Decision of Timings of Signal Program Switching in Pretime Multi-Program Control

Masao Kuwahara and Masaki Koshi
Institute of Industrial Science, University of Tokyo
7-22-1 Roppongi, Minato-ku, Tokyo 106, Japan

Abstract

For the operation of a pretime multi-program signal, we have to decide in advance not only each program (a set of cycle time and green splits) but also times of day when one program should be switched to the next. Given time-dependent demand flow at an isolated undersaturated intersection, the present paper discusses the problems of (1) what should be the optimal signal parameters of each program and (2) when one program should be switched to another. For the first problem (1), based on several previous studies, the convex optimization programming is formulated with the cycle time and green splits as unknown variables so as to minimize the total delay during a given time period to which the program is applied. For the second problem (2), the decision of timings of program switching has largely relied on engineers' field experiences and few theoretical analyses on the decision have been reported. To theoretically analyse the structure of the switching decision problem, we show that the problem is reduced to the dynamic programming so as to minimize the total delay of all programs throughout a day, given the number of programs available. We examine some questions for practical applications such as the computation time and the delay reduction through a simple example. Then, we briefly discuss the extensions to other signal controls holding the similar problem structures such as the traffic responsive program selection control where the number of programs is limited but a program can be switched to another as many times as necessary.

1. Introduction

A pretime multi-program signal is operated in such a way that program *1* (a set of cycle time and green splits) is used, for instance, from *0:00* to *6:15*, program *2* from *6:15* to *10:45*, program *3* from *10:45* to *15:30*, and so on, as shown in *Fig.1*. Traffic engineers, therefore, have to decide in advance not only each of the programs but also times of day when one program should be switched to the next. The present paper deals with an isolated undersaturated intersection with given time-dependent demand flows.

We divide the whole problem separately into two parts :

Published 1990 by Elsevier Science Publishing Co., Inc.

(1) what are the optimal signal parameters of a program, given a time period for which the program is used, and

(2) how the time periods of different programs should be determined.

In the first part, the optimal signal parameters of a program are determined so as to minimize the total delay during a given time period to which the program is applied. There have been several studies on the determination of signal parameters for an isolated intersection such as *Allsop [1,2], Zuzarte [3], Hydecker et. al. [4], Gallivan et. al. [5], Improta et.al. [6],* and *Cantarella et. al. [7].* These studies have proposed methods to determine durations of green times and cycle time and/or sequences of green times. Particularly, in references *4, 5, 6,* and *7*, convex programs were formulated to simultaneously obtain sequences of greens and durations of green and cycle times by introducing the starting times of green times, their durations, and the inverse of a cycle time as unknown variables. Several kinds of objectives were discussed in these programs such as the optimization of delay per unit time, the reserved capacity, and so on. Depending on the types of the objective functions, these convex programs can sometimes be reduced to linear programs and solved by the simplex method. For other objective functions, we would apply a standard non-linear optimization technique to solve the convex programs. In this paper, we adopt the formulation following the previous studies.

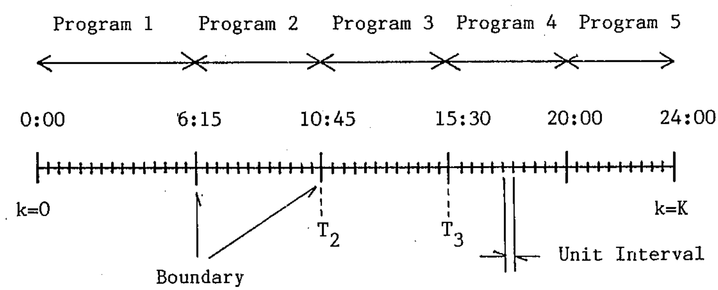

Figure 1. Signal Programs on a Discrete Time Axis

In the second part, the core part of this paper, we discuss how to determine the time periods of different programs so as to minimize the total delay of all programs throughout a day, given the number of programs available. In current practice, the decision of timings of program switching has largely relied on engineers' field experiences and few theoretical analyses on the decision have been reported. The purpose of this paper is to disclose the essential structure of the problem of switching decisions.

