence, instead of formulating a problem as an MIP, one may use a continuous approach (see Chap. 1 for a comprehensive review of continuous facility location models).

4.3 Supply Chain Planning with the SAP Advanced Planner and Optimizer

The SAP Advanced Planner and Optimizer[5] (SAP APO) is an integrated software application for supply chain planning (Bartsch and Bickenbach 2000). It is part of the SAP Supply Chain Management Solution (SAP AG, Germany 2000) which combines the Enterprise Resource Planning (ERP) system SAP R/3 with an advanced supply chain planning tool offering integrated data and transactions for the management of the entire supply chain. It consists of a series of application tools for decision support ranging from the long–term strategic to the short–term operational planning. All applications use a consistent basis of master and transactional data. Data integration is offered to execution systems like SAP R/3 or Non–SAP systems. In addition, the connection to internet technologies enables enterprises the collaborative planning of their logistics activities. SAP APO comprises a series of advanced optimization techniques and algorithms combined with a high performance memory–resident object management technology.

Figure 13 gives an overview of the main planning applications of SAP APO. Next, a brief description of each application is presented.

Supply Chain Cockpit: A graphical instrument panel for monitoring and controlling the supply chain.

Network Design: Strategic planning module to analyze and optimize the entire supply chain. Facility location decisions are supported as well as long–term sourcing, production and distribution decisions.

Demand Planning: A toolkit of statistical forecasting techniques and demand planning features for generating accurate forecasts.

Supply Network Planning: A planning module to create tactical production plans and sourcing decisions that considers the complete supply network and its constraints.

Procurement Planning: Workplace for the procurement of products, using optimization techniques and linked with collaboration processes.

[5] SAP Advanced Planner and Optimizer^{TM} is a registered trademark of SAP AG, Germany.

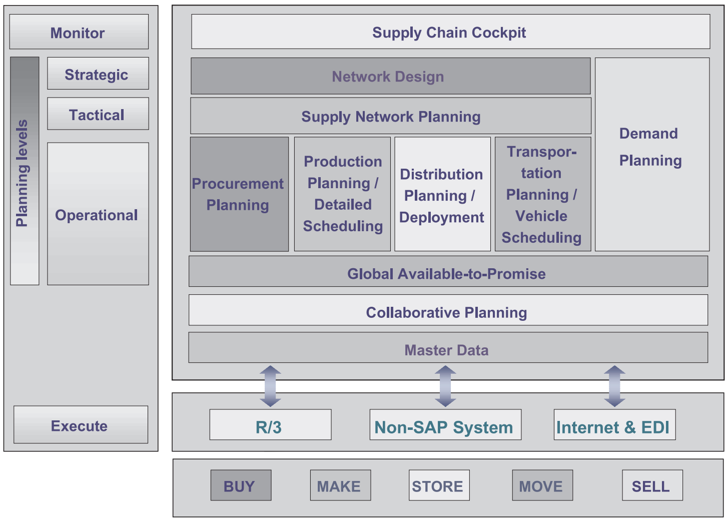

Fig. 13. The SAP APO architecture

Production Planning/Detailed Scheduling: Production planning tool using dynamic pegging and optimization techniques to generate executable plans. In addition, optimal production schedules are created for real–time scheduling for finite sequencing and final assignment of production resources.

Distribution Planning/Deployment: Module for establishing shipment plans for all products among the different facilities in the supply chain.

Transportation Planning/Vehicle Scheduling: A toolkit for determining optimal truck loads, freight consolidation and carrier selection based on shipment plans as well as individual orders, considering constraints for route determination and vehicle scheduling.

Global Available–to–Promise: An application for checking product availability which considers allocations, production, transportation capacities and costs along the entire supply chain.

Collaborative Planning: Using internet technologies enterprises are enabled to collaboratively plan supply chain activities with their business partners ranging from forecasting to shipment planning.

4.4 SAP APO Network Design

SAP APO Network Design[6] is a decision support tool which assists management in the analysis and redesign of supply chains. In particular, the tool supports strategic decision–making regarding the number, location and capacities of service facilities as well as the flow of products through the logistics network so as to minimize total costs. Both continuous and discrete mathematical models to be used under different levels of data availability are included in the tool. Typical decisions supported by SAP APO Network Design include those listed in Sect. 4.1 which arise in most production–distribution network systems.

SAP APO Network Design is closely linked with the Demand Planning and Supply Network Planning modules described in the previous section. Furthermore, it makes full use of the integrated basis of master data within SAP APO. Cost data required by Network Design can be obtained using the SAP Strategic Enterprise Management (SEM) module which allows the planner to create business models and to analyze financial plans, e.g. for the calculation of facility location costs taking taxes, interest rates and capital charges into account. Finally, an integration of the SAP Business Information Warehouse (BW) is available in which results obtained with Network Design can be explored more deeply using advanced query and reporting functions. Figure 14 illustrates the different components within the SAP APO system that are directly connected with the Network Design module. Furthermore, the integration with other SAP applications is shown.

