ABOUT THE SOLUTION OF COMBINATORICAL PROBLEMS WITH PROBLEM SOLVING METHODS

H. Hagendorf, W. Kolbe, and E. Sommerfeld
Academy of Sciences of the GOR
Central Institute of Cybernetics and
Information Processing, Berlin, GOR

Abstract

In combinatorical problems from a set of great size elements with specified properties must be selected. Such problems arise in electrical and mechanical engineering. In this paper the characteristics of such problems are outlined. Then the problems are formulated in the state-space approach of problem solving. For the solution of these problems we applicate methods of artificial intelligence. The main steps of the algorithms, which we got as a result of our investigations from the point of viow of problem solving, *are* described. The algorithms are based on heuristic rules for choosing effective operators during the solution process. Results of application are given.

Introduction

In the field of problem solving information has been gathered about efficient search techniques and about kinds of representation of problems (1). We have only found few cases of application of these problem solving methods in computer-aided design (2,3,4). Fields of application of these results are mainly the organization of search processes for robots and for question answering systems.

This paper deals with the rationalization of design processes by means of problem solving methods. We formulate general features of the class of combinatorical problems, which we found in various phases of design processes. One example of this class is placing the components of an electronic circuit onto printed circuit boards or substrates so as to minimize the number of connections between boards. For the solution of these problems we applicate methods of artificial intelligence- The algorithms, which we got as a result of our investigetione, are described. Initial trials of application of these algorithms indicate that they have significant power for effectively solving the described problems.

Problem Formulation

In a paper (5) we have shown that combinatorical problems arise in electrical and mechanical engineering. The effective solution of these problems is one way in rationalizing the design process. The main properties of these problems are the following:

1. The elements x of the set X of potential solution objects are composed of elements a,b,.. of sets A,B,.... . An element x could be a subset of A, a subset of AxB, a partition of A, and so on.
2. The elements x of the set X of admissible solution objects are characterized by a set P" of predicate P (x). With P (x) we describe the properties determining the admissibility of elements x.
3. The elements x of the set X_e of optimal solution objects *are* described by means of a set P of predicate wxth p = p*U (P (x)). $P_e(x)$ is formulated on the basis of the cost g(x):

$$P_e(x) = \begin{cases} \text{'True' if } g(x) = \underset{x' \in X}{\text{extr}}\, g(x') \\ \text{'False' else.} \end{cases}$$

In particular tasks specifications of these points are given. Choosing the state-space approach of problem solving the set of states of the state-space is the set X of possible solution objects and the set of goal states X_g is characterized by P* or P. For changing state x into state x' we must define operators o. The problem is to generate states $x \in X_g$ from states $x \in X \setminus X_g$ by means of the operators o of a set 0. To illustrate these ideas we give some examples for such combinetorical problems.

Decomposition problems

Decomposition problems arise in the field of electrical circuit design, in the field of organization of work, and in system analysis. The solution object x of a decomposition problem is a partition of a set A in n subsets A_i. We set for decomposition problems x - (A_1 A). A relation F over A is defined tRrough these subsets A_i. Relation F_x holds between the elements a_i and a_j of set A if these elements are in the same subset A_i.

For the layouts of electrical circuit design the following decomposition problem with restrictions must be solved:

Given: A = $(a_1,\dots,a_k)$ a set of elements of the electrical circuit which is to be decomposed;
c(i,j) a coefficient equal to the

number of connections between the elements a_i and a_j with $c(i,i) = o$;

$K = (n,k_1,...,k_n)$ a set of natural numbers characterizing a partition x through the number n of the subsets A_l in x and the size k_l of the subset A_l for each l;

$g(x)$ the cost for a partition is the sum of the number of connections $c(i,j)$ of all pairs (a_i,a_j) with a_i and a_j in different subsets.

Find: a partition x of A in n subsets A_l with size of A_l equal to k_l for each l, so that the cost g(x) is minimal.

So the set X for this problem is the set of all possible partitions of A, the set X_z is the set of all partitions with the size of A_l equal to k_l for each l. X_e is the subset of partitions $x \in X_z$ with minimal cost g(x).