We formulate the second part of the problem in the dynamic programming based on *the Bellman's optimality principle* assuming that each program is used in only one time period just for the sake of simplicity, and theoretically analyse the efficient way of the calculation. For future extensions, we briefly mention some other signal controls which are recognized to have the same structure as this switching problem such as a pretime control in which the same program is used not only in a unique time period but also two or more different time periods, and the traffic responsive program selection control where the number of programs is limited but a program can be switched to another as many times as necessary.

2. Optimal Signal Parameters of Each Program

2.1. Discrete Time Axis and Time-Dependent Traffic Demand

In general, traffic demand vary time-dependently through a day. To describe the flow variations, we divide the *24-hour* time of a day into a number of unit intervals of equal length as shown in *Fig.1*. The time boundaries are denoted k, $k = 0, 1, 2, \ldots, K$, and the k th interval is written as $[k-1, k]$. Practically, the length of a unit interval would be about *15* minutes resulting in $K = 96$ intervals a day, so that traffic demand can be considered as stationary during an interval.

Traffic flow demand is classified according to the vehicle maneuver such as turning right or left and going straight. These segments of traffic are called streams. For each of the time intervals, the average traffic volume of each stream i, $i = 1, 2, ..., I$ is assumed to be given:

x_{ik} = an average normalized traffic flow of stream i in interval $[k-1,k]$.

2.2. Optimal Signal Parameters for a Program

In the first part of the problem, the optimal signal parameters of a program is determined so as to minimize the total delay during a given time period to which the program is applied. In this section, we assume that the switching timings of programs have been determined such that program j, $j = 1, 2, ..., J$, is operated from time T_{j-1} to T_j as shown in *Fig.1*.

Let us define signal parameters of program j as follows :

G_{ij} = an effective green time of stream i in program j *[sec]*,

C_j = a cycle time of program j *[sec]*,

g_{ij} = a green split of stream i in program $j = G_{ij} / C_j$.

From the results of previous studies mentioned above, we similarly formulate the minimization programming which determines the optimal signal parameters so as to minimize the total delay during a time period $[T_{j-1},T_j]$. For the objective function, the total delay of stream i during the k th interval, $T_{j-1} < k \leq T_j$, is described as a function of signal parameters of C_j and g_{ij}, given x_{ik}'s: $D\{x_{ik}, g_{ij}, C_j\}$. Therefore, the total delay associated with program j, $TD(T_j / T_{j-1})$, is written as

$$TD(T_j \;/\; T_{j-1}) = \sum_{k=T_{j-1}+1}^{T_j} \sum_{i=1}^{I} D\{x_{ik}, g_{ij}, C_j\}. \tag{2.1}$$

We could employ several proposed delay functions such as *the Webster's 2-term delay formula* which is known as a convex function in g_{ij} and $1/C_j$.

Although it is difficult to analytically show the convexity for other formulae such as *the Webster's 3-term formula* and *the HCM delay formula*, these functions are practically also convex in g_{ij} and $1/C_j$ at the proximity of the optimal solution.

The convexity of the feasible region depends on constraints employed. Usually, the green splits and cycle time are optimized given the phase plan that is a sequence of green times of streams. But for different formulations, sometimes the green time sequence is also unknown and simultaneously determined as well as g_{ij} and C_j. However, for both types of formulations, it has been known that the feasible region is convex in g_{ij} and $1/C_j$.

Therefore, the determination of signal parameters is generally reduced to a convex problem and the optimal signal parameters could be determined using a standard non-linear optimization technique. The optimized value of the objective function (= total delay for program j) can be also evaluated.

As an example, a convex program is formulated for a simple four-leg intersection shown in *Fig.2*. A three-phase plan is assumed to be determined in advance for the total of ten streams as in the figure.

Green split g_{ij} and cycle time C_j must be determined within the feasible region bounded by several constrains. First, since we consider the undersaturated traffic condition, a sufficient green time must be given to each stream.

$$g_{ij} \geq x_{ik}, \qquad i = 1, 2, \ldots\ldots, 10, \quad T_{j-1} < k \leq T_j. \tag{2.2}$$

Second, relationships of green splits of streams and phases are defined according to the pre-determined phase plan as in *Fig.2*.

$$\begin{aligned} &g_{1j} = g_{2j} = g_{3j} = g_{4j} = \phi_{1j}, \\ &g_{5j} = g_{6j} = \phi_{2j}, \\ &g_{7j} = g_{8j} = g_{9j} = g_{10j} = \phi_{3j}, \end{aligned} \tag{2.3}$$

where ϕ_{pj} = a time length of phase p of program j divided by C_j, called a phase split here.