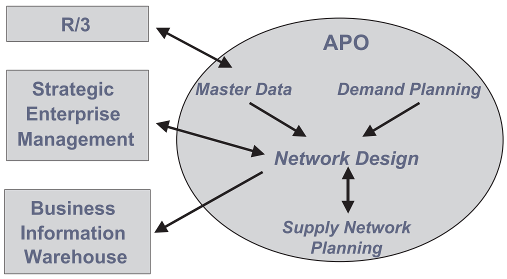

Fig. 14. Integration of the SAP APO Network Design system with other SAP applications

4.5 SAP APO Network Design Planning Algorithms

In industrial settings, the optimization of location and allocation decisions is often preceded by an evaluation of the structure of the already existing production–distribution network. Such an evaluation involves assessing the quality of the

[6] SAP® APO Network Design is a registered trademark of SAP AG, Germany.

current locations of the service facilities, the degree to which their capacities are being well used and the allocation of customer demands to these facilities. SAP APO Network Design offers several tools for obtaining appropriate configurations of the supply chain. Both continuous and discrete models are integrated in SAP APO Network Design to support planning decisions under different levels of data availability (Kalcsics et al. 1999). If detailed data on the structure of the supply chain and its associated costs are known, then a discrete planning model should be employed, otherwise a continuous procedure is sufficient.

Upon exploring the configuration of the existing network, the decision maker can use additional tools to improve the network performance. These focus on the redesign of the network structure with respect to facility location, procurement, production and distribution decisions. Similar to the evaluation step, SAP APO Network Design offers tools for different levels of data aggregation in the redesign phase.

Analysis of a Production–Distribution Network System. To roughly assess the extent to which an operating supply chain is being well utilized with respect to shipping patterns and capacity utilization, SAP APO Network Design offers a quick diagnosis tool with low data requirements. The tool applies to the situation in which several facilities serve a set of geographically dispersed customer zones with known demands for a variety of products under the following simplifying assumptions: (i) all service facilities are identical; (ii) the price of a particular product is the same at every facility; (iii) the cost of acquiring the product is equal to the price plus the transportation cost from the facility; (iv) the transportation cost equals the product of distance traveled and a fixed price per unit distance; (v) each customer wishes to minimize the cost of acquiring the product.

These assumptions are often not completely satisfied in practice. However, our interest lies in the geometric interpretation of the assignment of customer zones to supplying facilities and, thus, a rough approximation of the catchment or trading areas of the facilities suffices. The above assumptions induce a subdivision of the total area under consideration into regions – the catchment areas of the facilities – such that all customers located in the same region are served by the same facility. This implies that the catchment area of a given site consists of all points for which that site is closer than any other site. For example, the catchment areas of M service facilities are determined by generating a *Voronoi diagram* which subdivides the plane into M polygonal regions, one for each facility. Customer locations within a Voronoi polygon are assigned to the facility in that cell. As a result, the size of a facility corresponds to the total amount of products shipped from that facility.

For a small example consisting of two manufacturing plants, two DCs, eleven customer zones and one product type, Fig. 15 shows the corresponding Voronoi assignment of customers to DCs and DCs to plants as displayed in the SAP APO Network Design application. The dottedd line separates the catchment area of the plant in Cheyenne from the catchment area of the plant in Chicago, while the solid line divides the areas served by the DCs located in Salt Lake City and Cincinnati.

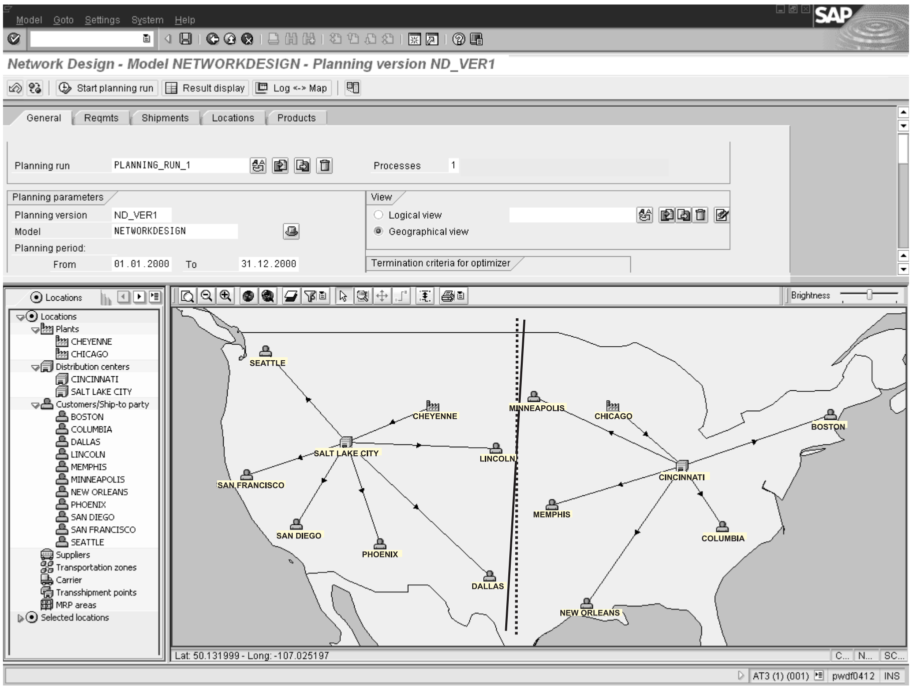

Fig. 15. The Voronoi assignment of customers to DCs (solid line) and DCs to plants (dotted line) as displayed in the SAP APO Network Design application

