A COMPUTATINAL METHOD FOR DYNAMIC LINEAR PROGRAMMING

TAKESHI FUKAO

(Electro-technical Laboratory)

(Presented at the 4 th meeting November 16, 1958)

(Received October 5, 1959)

1. INTRODUCTION.

The dynamic linear problem, consisting in optimizing the cost (or measure of effectiveness) over some time period, consists of a sequential set of static problems which have nearly the same structures each other.

Applying the standard linear programming technique to such a dynamic linear problem without noticing the above feature, the problem becomes an inhibitingly large linear programming problem, and this situation is not desirable in view of the difficulties of computations, and the necessity of higher accuracy of computational results, etc.

On the other hand, if we notice the above feature of dynamic problem, and if we solve the problem following to some suitable sequential process, it is only necessary to solve many smaller, but nearly same structural linear programming problems, and the faults described above could be removed. A typical type of these sequential processes is the Bellman's dynamic programming, and the other typical one is the Danzig's dynamic linear programming method.[2] Although they are excellent general methods, there are some difficulties in application to the practical problems. That is, in Bellman's dynamic programming, the complexity of each static problem is an essential difficulty in computations, and it is necessary to invent some devices suitable to the problem in the Danzig's method.

In this paper we will describe a more natural computational method (one type of relaxation method) and its numerical examples, and we will touch on the extensions of the basic concepts of this method.

2. REDUCTION OF A DYNAMIC PROBLEM TO A MULTI-STAGE DECISION PROCESS (OR A GENERALIZED VARIATIONAL PROBLEM).

We define a multi-stage decision process (problem) as follows:

1) It includes decisions (variables) and states (variables), as shown in Fig. I., and the relations

$$\left.\begin{aligned} S_1 &= T_1(S_0,\ X_1) \\ S_2 &= T_2(S_1,\ X_2) \\ &\cdots\cdots\cdots\cdots \end{aligned}\right\} \qquad (1)$$

hold,

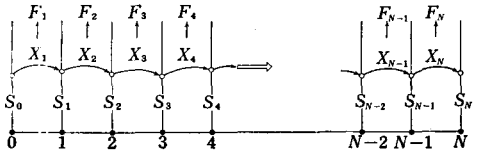


Fig. 1.

2) the cost (measure of optimization) of it is separable:

$$F = F_1(S_0,\ X_1,\ S_1) + F_2(S_1,\ X_2,\ S_2) + \cdots\cdots, \qquad (2)$$

and

3) the constraints are also separable:

$$\left.\begin{aligned} G_1(S_0,\ X_1,\ S_1) &\leqq 0 \\ G_2(S_1,\ X_2,\ S_2) &\leqq 0 \\ &\cdots\cdots\cdots\cdots \end{aligned}\right\} \qquad (3)$$

Where, S_i describes a set of states at the i-th "time" (It is not necessary "time" is time. It is sufficient "time" is only an order number of the sequential state.), X_i, a set of decisions at the i-th stage, respectively. Dynamic programming is applicable to such multi-stage decision problems. Then it is necessary to transform a given problem to a problem which satisfies the conditions (1), (2), and (3) by introduction of an adequate set of decisions and states, in order to be able to apply the dynamic programming concept.

Particularly, if we assume the linearities in conditions (1), (2), and (3), the problem becomes a linear programming problem, and its structure is represented by the coefficients matrix form of the following equations (3′):

$$\left.\begin{aligned} G_1(S_0,\ X_1,\ S_1,\ \lambda_1) &= 0 \\ G_2(S_1,\ X_2,\ S_2,\ \lambda_2) &= 0 \\ &\cdots\cdots\cdots\cdots \end{aligned}\right\} \qquad (3')$$

Where, λ_1, $\lambda_2\cdots\cdots$ are slack variables.