A sum of the green times and lost time per cycle, L_j, must be equal to the cycle time:

$$\phi_{1j} + \phi_{2j} + \phi_{3j} + L_j / C_j = 1, \tag{2.4}$$

where L_j = a given lost time per cycle of program j.

The objective function of *Eq.(2.1)* should be minimized subject to the constraints of *Eq.(2.2)*, *(2.3)*, and *(2.4)*. As mentioned before, the feasible region bounded is convex in g_{ij} and $1/C_j$, since all the constraints are linear functions of the unknown variables.

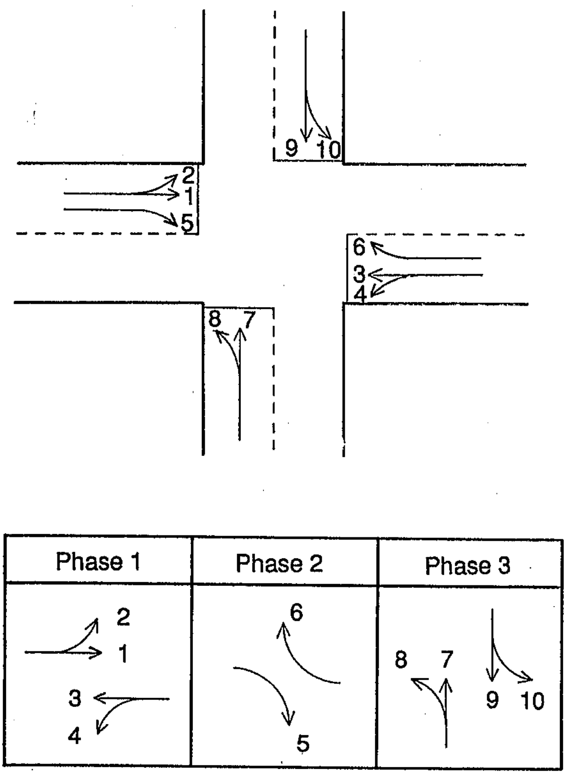

Fig. 2. A Four-Leg Intersection with a Three-Phase Control

3. Decision of Timings of Signal Program Switching

3.1. Outline of the Approach

We have discussed how to determine the optimal signal parameters for a particular program j assuming that switching timings T_{j-1} and T_j are given. In this section, we consider the second part of the problem; that is, how the switching timings should be determined so as to minimize the total delay of all programs throughout a day. We assume that the number of programs used, J, are given and each program is used in only one time period just for the sake of simplicity. This problem is equivalent to the determination of a certain number of boundaries on the discrete time axis shown in *Fig.1*, where a program is applied to a time period between two adjacent boundaries.

The second part problem is, thus, reduced to a combination problem. For instance, if five programs are used which means that we should determine four boundaries on the discrete time axis, the number of possible combinations of the switching plan in case of *96* unit intervals is ${}_{96-1}C_4$ = *3,183,545* which is obviously too many to examine in practice. In order for us to reduce the number of calculations, the following property of this combination problem should be noticed; *i.e.*, if the four boundaries in *Fig.1* were optimal for the *24-hour* control by five programs, the first three boundaries must be also optimal for controlling up to *20:00* by four programs, the first two boundaries must be optimal for controlling up to *15:30* by three programs, and so on. Otherwise, we can find the better timings for the first three switching timings, for example, and then these better timings are apparently better for the five-program operation for a whole day, too. This property is famous as *the Bellman's optimality principle*, which allows us to apply the dynamic programming method to obtain the solution of the second part.

3.2. Formulation of Dynamic Programming

Let $F_j(T_j)$ be the minimum total delay up to time T_j; that is, the minimum total delay associated with programs *1* through *j*. Based on *the Bellman's optimality principle*, the $F_j(T_j)$ can be written as the following recursive form:

$$F_j(T_j) = \underset{T_{j-1}}{Min.} \left[TD^*(T_j \mid T_{j-1}) + F_{j-1}(T_{j-1}) \right], \tag{3.1}$$

where $TD^*(T_j \mid T_{j-1})$ = the minimized total delay of program *j* which is used from time T_{j-1} to T_j .