In mathematical terms, the Voronoi polygon associated with the ith facility is defined by

$$V_i = \bigcap_{1 \le k \le M : k \ne i} \left\{ p \in \mathrm{I\!R}^2 : d(p, Ex_i) < d(p, Ex_k) \right\}, \qquad i = 1, \ldots, M$$

with Ex_i denoting the location in the plane of the ith facility and $d(p, Ex_i)$ denoting the distance between a point p and Ex_i. Distances in the plane are measured by the Euclidean norm and, as a result, the Voronoi diagram is computed with a sweep line algorithm which runs in $O(M \log M)$ time (Fortune 1987). The Voronoi diagram has applications in many fields such as physics, robotics and facility location (Okabe et al. 2000).

The Voronoi assignment model provides a quick analysis of possible deviations to the current network configuration regarding demand allocation and capacity utilization at the existing facilities. In addition, the clustering of customers into regions such that each region is served by a single facility provides useful information for developing tactical and operational plans regarding the fleet of vehicles required to deliver the goods and the routes traveled by them. Another major advantage of the tool is that it helps to create a clear structure in the supply chain with very low data requirements.

If detailed data are available, for example, on the actual transportation channels for shipping products between facilities and on the corresponding transportation costs, then the optimal flow of products through the supply chain is determined by solving a linear programming problem. The SAP APO Network Design application uses the mathematical programming software CPLEX[7] (2000) to solve this problem. Various features can be incorporated into the model that describe, for instance, production activities, resource consumption and capacity requirements. The goal is to determine the amount of raw materials to be procured, the production levels for each product at each manufacturing facility, and the transportation flows between facilities in such a way that total procurement, production, handling, storage, operating and transportation costs are minimized.

Both the Voronoi assignment model and the linear programming model serve exploratory purposes by providing different network configurations that can be compared to the existing supply chain. Furthermore, they assist in developing insight into improvement areas. For example, management may consider the current set of facilities inappropriate due to changing demand patterns or the termination of a leasing contract for certain facilities. As a result, the production–distribution network needs to be redesigned. Clearly, this may lead to a selection of new suppliers, a change in production levels, and generally to a new flow pattern of goods throughout the whole network.

Redesign of a Production–Distribution Network System. The redesign of a supply chain is supported by SAP APO Network Design through continuous and discrete models.

To solve a *conditional* location problem in which a number of facilities (say M) are already in place and the decision maker is considering the construction of some additional facilities (say N) without altering the positions of the existing facilities, SAP APO Network Design offers a geometric tool based on the Voronoi diagram technique. The major advantage of this approach is that it has very low data requirements which is a common situation in the strategic design of supply chains. Due to the long–term nature of facility location decisions, at the beginning of the planning horizon there is a considerable amount of uncertainty regarding relevant parameters that influence location decisions. Costs (e.g. facility, transportation) and demands (e.g. quantity, location) are examples of such parameters. Therefore, in such situations it is advisable to first use a continuous model and later apply a discrete model when detailed information becomes available.

The heuristic procedure developed for solving the conditional facility location problem considers the placement of the new facilities sequentially. Figure 16

---

[7] CPLEX^{TM} is a registered trademark of ILOG, Inc.

summarizes the main steps required to locate the kth new facility given that $k-1$ new facilities were previously placed ($1 \leq k \leq N$).