Then, let us discuss the characteristics of the dynamic problems

owing to this coefficients matrix (technology matrix) form. These discussions come largely from Danzig's paper.[2]

Formulating a dynamic problem (or a sequential problem) by linear programming, its coefficients matrix or technology matrix is generally a block triangular one.

For example, Fig. 2. shows the matrix of 4-stages case. Where, A_{ij} is the sub-matrix which shows the effects of the j-th stage variables onto the i-th stage constraints.

If only the matrices A_{11}, A_{22}, A_{33}, A_{44} in diagonal parts are not zero matrices and others are zero matrices, no relations exist between the stages. It shows a set of four independent static problems. Then not all the matrices A_{21}, A_{32}, A_{43} are zero in dynamic problem. (See Fig. 3)

	1	2	3	4
1	A_{11}			
2	A_{21}	A_{22}		
3	A_{31}	A_{32}	A_{33}	
4	A_{41}	A_{42}	A_{43}	A_{44}

Fig. 2.

	1	2	3	4
1	A_{11}			
2	A_{21}	A_{22}		
3		A_{32}	A_{33}	
4			A_{43}	A_{44}

Fig. 3.

Fig. 4 shows the case in which only two variables are in common in the constraints of the neighboring two stages. If these common variables were fixed constant, the matrix which represents the structure of the problem would be reduced to the matrix of a set of the independent static problems.

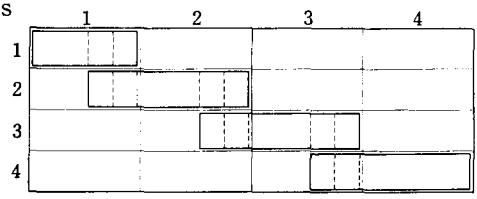


Fig. 4.

This means that we can divide the over-all problem to many smaller problems by considering such variables as parameters. Such common variables are states (variables), and the remaining variables are decisions (variables). Number of these common variables or states variables becomes larger according to the higher dimensional system

and to the system including the time lag and acceleration effects. Then, at last, the matrix may become such as Fig. 5. In this case, it is necessary to consider two stages as one stage in the new problem in order to reduce the problem to a multistage decision process.

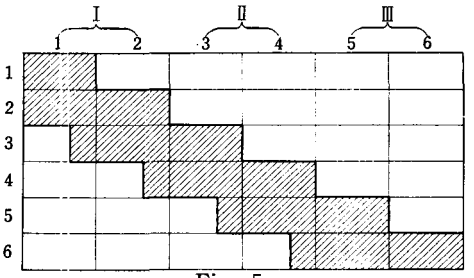


Fig. 5.

Because the fewer states are in a stage, the more easily the problem is dealt with, the central problem is how to obtain the more loosely-coupled matrix by the introduction of the suitable states and decisions.

3. ARBITRARINESS OF THE CHOICES OF STATES AND DECISIONS.

In Fig. 5 case, it was necessary to take two stages in the old problem as one stage in the new problem in order to reduce the problem to a multi-stage decision process.

Let us call the variables which the constraints of a stage and the constraints of the next stage have in common as the coupling variables. We designate the number of the coupling variables, the states, the decisions and the all variables of one stage, n_c, n_s, n_d, and n, respectively.

If

$$(K-1)n \leqq n_c \leqq Kn, \tag{4}$$

then it is necessary to take at least K stages in the old problem as one stage in the new problem in order to reduce the problem to a multi-stage decision process. If we take K stages as one stage in the new phase, under the condition $K \geqq 2$, the relation

$$n_d \leqq n_s(=n_c) \tag{5}$$

exists. The equality is valid only when $n_c=n$, that is, Fig. 6 case.

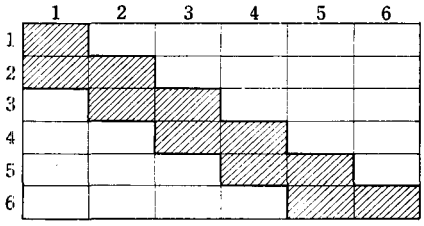


Fig. 6.