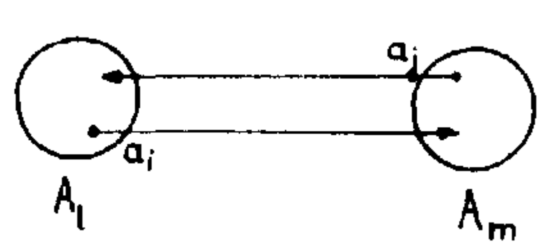
We investigated solution processes based on three sets O_i (i=1,2,3) of operators. An operator o(i,l;j,m) of the set O_1 interchanges the element a_i of the subset A_l and the element a_j of the subset A_m (see figure 1). For a partition x the number of operators in the set O_1 is

$$\sum_{l=1}^{n-1} k_l \cdot \sum_{m=l+1}^{n} k_m$$

An operator o(i,n;m) of the set O_2 joins together an element a_i of the subset A

and the subset A_m (see figure 1). For a partition x the number of operators in set 0_2 is equal to k_n multiplied by (n-1). An operator o(N,n;m) of the set 0_3 joins together a subset N of A_n and subset A (see figure 1). An operator $o \in O_2$ is an operator $o \in O_3$ with size of N equai to 1. For a partition x the number of operators in set 0_3 is equal to the number of possible subsets N in A multiplied by (n-1).

Interchange of $a_i \in A_l$ and $a_j \in A_m$



Junction of $a_i \in A_n$ with A_m

Junction of $N \subseteq A_n$ with A_m

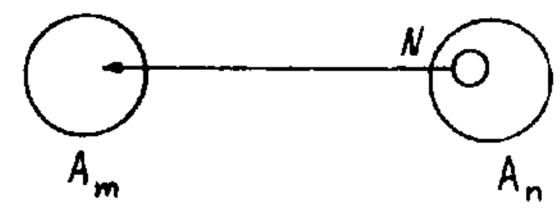


Fig. 1: Examples for the operator sets

A decomposition problem without restrictions is of importance for finding the constituents of systems or for a decomposition of task graphs. In reference to the first problem we give only the differences: The coefficient c(i,j) characterizes the binding strength between the elements a and a . The set K is not given. Vie must find^a partition x so that elements from the same subset A., have a great binding strength, but elements of different subsets have a low binding strength. On the baois of these goal we defined the cost g(x) for a partition x O):

$$g(x) = \frac{1}{k} \cdot \sum_{i=1} h_i(A_l)$$

$$h_i(A_l) = I_i(A_l) - \max_{m \neq l} E_i(A_l,A_m)$$

$I_i(A_1)$ is the mean value of the binding strengths c(i,j) between the element $a_i \in A_l$ and all other elements e.c A,, $E_i(A_l,A_m)$ this one between the element $a_i \in A_l$ and all elements $a_j \in A_m$. We have special definitions for cases with size of subset A_m equal to 0 and with size of subset A, equal to 1. We must find a partition with maximal g(x). To solve this problem we investigated solution processes on the basis of sets 0_4 and 0_5 of operators. An operator o(i,l;m) of set 0_4 units the element a. of the subset A^ and the subset A . This is analogous to operator $o \in O_2$. For a partition x there are k(n-l) operators of this set. An operator o(l,m) of the set 0_5 joins together the subsets A, and A_m. For a partition x we have o,5(n-l)n operators of this set.

Selection Problems

For generating optimal variants realizing a desired technical function the following selection problem must be solved:

Given: $(A_1,..A_n)$ a partition of a set A of constructional elements a_i;

$(b_1,...,b_n)$ a set B of functional elements, whereby each a_1 of the subset A_1 is the realization of the functional element b_1;

$S \subseteq B \times B$ a relation between elements of the set B. S holds between the elements b_1 and b_m if these elements must be coupled in order to the accomplishment of the system function;

$Q \subseteq A \times A$ a relation between elements of the set A. The relation Q holds between the elements a_i and a_j if they could be coupled;

$g(x)$ the cost for an object x is the sum of the costs of each element a_i in x.

Find: an object $x = (a_{i_1},...,a_{i_n})$ with $a_{i_1} \in A_1$ and $P_e(x)$ = 'True', whereby relation Q must hold between a_{i_1} and a_{i_m} if relation S holds between the elements b_1 and b_m.

So we have $X = A_1 \times \ldots\ldots \times A_n$ for this selection problem. Possibilities for the effective solution of this selection problem are described in (5).