The brackets *[.]* in the right hand side of *Eq.(3.1)* means that if program *j* were used from time T_{j-1} until T_j, the minimum total delay from time *0* until time T_j would be the sum of the minimum total delay up to program *j-1*, $F_{j-1}(T_{j-1})$, and the minimized total delay associated with program *j*, $TD^*(T_j/T_{j-1})$. We imagine that we have decided the switching timings of up to program *j-1* and spent the *past* delay of $F_{j-1}(T_{j-1})$. Since, given T_{j-1}, our *future* delay $TD^*(T_j / T_{j-1})$ is independent of the past switching timings, the recursive form is guaranteed. In order to determine $F_j(T_j)$, which is the minimum total delay up to program *j* provided that program *j* ends at time T_j, the brackets *[.]* should be thus minimized with respect to time T_{j-1}.

The recursive relationships are written from the beginning:

$$\begin{aligned}
j = 1 :&\quad F_1(T_1) = TD^*(T_1 / 0) + 0 \ , \\
j = 2 :&\quad F_2(T_2) = \underset{T_1}{Min.} \left[TD^*(T_2 / T_1) + F_1(T_1) \right], \\
j = 3 :&\quad F_3(T_3) = \underset{T_2}{Min.} \left[TD^*(T_3 / T_2) + F_2(T_2) \right], \\
&\ \ \vdots \\
j = J :&\quad F_J(T_J) = \underset{T_{J-1}}{Min.} \left[TD^*(T_J / T_{J-1}) + F_{J-1}(T_{J-1}) \right].
\end{aligned} \tag{3.2}$$

Since program 1 is assumed to start from time $k = 0$, the $F_1(T_1)$ is evaluated from $TD^*(T_1 / 0)$ for every T_1, $1 \le T_1 \le K-(J-1)$. Since programs 2 to J must be allocated on the time axis after program 1, it is sufficient to evaluate $F_1(T_1)$ until $T_1 = K-(J-1)$. For program 2, the $F_2(T_2)$ is evaluated for every T_2, $2 \le T_2 \le K-(J-2)$, by minimizing the sum of $TD^*(T_2 / T_1)$ and $F_1(T_1)$ with respect to T_1. In this minimization, the $TD^*(T_2 / T_1)$ should be only calculated, since $F_1(T_1)$ has been already evaluated for all T_1. That is, fixing the ending time of program 2, T_2, we need to calculate $TD^*(T_2 / T_1)$ for every T_1 ($< T_2$). Then, choose time $T_1 = T_1(T_2)$ when $TD^*(T_2 / T_1) + F_1(T_1)$ takes the minimum, where $T_1(T_2)$ is the best starting time of program 2 provided that program 2 ends at time T_2. To find time $T_1(T_2)$ for particular T_2 in the dynamic optimization process, about T_2 times we need to compare the $TD^*(T_2/T_1) + F_1(T_1)$. Therefore, to determine $T_1(T_2)$ for all T_2, we need approximately the total of $K^2/2$ dynamic optimizations (=comparisons of $TD^*(T_2 / T_1) + F_1(T_1)$), if $K >> J$.

Repeating the process until $j = J$, we could finally obtain $F_J(T_J)$ and the optimal switching timings of all programs. Since approximately $K^2/2$ dynamic optimizations are necessary for each j, $2 \le j \le J-1$, the total dynamic optimizations required is $(J-2)\cdot K^2 / 2$.

Although the application of the dynamic programming reduces the number of calculations substantially, it still requires a number of dynamic optimizations of the total delay. However, traffic engineers usually have *a priori* knowledge on times of switchings. For instance, if program *1* were used for traffic demand before the morning peak and program *2* for the morning peak traffic, we could roughly estimate that switching timing T_1 would be around *6:20* to *8:20*, etc. If this kind of *a priori* range of switching timings is incorporated, the computation time can be further saved.