The fewer states are in a stage, the more effective a multi-stage deci-

sion process is. That is, it is more effective when applied to the problem which has the week couplings between the stages. Moreover, when the couplings are stronger, the classifications of the states and the decisions are more and more artificial, and there are no physical differences between them at all.

In this situation, the application of the following relaxation method, based on a standpoint of taking K stages as one stage in the new problem only from the view point of the division of the variables (not necessarily from the view point of the division of the constraints), is more systematic and more effective sometimes in the stronger coupling problems.

Let us consider the Fig. 5 case, for example. We take the combination of the 1-st and the 2-nd stages in the old problem as the I-stage, the combination of the 3-rd and the 4-th as the II-stage, and the combination of the 5-th and the 6-th as the III-stage, respectively, in the new division of the variables. See Fig. 5.

We assume that a multi-stage decision process M_1 which has the variables of the odd stages (the I-, the III-stages) as its decisions and the variables of the even stage (the II-stage) as its states, the another multi-stage decision process M_2, which has the variables of the even stage as its decisions and the variables of the odd stages as its states.

First, from a standpoint of M_1, the decisions of the I-stage are determined independently of the III-stage decisions, preserving the variables of the II-stage (states) constant, Next, from a standpoint of M_2, the decisions of the II-stage are determined, preserving the variables of the I- and III-stages constant (using the newest informations). In the next place, from a standpoint of M_1 again, the decisions of the III-stage are determined, preserving the variables of the II-stage constant. This processes are repeated again and again.

In this method, two kinds of multi-stage decision processes are used alternately, and corresponding to the kind of multi-stage decision process the variables of a stage are regarded as the decisions, and also as the states in the another one.

We should bear in mind that the division of the constraints do not necessarily correspond to the division of the variables. For example, in the above case, from a standpoint of M_1, the decisions of the I-stage

are the variables of the 1-st and the 2-nd stages of the old problem, but the constraints to these decisions are the constraints of the 1-st, 2-nd, 3-rd and 4-th stages of the old problem.

This method is a relaxation method, and similar to the overlapping method described in the following paragraph.

4. OVERLAPPING METHOD AND ITS AVAILABILITIES.

Although this method is applicable both to the multi-stage decision processes (paragraph 2.) and its extensions (paragraph 3.), we shall illustrate it with respect only to the former.

4-1. ONE FORM OF OVERLAPPING METHOD.

If a problem is reduced to a multi-stage decision process, the relations between decisions and states are shown in Fig. 1, as already described.

Now, we take the successive approximations due to overlapping as follows:

The initial estimates of states are $S_0^{(0)}$, $S_1^{(0)}$, $S_2^{(0)}$, $S_3^{(0)}$, ·····.

(i) The two-stages optimization problem of which boundary conditions are $S_0=S_0^{(0)}$, $S_2=S_2^{(0)}$, is solved. The solution determines $S_1=S_1^{(1)}$. See Fig. 7.

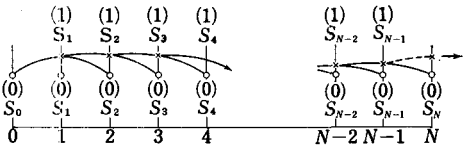


Fig. 7.

(ii) The two-stages optimization problem of which boundary conditions are $S_1=S_1^{(1)}$, $S_3=S_3^{(0)}$, is solved. The solution determines $S_2=S_2^{(1)}$. See Fig. 7.

(iii) The two-stages optimization problem of which boundary conditions are $S_2=S_2^{(1)}$, $S_4=S_4^{(0)}$, is solved. The solution determines $S_3=S_3^{(1)}$. See Fig. 7.

The similar step, that is, solving the two-stages optimization problem using the newest S informations as its boundaries, and then moving one stage to the right, is repeated until getting $S_N^{(1)}$. Next, again back to the initial two-stages optimization problem. These approximation steps continue until all S_i have converged.