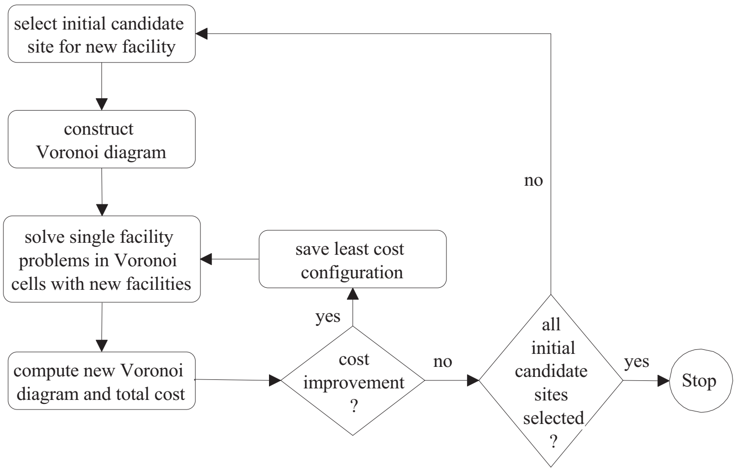

Fig. 16. Heuristic procedure for locating a new facility

Based on the selection of an initial candidate site for the new facility, the plane is partitioned into sectors by constructing a Voronoi diagram using the $M+k$ facilities as the generator set. Next, the initial location of the kth facility is refined by determining the optimal site within its Voronoi cell. Moreover, improved positions for the first $k-1$ new facilities are also sought within their Voronoi cells. This is accomplished by solving a 1–median problem with the Weiszfeld algorithm (Love et al. 1988) in each cell containing a new facility. Upon obtaining new positions for all or some of the first k facilities, the network is reconfigured through the generation of a new Voronoi diagram. The entire process is repeated until the cost difference between two consecutive network configurations is less than a given tolerance. The cost of a solution is determined by summing up the system–wide costs which may range from procurement, production, handling, transportation and operating costs to fixed charges for opening new facilities. As soon as no cost improvement is obtained, another initial candidate site is selected for the kth new facility and the procedure is restarted. After exploring all initial candidate sites for the kth facility, the algorithm proceeds to find a suitable location for the $(k+1)$th facility using as a starting network structure the best configuration obtained so far. This process is repeated until the total desired number of new facilities has been located.

Based on a starting network configuration, six different initial candidate sites for a new facility are considered. The first set comprises Voronoi intersection nodes. Points are selected according to their weight which is measured by the sum of the demands of customers located in the cells adjacent to the points.

The three points with the largest weights are selected. The second candidate set contains the Weber points of the three Voronoi cells with the largest customer demands.

For production–distribution systems with two echelons of facilities in which products are shipped from the first echelon (e.g. plants) to the customers via an intermediate second echelon (e.g. DCs), the procedure outlined in Fig. 16 can also be applied. In this case, it is desired to expand the second echelon by opening a given number of new facilities. From a transportation viewpoint, this raises the question of balancing the advantages of nearness to customers against nearness to plants. To deal with this problem, first a Voronoi diagram is constructed using the plant locations as the generator set. With the partition of the plane thus obtained, the assignment of DCs to plants is straightforward. Next, the algorithm shown in Fig. 16 is applied. While solving the 1–median problem in each Voronoi cell containing the kth new DC, the attraction effect incurred by the plant to which the DC is assigned is taken into account by including the distance from any point in the cell to the plant, weighed by

$$\max\left(\max_{m\in V_k} w_m, \frac{\sum_{m\in V_k} w_m}{2}\right)$$

with V_k denoting the Voronoi polygon associated with the kth new facility, and w_m denoting the total demand of the mth customer in V_k. From the numerical tests performed with different weights, the above choice yielded on average the best results.

The above described algorithm can easily be adapted to the situation in which management desires to open new facilities but does not know the exact number. In this case, a site–independent fixed cost for locating a new facility is considered, and the most economical number of new facilities as well as their locations are determined in such a way that total costs are minimized. The decision maker can also specify a minimum and/or maximum number of new facilities to be located, and the heuristic procedure determines the extended network configuration with lowest total costs. For the small example depicted in Fig. 15 with an additional requisite for locating one new facility, Fig. 17 displays the network configuration and the associated customer–to–DC and DC–to–plant assignments obtained by SAP APO Network Design.

Often, data are not accurate enough to determine positions of new facilities exactly. Hence, the purpose of the continuous facility location algorithms implemented in SAP APO Network Design is to give the decision maker an indication of the approximate regions where it would be sensible to site new facilities. In the case that detailed data become available, facility location can be performed using a so–called discrete approach in which a finite set of alternative sites for the new facilities is defined beforehand, and the problem is to choose from these alternatives the best subset to satisfy demands. Furthermore, decisions regarding whether given existing facilities should continue to be operated or should be closed are also supported by the discrete model.

The advantage of this approach is that it can take explicit account of many items. These include fixed costs for setting up a new facility or closing an exist-