Algorithms

The power of algorithms for the solution of such problems is determined by the time for generating objects x and by the cost g(x). In the field of artificial intelligence search techniques are developed for generating goai states in the state-space approach of problem solving- We adapted and developed these techniques to get efficient algorithms. The algorithms do not generate optimal solution objects with probability 1. The objects x *are* the states in the problem space, the operators of the sets 0. (i=1,..,5) realize the transitions between states, we are going to describe briefly the general solution algorithm for the decomposition problems. The efficiency of the search process is determined by the following steps (6) :

- Choice of a start object x .
- Selection of a set of operators,
- Selection of a subset of applicable operators from the chosen set.
- Selection of an operator from this subset. The application of this operator produces the next object. We repeat these steps until a break-off point is reached.
- Determination of a break-off point.

The selection process is based on heuristic rules for operators oxx->x'. A rule is defined by means of the weight Δ for an operator o. We got these weights as a result of the analysis of the costs g(x) and g(x'). so these weights are dependent from the object x and/or from the object x'. Through the kind of computation of these weights we are sure that they are only computed for applicable operators.

For the decomposition problem with restrictions we determined the following weights:

1. Weight $\Delta_1(i,n;m)$ for the selection of an operator $o(i,n;m)$ is the sum of the numbers $c(i,j)$ of connections from element $a_i \in A_n$ to all elements $a_j \in A_m$:

$$\Delta_1(i,n;m) = \sum_{a_j \in A_m} c(i,j) \text{ with } a_i \in A_n .$$

2. Weight $\Delta_2(i,n;m)$ for the selection of an operator $o(i,n;m)$ is the difference between the sum of the numbers $c(i,j)$ of connections from element $a_i \in A_n$ to all elements $a_j \in A_m$ and the sum of the numbers $c(i,j)$ of connections from the element $a_i \in A_n$ to all other elements $a_j \in A_n$:

$$\Delta_2(i,n;m) = \sum_{a_j \in A_m} c(i,j) - \sum_{a_j \in A_n} c(i,j).$$

3. Weight $\Delta_3(i,l;j,m)$ for the selection of an operator $o(i,l;j,m)$ is the sum of weights $\Delta_2(i,l;m)$ and $\Delta_2(j,m;l)$ subtracted by $2c(i,j)$:

$$\Delta_3(i,l;j,m) = \Delta_2(i,l;m) + \Delta_2(j,m;l) - - 2c(i,j).$$

4. Weight $\Delta_4(N,n;m)$ for the selection of an operator $o(N,n;m)$ is the difference between the sum of the coefficients $c(i,j)$ of all pairs (a_i,a_j) in the subset $A_m \cup N$ and the sum of coefficients $c(i,j)$ of all pairs (a_i,a_j) in the subset A_m:

$$\Delta_4(N,n;m) = \sum_{a_i \in N} \Delta_1(i,n;m) + \sum_{a_i,a_j \in N} c(i,j) .$$

In all ce&es the best operator of the set of applicable operators is this one with the maximal weight. Distinct variants of search algorithms based on these operators and weights are implemented. On the basis of the efficiency of the various algorithms a variant with the following characteristics is firstly choosed:

- Select the first element a of the subset A from A_n= A. There are several possibilities for this selection which here we do not discuss.
- Compose the set A_1 with operators from the set O_2. The first $o,5k_1$ elements of A_1 we select from A_n on the basis of operators choosed by weights $\Delta_1(i,n;m)$. The other $o,5k_1$ elements of A_1 we select on the basis of operators choosed by weights $\Delta_2(i,n;m)$. So with k_1 steps we have generated the subset A_1.
- After this the subsets $A_2,...,A_{n-1}$ are generated element by element in an analogous way. So after $(k-k_n)$ steps an admissible object $x \in X_z$ is generated.
- The object x generated by the application of operators from the set O_2 is the start object for the next phase: We interchange pairs of elements by means of operators from the set O_1. We select the operators on the basis of weights $\Delta_3(i,l;j,m)$. The process stops if after the application of an operator from the set O_1 onto object x the generated object x' has cost g(x') not less than the cost g(x) of object x. We found this last phase also in (7).

With the kind of computation of the weights it is provided that only applicable operators are selected. With the help of this algorithm we found out that the cost g(x) is hardly to increase by the application of operators from the set O_1 after the generation of the first aduissible object by means of oe 0~.