For the problems having free end states conditions, many modifications are possible. For example, modifying the end states S_N

gradually, taking a particular sub-problem of which boundary conditions are $S_{N-1}=S_{N-1}^{(n)}$ without $S_N=S_N^{(n)}$, etc.

The validity of this method is proved as follows. We consider a maximization problem, and we suppose the uniqueness of the solution for simplicity.

(1) Each step of "overlapping" (solving the two-stages linear programming) has monotone increasing property. From this property and the boundedness of the problem, it is clear to converge to some states.

(2) We assume that $S_0(t)$ is a series of states or a set of argument functions converged by overlapping method, and $\delta S_{i,i+2}(t)$ is a set of variational functions which are zero before the i-th "time" and after the i+2-th "time".

Then, general argument functions are expressed by

$$S(t)=S_0(t)+\sum_{i=0}^{N-2}\delta S_{i,i+2}$$

Because the cost is linear with respect to decisions, and the relations between decisions and states are linear,

$$\text{cost}\ \{S(t)\}=\text{cost}\{S_0(t)\}+\sum_{i=0}^{N-2}\text{cost}\{\delta S_{i,i+2}(t)\}$$

Now, the following conditions hold:

$$\text{cost}\{\delta S_{i,i+2}(t)\}\leqq 0\ \ (i=0,\ 1,\cdots,\ N-2).$$

For if any cost $\{\delta S_{i,i+2}\}$ were positive, the overlapping method would have not yet converged.

From (1) and (2), the optimality of $S_0(t)$ has been proved.

There are many types of overlapping method which vary due to the number of stages of the sub-optimizing problems, the order of overlapping (Nonsystematic overlapping is a general relaxation method for optimization problems.), etc. We can reduce the number of sub-optimizing problems if we increase the number of stages of each sub-optimizing problem. The same proof is established for the extension of the multi-stage decision process(See paragraph 3.).

4-2. EFFECTIVENESS OF OVERLAPPING METHOD. ····· APPLICATIONS OF PARAMETRIC LINEAR PROGRAMMING.

There are so many dynamic problems each of which consists of a sequence of the static problems. Although these static problems have

nearly same structures each other, their envoirments are different. If these envoirments vary not suddenly but slowly, according to the progress of the "time" sequence, a sub-optimization problem is easily solved from the solution of the neighboring sub-optimization problem by the parametric linear programming technique.

Then, making an extreme argument, we shall be able to say the complicated computations of the linear programming are necessary only for the initial sub-optimization problem only once, and others can be solved easily depending on this solution or on the solution of the neighboring problem.

After having solved the all sub-optimization problems once for each, we can take the newest matrix as its start point for solving a sub-optimization problem.

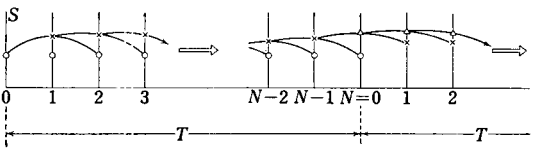
4-3. DETERMINATION OF THE MOST DESIRABLE INITIAL STATES S_0.

In the ordinary variational problems, the fixed terminal conditions, that is, fixed initial and end states are usually given as their boundary conditions. However we can not confine the practical dynamic problems within the fixed terminal conditions type. There are cases which have the free end states, which should be chosen the best values of the initial states, and so on.

One of the most typical conditions of the non-fixed terminals problems will be the determinations of the most desirable initial and end states when the same dynamic problems are repeated cyclically in T interval period.

For such determinations, the overlapping method determines automatically the most desirable initial (and end) states S_0 by continuing the overlapping endlessly, as shown in Fig. 8.

This is one of the remarkable features of the overlapping method.



○ : the first estimation
× : the second approximations
△ : the third approximations

Fig. 8.