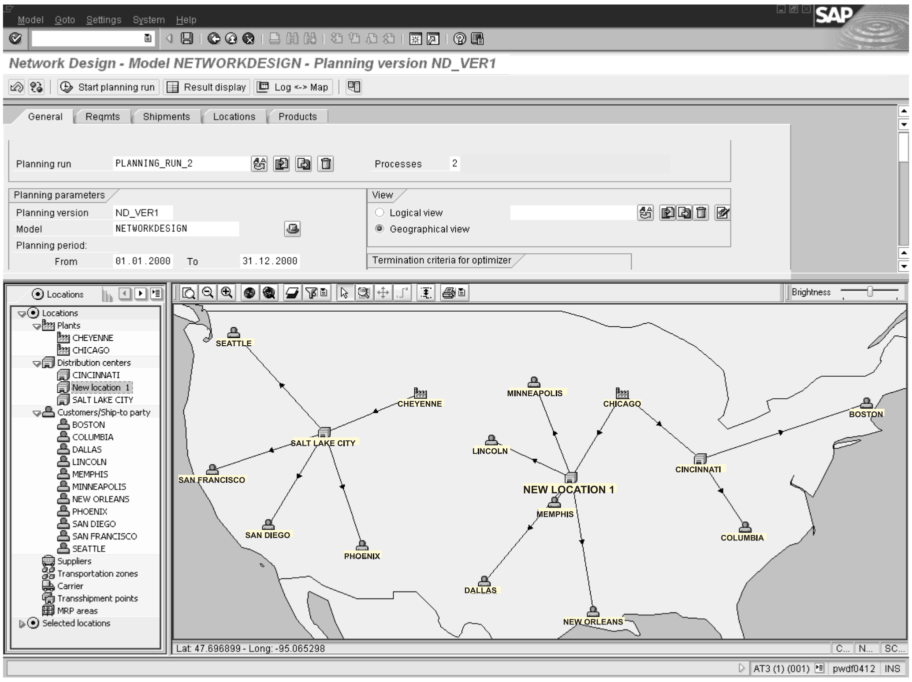

Fig. 17. Location of the new facility as displayed in the SAP APO Network Design application

ing facility, procurement and production activities, actual transportation links for shipping intermediate and finished products throughout the network and the corresponding costs. In addition, production and handling resources with limited capacity can be considered, and the extension of the normally available amounts through, for example, overtime work at the expense of extra costs can also be modeled. The decision maker is also given the possibility to allow external demand not to be completely met. In the case that part or all of the demand of a customer for a given product is not satisfied, a penalty cost is charged. Observe that when resources are scarce, allowing demand to be partially satisfied ensures feasibility of the corresponding capacitated problem. The aim is to decide on the set of facilities to operate, the amount of products to procure, the amount of products to manufacture and the flow of products throughout the network so as to minimize system–wide costs ranging from procurement, production, distribution, handling, operating and transportation to fixed charges for opening or closing facilities. The resulting model is a large scale MIP problem which is solved with the commercial package CPLEX. Again using the small example of Fig. 15 and defining the set of alternative sites with the existing DCs in Cincinnati and Salt Lake City and the potential new DCs in Kansas City and Nashville, the optimized network configuration is shown in Fig. 18. Observe that only one of the potential new DCs is selected, namely Kansas City, and no existing DC is closed.

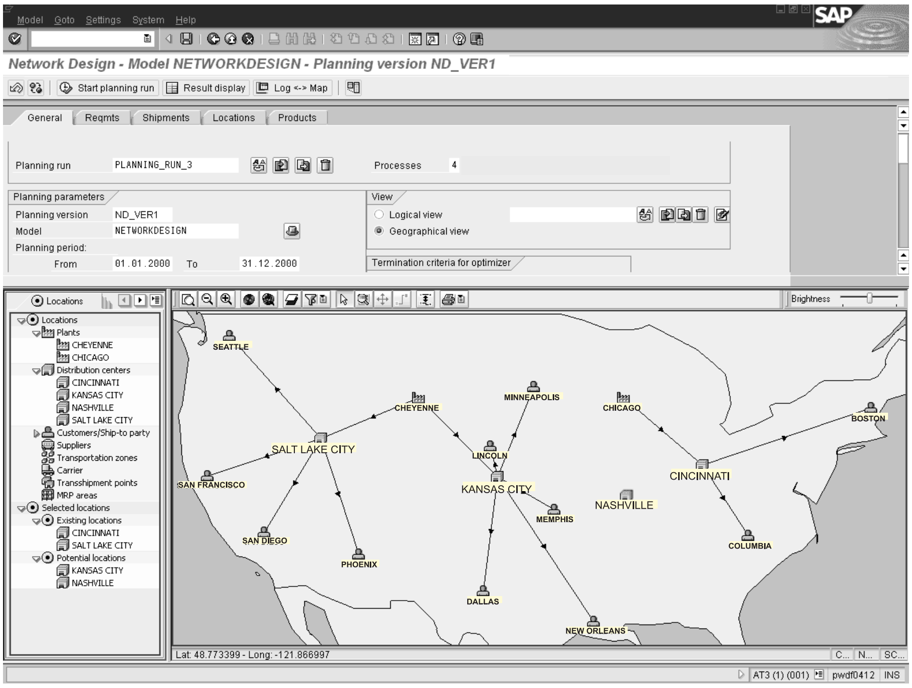

Fig. 18. Solution of the discrete facility location problem

4.6 Planning with SAP APO Network Design

Before using SAP APO Network Design, the decision maker needs to define a model and a set of parameters. The model describes the supply chain and consists of master data objects such as locations, products, resources, production process models, transportation lanes and product demands. In general, the planning procedure consists of the following steps:

- setting up a planning scenario;
- performing the planning run;
- analyzing the planning results.

The decision maker can define several planning scenarios describing different conditions in the supply chain such as various trends in product demands. The results of the corresponding planning runs can be saved to be later compared.

Input data and parameters. To set up a planning scenario, the following data need to be specified.

Locations define facilities such as production plants, distribution centers and customers. Operating costs as well as fixed costs for opening a new facility or closing down an existing facility can be specified.

Products describe raw materials, intermediate and finished products. Specific product parameters can be defined in each facility where the product exists. These include variable storage costs and handling costs for the receipt and the issue of the product. Delivery time quotas can also be set to model service times for product delivery. Costs for late delivery apply in the case that transportation duration is longer than the required delivery time.