A better solution object at the expense of the search time is attainable through the application of operators o(N,n;m) instead of operators o(i,n;m). With these operators we select subsets $N \subset A_n$ with a predetermined number V of elements and join together N and A . In our algorithm we consider not all possible V-subsets N of A . Instead we compute subsets N element by element by means of operators from the set O_2:

We generate all possible sequences of length v with weights $\Delta_1(i,n;m)$ and $\Delta_2(i,n;m)$ as elements. On the basis of each of these sequences we determine a sequence of operators o(i,n;m). The application of each operator sequence results in a single v-eubset N. Each of these subsets generated by an operator sequence is a subset N of on operator o(N,n;m), The weights $\Delta_4(N,n;m)$ are then determined and the operator with the maximal weight is selected.

In an analogous wey the algorithm for the solution of the decomposition problem without restrictions is generated on the basis of operators of the sets 0. and O_5. For the selection of an operator from the set O_4 we choose as weight $_5(i,1;m)$ the component h.(A,) of the cost g(x). we select the operator with the minimal weight. After reaching the subgoal $h_i(A_1)$^ 0 for i=1(1)k we applicate operators of the set O_5. The selection process of these operators is controlled by the weight $\Delta_6(m,l)$:

$$\Delta_6(m,l) = \max_{a_i \in A} \frac{1}{k_m} \cdot \sum_{a_j \in A_m} c(i,j)$$

A, is determined by the element a. in this computation. The operator with the maximal weight is applicated. The search process stops after reaching a threshold value for $\Delta_6(m,1)$.

Applications

The presented algorithms are based heavily on experimental evidence, although there are quite plausible reasons for performing the particula r operations (5,7). In the following we characterize some applications.

We examined the behaviour of the algorithm for the decomposition problem without restrictions if the matrix of coefficients c(i,j) is a matrix with known clusters. So the algorithm always finds in a set A the partition x* with the following properties if this partition in A exists: $c(i,j) = c_1$ for $(a_i,a_j) \in F_{x^*}$ and $c(i,j) = c_2$ for $(a_i,a_j) \notin F_{x^*}$ with $c_1 > o,5c_o > c_2$ whereby c_o is the maximal possible coefficient in the matrix of coefficients. With the start object $x = (A_1=(a_1),...,A_n=(a_n))$ the algorithm stops after reaching the partition x*. This algorithm is applicated for the clustering of experimental data, which are the results of an experiment for the identification of inner structures in human problem solving. Baseo on such groupings we could give statements about properties of inner structures by means of scaling methods (8). This algorithm helps also the projectants of systems in the analysis of the system structure end their decomposition. An example show this: For decomposing the orgenizatlonel structure of a production department the flow of Information between units was determined by experts* This flow is set equal to the binding strength c(i,j)- The algorithm generated a partition x with cost g(x) = 1,3, The maximal possible value for g(x) was 5. The analysis of these results show that the costs $h_i(A_i)$ for each element

a. are greater than 0. This means, that

the flow of information between each element a. e A, to the subset A, is greater

than tho flow of information between the element a_l c_lA, to each other subset A_m.
The algorithm for the decomposition problem without restrictions is also a part of a program for the realization of the structure of a technical system. With this algorithm we determine the constituents of the system structure (5).

We tested the algorithm for the decomposition problem with restrictions on a variety of practical problems. To illustrate the power of the algorithm we give some statistical results:

1. Three different sets A of size 72 were partitioned. For each set A we computed with the algorithm four different partitions x characterized by the following sets K: (3,24,24,24), (3,30,24,18), (3,30,30.12) and (3,36,18,18). For each set K we also generated 10 random partitions over A The main result is the following: In the mean with our algorithm we generate partitions with cost g(x) 10 % less than the cost of random partitions. The variability coefficient of the distribution of the random partitions is 3,5 ^.
2. A practical example with set A of size 24 is also computed, We generate partitions with K * (3,10,8,6) and K = (3,9,9,6). For this problem we also generate analogous to point 1 a set of 10 random partitions. In the mean our algorithm generates partitions with cost g(x) 18 % less than the cost of random partitions. The variability coefficient of the distribution of random partitions is 3 %.