4-4. INITIAL ESTIMATIONS.

We must give the initial estimations of the states at each "time" $S_0^{(0)}, S_1^{(0)}, \cdots, S_N^{(0)}$, in the overlapping method.

These initial estimations are not entirely arbitrary, and have to be consistent with all the constraints. Usually, we can find such estimations easily, because in many problems we have known the "approximate" solution. (It is sufficient to know only the "states" approximation.)

However, we shall require some suitable techniques for the problems which have the complicated constraints and have not been known their appearances at all.

One of the powerful methods for these situations is the relaxation of the constraints. That is, at first, establishing the initial estimations, we then alter the given problem to a new one of which constraints are consistent with the initial estimations. After solving it, we back its constraints to the proper constraints gradually. The parametric linear programming techniques are also effective means in this phase.

5. ITERATIVE METHOD FOR A SET OF THE ALMOST SAME STRUCTURAL PROBLEMS.

Let us consider the following constraint:

$$a_{j1}X_1+a_{j2}X_2+\cdots\cdots+a_{jn}X_n=b_j \tag{6}$$

If we express a_{j1}, a_{j2}, $\cdots$, as

$$\left.\begin{aligned} a_{j1}&=\bar{a}_{j1}+\Delta a_{j1}\\ a_{j2}&=\bar{a}_{j2}+\Delta a_{j2}\\ &\cdots\cdots\cdots\cdots\\ a_{jn}&=\bar{a}_{jn}+\Delta a_{jn}\end{aligned}\right\} \tag{7}$$

equation (6) becomes

$$\begin{aligned}\bar{a}_{j1}X_1+\bar{a}_{j2}X_2+\cdots+\bar{a}_{jn}X_n=b_j-(&\Delta a_{j1}X_1+\Delta a_{j2}X_2+\cdots\\&+\cdots+\Delta a_{jn}X_n)\end{aligned} \tag{8}$$

Generally speaking, if Δa_{ji} (i=1, 2, 3, $\cdots$, n) are sufficient small compared with $\bar{a}_{ji}$ (i=1, 2, 3, $\cdots$, n), it will be possible to solve the problem by an iterative method in which at first we give the adequate initial estimates to X_i (i=1, 2, 3, $\cdots$, n) in the bracket of the right hand side of (8) and solve the problem regarding the right hand side of (8) constant, and next, substituting the new solution X_i into the bracket of the right hand side of (8) we solve again the problem regarding the

right hand side constant, and this process is repeated until getting the convergent X_i.

However, in general, if Δa_{ji} are comparable with $\bar{a}_{ji}$, or if the initial estimates of X_i deviate from the true answers of X_i very much, this method may be incorrect. However, in our overlapping method, the above iterative method is always correct, however large Δa_{ji} are, or whatever values X_i take.

For we suppose that the initial estimation of each sub-optimization problem is a feasible solution of it, and each sub-optimization problem in the following overlapping steps has at least one feasible solution of it certainly.

This iterative method is easily carried out by parametric linear programming techniques, and it is effetive when

(i) all the static problems constituting a dynamic problem have the nearly same but slightly different structures, or when

(ii) a_{ji} are nearly equal to the simple integers, and if we take these integers as the coefficients of the technology matrix instead of a_{ji} themselves, the computations of the linear programming is simplified very much. For example, if we can approximate the technology matrix by a matrix of which coefficients are 0, +1, and −1, its effectiveness is remarkable.

6. EXAMPLES.

Example I.

Let us consider a power system in which three hydroplants, flow-interconnected on a stream, and an integrated thermal system supply the power to a lumped load. See Fig. 9.