Transportation lanes define the transportation links that are available for shipping the products between the facilities in the supply chain. Variable transportation costs and times can be specified for different transportation modes (e.g. rail, truck).

Production process models (PPM) describe the bill of materials in a manufacturing facility. Each PPM comprises a list of material requirements for a given output product. Material consumption factors as well as variable production costs can be indicated.

Resources can be specified for production processes and handling activities. In addition, resource consumption factors per product and facility as well as total resource capacities are defined. The availability of a given resource can be extended beyond its normal capacity at the expense of some extra variable cost.

Demand can be specified for different products in various facilities. Demand plans can be obtained by using the SAP APO Demand Planning module which generates forecasts and aggregated demand quantities. To support different planning scenarios such as worst case and best case demand trends, demand quantities can be changed by using a factor which automatically increases or decreases all product demands. Variable penalty costs for non–delivery of customer demands can also be defined.

In addition to the above required data for describing a supply chain model, a planning period needs to be indicated. Furthermore, it is possible to create different planning scenarios by selecting only some of the objects of a given model. The cost structure of a scenario can also be easily modified by specifying several factors which allow to represent different cost situations.

Finally, in the discrete optimization models the decision maker can set the maximum desired runtime of an algorithm, and define the maximum desired deviation from the optimal solution. When a relative optimality tolerance is specified, the optimization terminates as soon as a feasible solution is found such that the difference between the value of that solution and that of the best lower bound, divided by the solution value, falls below the user defined tolerance. Both parameters, time limit and relative gap tolerance, enable the planner to obtain good solutions when the available runtime is restricted and/or the problem to be solved is very complex.

Performing Planning Runs. All algorithms can be started interactively from the Network Design application. The planning runs can also be executed as batch jobs, thus enabling the user to schedule a set of operations in advance. Upon termination of a planning run, the results are saved and can be retrieved later for comparison with other scenarios. Hence, the planner can analyze situations modeled with different data, e.g. worst and best case demand scenarios, various cost structures and a selection of different facilities.

Planning Results. The results of a planning run are displayed on a geographical map and also in tabular format. The results include the total costs of the planning scenario with all costs being differentiated by cost type (for example, overall production costs, costs for operating the selected facilities), and the number of selected and non–selected facilities. Moreover, the total costs per facility are specified along with the corresponding new geographical position in the case that one of the continuous facility location algorithms was run. Information regarding transportation lanes indicates transportation quantities and transportation costs as well as the transportation duration and the resulting penalty costs for late delivery. Product specific data in each facility include production, handling, storage and procurement quantities, and the corresponding costs. Data concerning the products can be aggregated by product type or by facility to enable the analysis, for example, of overall quantities of products at all facilities. Results regarding production processes give information on production quantities and the capacity utilization of the corresponding resources. In this way, the decision maker is able to see long–term capacity requirements.

For the solution displayed in Fig. 18, the different levels of detail that are available in the SAP APO Network Design application are shown in Fig. 19.

To compare different planning scenarios, the decision maker carries out various planning runs. This supports the analysis of what–if scenarios or best case and worst case calculations in order to find an overall best planning decision. All results can be downloaded to spreadsheet software like Microsoft Excel[8]. Results obtained with SAP APO Network Design can also be explored more deeply using the SAP Business Information Warehouse. It is possible to use maps to display a planning decision and show the selected locations, product quantities and costs assigned to them.

[8] Microsoft® Excel is a registered trademark of Microsoft Corporation, USA.