Further results of this kind must be computed.
This algorithm is also a part of a program for the placement of logical elements on double sided printed circuit boards. On figure 2 the structure of this program is given:

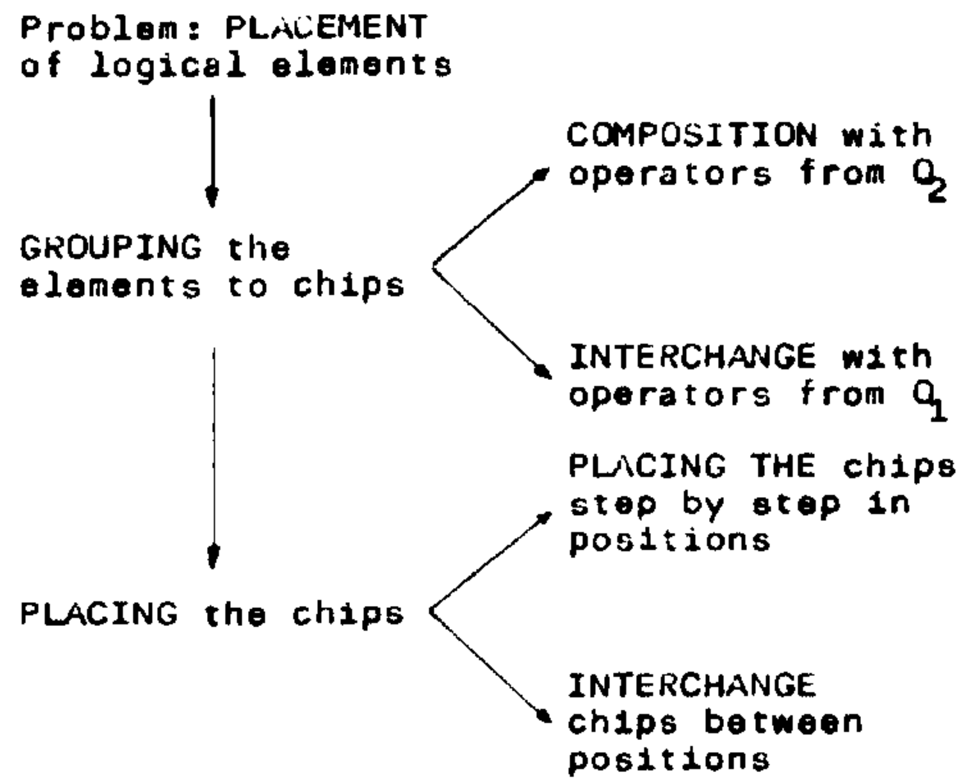


Figure 2: Structure of the program for the placement of logical elements.

On the basis of placements which we computed with this program the routing of all connections between positions on the board is possible with a routing program with 96 - 98 %

Conclusions

The results show that the application of problem solving methods is one way in rationalization of design processes. The practical applications are encouraging. In the future we are interested to find estimates of the probability for reaching extremal values of the cost g(x) for such problems. In the past we restricted our investigations on problems with only structural relationships between elements a e A such as the number of connections, the flow of information and so on. From the point of view of electrical and mechanical engineering the functional features of the elements a < A can not be neglected in the future. So we hope to go one step in the direction of automatically solving synthesis problems as parts of design processes by means of problem solving methods.

Acknowledgements

The computation of routings was carried out by P. Bennewitz by means of the program constructed by K, Schaller under the leadership of O. Seedier, 'Institut fur Nachrichtentechnik Berlin'.

References

1. Nilsson, N.O. : Problem solving methods in artificial intelligence. New York 1971.

2. Eastman, CM.: Heuristic algorithms for automated space planning. Proceedings of the IJCAI-71. London 1971.

3- Goranski, F.K., Gorelik. A.C, Sosulewitsch, D.M., and V.A. Trainev: Elementy teorii avtomatizacii masinostroitelnogo proektirovanija s pomoscju vycislitelnoj techniki. Minsk 1970.

4. Powers, CO.: Non-numerical problem solving in computer aided design. In 3. Vliestra and R.T- Wiengla (eds.), Computer aided design. Amsterdam 1973.

5. Hagendorf, H., Kolbe, W., and E. Sommerfeld: Heurlstische Losungsverfahren bei der koabinatorischen Optimierung. In F. Klix, H. Sydow, and W Krause (eds.), Kybernetik-Forschung 5. Berlin, in press.

6. Newell, A- and H.A. Simon: Human problem solving. New York 1972.

7. Kernighan, B.W and S. Lin: An efficient heuristic procedure for partitioning graphs. *&STD,* vol, 49, *no,* 2, 1970 (291-308).

8. Oulge, B., Haqendorf, H., and H. Sydow: Analyse der Strukturerkennung in Problemlosungsprozessen mit skalierungsverfahren. In F. Klix (ed.), Organismische Informationsverarbeitung. Berlin 1974.

9. Rubin, C: Optimal classification into groups. 0. Theoret. Biol., vol. 15. 1967 (103-144).