Our problem is the minimization of the total fuel cost over a day:

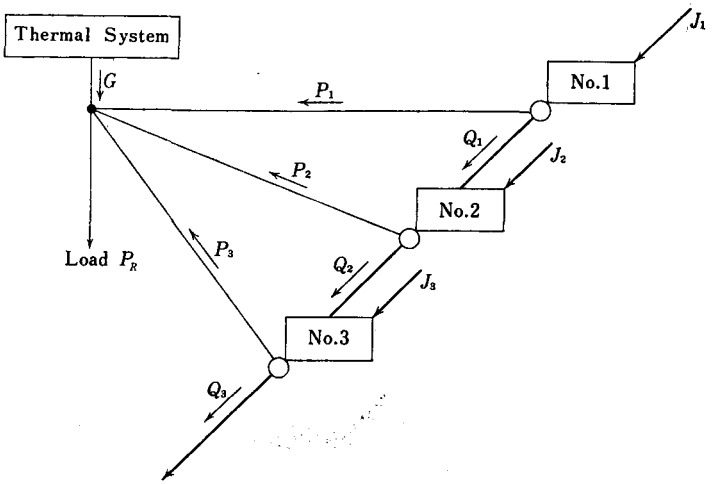
$$\text{Min.} \int_0^T F \, dt$$

where, F is the fuel cost per unit time of the thermal system.

Characteristics and constraints of the system:

(1) Load $P_R(t)$ is shown in Fig. 10, and the following relation must be satisfied:

$$P_R(t) = G(t) + \sum_{i=1}^{3} P_i(t). \quad \text{(MW)}$$



—— : The water flow paths.

—— : The electric power flow paths.

Fig. 9.

Where, G is the thermal output (MW), and P_i is the i-th hydroplant output (MW).

(2) Characteristics and constraints of the hydro-plants:

$$P_i=\sigma_i Q_i,$$

where, Q is the discharge flow into the hydroturbine (m^3/sec), and σ is a constant.

$$Q_{i,\min} \leqq Q(t) \leqq Q_{i,\max}$$

$$S_{i,\min} \leqq S(t) \leqq S_{i,\max}$$

Where, S_i is the water storage of the pond of the No. i-hydroplant (m^3).

There are the relations between S_i and Q_i as follows:

$$Q_1(t)+dS/dt(t)=J(t)$$

$$Q_i(t)+dS/dt(t)=J_i(t)+Q_{i-1}(t). \quad (i=2, 3.)$$

We have neglected the time-lag of flows for simplicity.

(3) Constraints and fuel cost characteristics of the thermal system.

$$G_{\min} \leqq G(t) \leqq G_{\max}$$

The fuel cost per unit time F is a monotone increasing no nlinear function of the themal output G.

In order to be able to apply the linear programming to this problem, there are two methods, one of which is to approximate F a broken line convex function of G, and the other method is a parametric linear programming in which we use the average $(dF/dG)\ (t)$ in some small domain, and alter the domain and $(dF/dG)\ (t)$ successively.