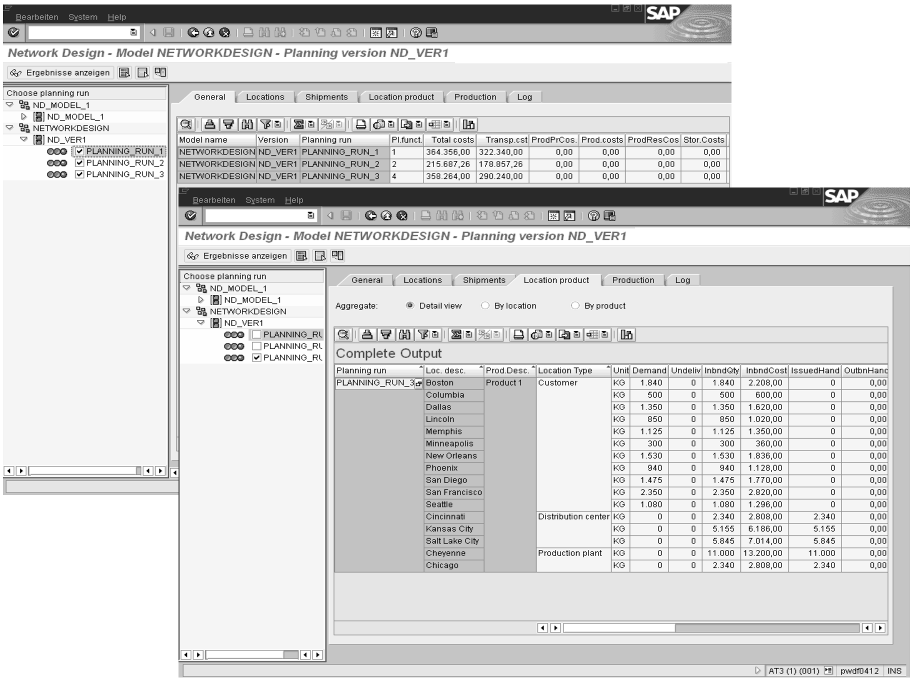

Fig. 19. Details of the solution shown in Fig. 18

5 Outlook

In this paper, we have shown how location software can successfully be developed by taking into account the different needs of the various users of such software. However, developing software is an ongoing process; and, therefore, a number of extensions are planned for the future. In the LoLA project, we intend to replace some of the software components used so far by more modern components. Until now, the LoLA *Frontend* has been completely programmed in Tcl/Tk. Replacing this by a *Frontend* developed with JAVA[9] would allow us to bring the LoLA *Frontend* to the World Wide Web. Consequently, it would be possible to run LoLA inside a web browser, and the user could test (and use) LoLA without having to install the program on his local machine. Furthermore, all user data would be locally available.

We are also testing several graph editing programs and graph drawing tools to link with LoLA, and these would replace the currently available simple editor. Moreover, we intend to substitute the data structures of LoLA, which were implemented using the shareware tool LEDA, with the freeware C++ Standard

[9] JAVA® is a registered trademark of Sun Microsystems, Inc., USA.

Template Library (STL). STL is a powerful library containing basic data structures and algorithms.

With respect to the LoLA–GIS integration, many improvements still need to be carried out. To increase speed, a completely memory–resident data exchange will be implemented. In addition, more LoLA algorithms, such as those dealing with forbidden regions, should be added to the LoLA–GIS interface. There are also plans for a tight integration of GIS, optimization routines and data mining by using a common database.

In the future, the above mentioned extensions of LoLA and the LoLA–GIS interface will continue to be available as an open source software. Hence, every potential developer will have access to the complete set of source codes. This will help us to provide the best possible quality to the location community and will guarantee an ever continuing development. We would also like to establish more interdisciplinary projects using the current available code basis of LoLA to speed up the information exchange within the location community.

For the commercial applications described in Sect. 4, we are mainly concerned with adding relevant functionality to the software. This includes developing heuristic procedures for the discrete facility location problems already covered by the software. In addition, location–routing models, multi–period and multi–criteria facility location models will also be subject of attention.

References

Aikens, C. H.: Facility location models for distribution planning. *European Journal of Operational Research* **22** (1985) 263–279

Bartsch, H., Bickenbach, P.: *Supply Chain Management mit SAP APO: Supply-Chain-Modelle mit dem Advanced Planner & Optimizer 3.0.* Bonn: Galileo Press (2001) (in German)

Berkelaar, M.: *Mixed Integer Linear Program Solver, Version 2.3.* ftp://ftp.es.ele.tue.nl/pub/lp_solve/ (1995)

Bramel, J., Simchi-Levi, D.: *The logic of logistics: Theory, algorithms and applications for logistics management.* New York: Springer-Verlag (1997)

Brandeau, M. L., Chiu, S. S.: An overview of representative problems in location research. *Management Science* **35** (1989) 645–674

Breitman, R. L., Lucas, J. M.: PLANETS: A modeling system for business planning. *Interfaces* **17** (1987) 94–106

Brown, G. G., Graves, G. W., Honczarenko, M. D.: Design and operation of a multicommodity production/distribution system using primal goal decomposition. *Management Science* **33** (1987) 1469–1480

Camm, J. D., Chorman, T. E., Dill, F. A., Evans, J. R., Sweeney, D. J., Wegryn, G. W.: Blending OR/MS, Judgment, and GIS: Restructuring P&G's supply chain. *Interfaces* **27** (1997) 128–142

Carrizosa, E. J., Conde, E., Muñoz, M., Puerto, J.: The generalized Weber problem with expected distances. *RAIRO* **29** (1995) 35–57

Carrizosa, E., Nickel, S.: Locating a robust facility. *Operations Research Proceedings* (1998) 532–540

CPLEX: *Reference Manual, Version 6.5.* ILOG, Inc., Incline Village, Nevada. http://www.cplex.com (2000)

Eiselt, H. A., Laporte, G., Thisse, J.-F.: Competitive location models: A framework and bibliography. *Transportation Science* **27** (1993) 44–54

Erkut, E., Myroon, T., Strangway, K.: TransAlta redesigns its service-delivery network. *Interfaces* **30** (2000) 54–69

EWGLA: Homepage of the European Working Group on Locational Analysis. http://www.vub.ac.be/EWGLA (2000)

Fortune, S. J.: A sweepline algorithm for Voronoi diagrams. *Algorithmica* **2** (1987) 153–174

Geoffrion, A. M., Graves, G. W.: Multicommodity distribution system design by Bender's decomposition. *Management Science* **20** (1974) 822–844