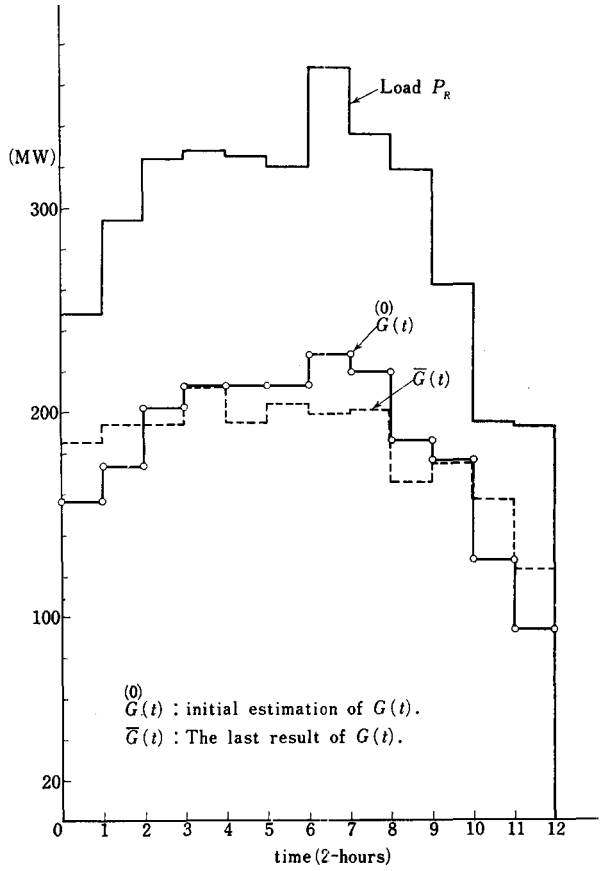


Fig. 10.

We will adopt the latter method. In this case, the above constraint becomes

$$\text{Max}\ [G_{\min},\ G(t)-\Delta G] \leqq G(t) \leqq \text{Min}\ [G_{\max},\ G(t)+\Delta G],$$

where, $G(t)$ is the solution of the just before iterative linear programming problem, and ΔG is the admissible amplitude of fluctuations for the new problem.

This problem is an exceedingly typical variational problem, because the time-lag effects of flows are not included, and, reduces to a multi-stage decision process if we suppose that Q_i are decisions and S_i arestates.

The overlapping method can be very easily applied to this process, and the results from it are shown in Fig. 10, and in Fig. 11.

Fig. 11 shows the situation of the cost reduction. The "O"-position in Fig. 11 means the cost of the initial estimation, and the "1"-position means the cost of the result of an application of a set of overlappings continued successively from the initial time to end time. Repeating the sets of overlappings processes, the the cost converges to the "$\bar{3}$" -position by way of "2".

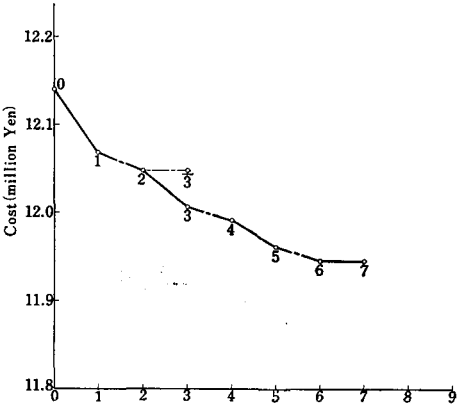


Fig. 11.

The difference of the costs of "2" and "$\bar{3}$" positions is very small, and then wa may say that the cost will almost converge by only two iterations of the sets of overlappings.

Therefore, if we start from the "2"-position using the average $(dF/dG)\ (t)$ calculated at "2"-position and $\Delta G=10$ MW, the cost decreases to the "3"-position by an application of the overlapping method. Repeating one more set of overlappings, the cost further reduces to "4". Again, $(dF/dG)\ (t)$ is recalculated at there, and so on.

Example II.

There are many dynamic problems which have the following

structures:

$$\begin{bmatrix} A & & \\ B & A & \\ B & B & A \\ C & C & C \end{bmatrix} \cdot \begin{bmatrix} X_1 \\ X_2 \\ X_3 \end{bmatrix} \leqq \begin{bmatrix} b_1 \\ b_2 \\ b_3 \\ b_4 \end{bmatrix} \tag{9}$$

For example, these are the bottle-neck problems, the expanding installation system problems, etc. A, B, C are matrices, and we suppose the elements of each row of C have the same signs. Then, if we introduce the variables S_1, S_2, S_3, S_1', S_2', S_3',

$$\begin{aligned} S_1 &= AX_1 \\ S_2 &= BX_1 + AX_2 \\ S_3 &= BX_1 + BX_2 + AX_3 \end{aligned} \tag{10}$$

$$\begin{aligned} S_1' &= C_1X_1 \\ S_2' &= C_1X_1 + C_2X_2 \\ S_3' &= C_1X_1 + C_2X_2 + C_3X_3, \end{aligned} \tag{11}$$

equations (9) are transformed into

$$\begin{array}{cccccccccc|c} [X_1 & S_1 & S_1' & X_2 & S_2 & S_2' & X_3 & S_3 & S_3'] & \\ & I & & & & & & & & \leqq b_1 \\ A & -I & & & & & & & & = 0 \\ C & & -I & & & & & & & = 0 \\ \hline & & & & I & & & & & \leqq b_2 \\ B-A & I & & A & -I & & & & & = 0 \\ & & I & C & & -I & & & & = 0 \\ \hline & & & & & & & I & & \leqq b_3 \\ & & & B-A & I & & A & -I & & = 0 \\ & & & & & I & C & & -I & = 0 \\ \hline & & & & & & & & I & \leqq b_4 \end{array} \tag{12}$$

We may suppose for example, (X_1, S_1, S_1'), (X_3, S_3, S_3') are decisions, and (X_2, S_2, S_2') are states, when we wish to deal with this problem as a multi-stage decision process. (Formally, we may introduce $(X_0 = 0, S_0 = 0, S_0' = 0)$ as the states.)

When we wish to deal with this problem by the relaxation method already decsribed in paragraph 3 (an extension of the multi-stage decision process), we may make (X_1, S_1, S_1'), (X_2, S_2, S_2') and (X_3, S_3, S_3'),

the I-stage variables, the II-stage variables and the III-stage variables, and we may use the two kinds of the multi-stage decision processes alternately.

7. BASIC CONCEPTS OF THE METHOD AND POSSIBILITIES OF THEIR EXTENSIONS.

The multi-stage decision problem described in paragraph 2 can be reduced to a problem which includes only the decisions X_1, X_2, $\cdots$, or a problem which includes only the states S_1, S_2,$\cdots$, if S_0 are given.

It may appear that the treatment of a dynamic problem by a multi-stage decision process is wasteful from a standpoint of reducing the size of the over-all problem.

However, it is true that the method of subdivision can be accomplished easily by the additions of the superfluous variables. We shall say as follows:

A matrix of a dynamic problem can be a matrix which has the stronger couplings between the stages, or can be a matrix which has the more loose couplings between the stages, according to the expression of the problem.

To make the matrix, loosely coupled is accomplished by the additions of the variables to the given system. This corresponds to expressing the inner structure more finely. On the other hand, a compact matrix expression of the system without the additional variables corresponds to the standpoint of viewing its structure from the outside, or of regarding the system as a black box, if we make an extreme argument.

This basic concept will be included in the G. Kron's "Diakoptics".[6]

The other basic concept of the method is to observe that the dynamic problem consists of the many static problems having nearly the same structures. On account of it, we can deal with each sub-divided static problem on a common base, and the computations become very easy. (parametric programming).

One of the extensions of these basic concepts is a possibility of "Network Programming", and the other extension is a possibility of the easier computations by the refinement of the system.

ACKNOWLEDGEMENTS.

I am very much thankful to all the members of Dynamic Programming group of JUSE, for their kind discussions.

REFERENCES:

1) R. Bellman: Dynamic Programming, 1958.
2) G. B. Danzig: On the status of multi-stage linear programming problems, The RAND Corporation, p. 1028, 1957.
3) G. B. Danzig: Recent advances in linear programming, Management Science, Vol. 2, No. 2, Jan. 1956.
4) G. B. Danzig: Upper bounds, secondary constraints, and block triangularity in linear programming, Econometrica, Vol. 23, No. 2, April, 1955.
5) T. Fukao, T. Yamazaki: A computational method of economic operation of hydro-thermal power systems including flow-interconnected hydro-plants, the Journal of I. E. E. of Japan, Vol. 79, No. 5, May, 1959.
6) G. Kron: A set of principles to interconnect the solutions of physical systems, Journal of Applied Physics, vol. 24, No. 8, August, 1953.