Geoffrion, A. M., Graves, G. W., Lee, S. J.: Strategic distribution system planning: A status report. In *Studies in Operations Management*. Hax, A. C. (ed.). Amsterdam: North-Holland (1978) 179–204

Hamacher, H. W., Hennes, H., Nickel, S.: *LoLA – Library of Location Algorithms, Version 2.0, University of Kaiserslautern*. http://www.mathematik.uni-kl.de/~lola (1999)

Hamacher, H. W., Nickel, S.: Classification of location models. *Location Science* **6** (1998) 229–242

Handler, G. Y., Mirchandani, P. B.: *Location on networks: Theory and algorithms*. MIT Press, Cambridge (1979)

Hinojosa, Y., Puerto, J., Fernandez, F. R.: A multiperiod two-echelon multicommodity capacitated plant location problem. *European Journal of Operational Research* **123** (2000) 271–291

Kalcsics, J., Melo, T., Nickel, S., Schmid-Lutz, V.: Facility location decisions in supply chain management. In *Operations Research Proceedings 1999*. Inderfurth, K., Schwödiauer, G., Domschke, W., Juhnke, F., Kleinschmidt, P., Wäscher, G. (ed.). Berlin: Springer-Verlag (2000) 467–472

Köksalan, M., Süral, H.: Efes beverage group makes location and distribution decisions for its malt plants. *Interfaces* **29** (1999) 89–103

Lee, C. Y.: A cross decomposition algorithm for a multiproduct-multitype facility location problem. *Computers and Operations Research* **20** (1993) 527–540

Love, R. F., Morris, J. G., Wesolowsky, G. O.: *Facilities location: Models & methods*. New York: North-Holland (1988)

Martin, C. H., Dent, D. C., Eckhart, J. C.: Integrated production, distribution, and inventory planning at Libbey-Owens-Ford. *Interfaces* **23** (1993) 68–78

Mazzola, J. B., Neebe, A. W.: Lagrangian-relaxation-based solution procedures for a multiproduct capacitated facility location problem with choice of facility type. *European Journal of Operational Research* **115** (1999) 285–299

Mehlhorn, K., Näher, S., Uhrig, C.: *LEDA – Library of Efficient Data types and Algorithms, Version 4.1*. Max-Planck Institut Saarbrücken. http://www.mpi-sb.mpg.de/LEDA/ (2000)

Okabe, A., Boots, B., Sugihara, K., Chiu, S. N.: *Spatial tessellations: Concepts and applications of Voronoi diagrams*. Second edn. Chichester: Wiley Series in Probability and Mathematical Statistics (2000)

Ousterhout, J.: *Tcl/Tk Tool Command Language, Version 8.3*. http://dev.scriptics.com/ (2000)

Owen, S. H., Daskin, M. S.: Strategic facility location: A review. *European Journal of Operational Research* **111** (1998) 423–447

Pirkul, H., Jayaraman, V.: Production, transportation and distribution planning in a multicommodity tri-echelon system. *Transportation Science* **30** (1996) 291–302

Pooley, J.: Integrated production and distribution facility planning at Ault Foods. *Interfaces* **24** (1994) 113–121

SAP AG, Germany: *SAP Advanced Planner and Optimizer.* http://www.sap-ag.de/solutions/scm/apo/apo_over.htm (2000)

SOLA: Homepage of the Section on Location Analysis. http://www.uscolo.edu/sola/sola.html (2000)

Teitz, M. B., Bart, P.: Heuristic methods for estimating the generalized vertex median of a weighted graph. *Operations Research* **16** (1968) 955–961

Van Roy, T. J.: Multi-level production and distribution planning with transportation fleet optimization. *Management Science* **35** (1989) 1443–1453

Verter, V., Dincer, M. C.: Global manufacturing strategy. In *Facility location: A Survey of applications and methods*. Drezner, Z. (ed.). New York: Springer-Verlag (1995) 263–282

Bisher erschienene Berichte des Fraunhofer ITWM

Die PDF-Files der folgenden Berichte finden Sie unter:
www.itwm.fhg.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 S., 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 S., 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 S., 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 S., 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 S., 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 S., 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 S., 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 Eley-Rideal and Langmuir-Hinshelwood 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 S., 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 S., 1998)

9. J. Orlik

Homogenization for viscoelasticity of the integral type with aging and shrinkage

A multi-phase 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 non-convolution 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 stress-field from known properties of the

components. This is done by the extension of the asymptotic homogenization technique known for pure elastic non-homogeneous bodies to the non-homogeneous thermo-viscoelasticity of the integral non-convolution type. Up to now, the homogenization theory has not covered viscoelasticity of the integral type.
Sanchez-Palencia (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 integral-modeled 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 space-operator kernels.
(20 S., 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 S., 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 S., 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 S., 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 S., 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 S., 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 S., 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 S. (vier 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 S., 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 S., 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 S., 2000)

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

Polyhedral Properties of the Uncapacita*ted 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 S., 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 S., 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 prescribed 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 S., 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 S., 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 S., 2001)

Stand: Juni 2001