If we apply non-overlapping *a priori* range of n intervals for each switching timing, the required number of dynamic optimizations would be about $(J\text{-}2)\cdot n^2$. *Table 1* describes the number of dynamic optimizations by several cases, where the total of *96* unit intervals $(K=96)$ and the five-program operation $(J=5)$ are assumed such as shown in *Fig.1*.

We have discussed the computation time in terms of the number of required dynamic optimizations. However, to complete the above procedure, the minimized total delay, $TD^*(T_j \,/\, T_{j-1})$ must be determined by implementing the convex programming discussed in section 2. When *a priori* knowledge is

Table 1. Comparison of the Number of Calculations

	# of Optimizations in the dynamic programming	# of Minimizations of the convex programming
All Combinations	3,183,545	4,278
Dynamic Programming (DP)	12,927	4,278
a priori Range of n = 8 without DP (2-hour range for each timing)	4,096	208
a priori Range of n = 8 with DP (2-hour range for each timing)	200	208

not used, we need to calculate $TD^*(T_j \,/\, T_{j-1})$ for almost all combinations of T_{j-1} and T_j, which are counted approximately $K^2/2$ even if the dynamic programming is applied. If, however, *a priori* knowledge of switching timings is incorporated, only a limited number of minimizations are required. If a non-overlapping *a priori* range of n intervals is specified for each switching timing, we need to evaluate $TD^*(T_j \,/\, T_{j-1})$ for combinations of n different T_{j-1}'s and n different T_j's; that is, n^2 minimizations are required for each j. *Table 1* also shows the number of necessary implementations of the convex programming.

One plausible way to reduce the number of minimizations would be the interpolation of the minimized total delays, where $TD^*(T_j \,/\, T_{j-1})$ are calculated for only certain time boundaries of T_j and T_{j-1} , and the minimized total delays for the rest of time boundaries are linearly interpolated. When we minimize the total delay for every m time boundaries of T_{j-1}'s and T_j's, the number of required minimizations clearly decreases to $1/m^2$. The appropriateness of the interpolation will be examined through a simple example in the next section.

Although we have so far implicitly assumed that program *1* starts always from time *0:00*, it is not practical. For example, a program for night time traffic would start from the late evening and last until the next early morning. We should consider a circular time axis as shown in *Fig.3* instead of the straight time axis so that time *24:00* is followed by time *0:00* of the next day. In this case, we simply repeat the same procedure as explained above but shifting the starting time of program *1* from $k = 0$ to K. Thus, the total number of required dynamic optimizations becomes $(J-2)\cdot K^3/2$, which is still considerably small number compared with all possible combinations, ${}_KC_J$, especially for large J.

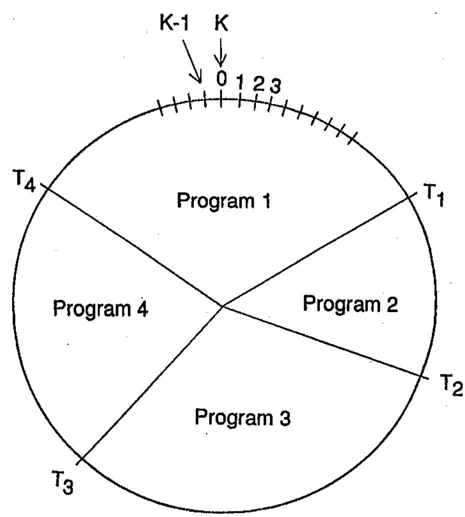

Fig. 3. Signal Programs on a Circular Time Axis.

3.3. Numerical Example

An example calculation is presented using an intersection of two one-way streets with a two-phase operation during *96* time intervals *(K=96)* as shown in *Fig.4*. The saturation flow rates are assumed *1500 [veh/hour]* for both streams and the total delay is estimated by *the Webster's 2-term formula.* Time-dependent demand flows of two streams during the *96* intervals are illustrated in *Fig.5*. Stream *1* is the major stream with the maximum normalized flow of about *0.65* and stream *2* with about *0.25* during the morning peak. We employ a four-program operation *(J=4)* such as in *Fig.3* and a four-hour *a priori* range is assumed for each of the switching timings as shown in *Fig.5*.

The results are summarized in *Table 2* and the optimal switching timings are also shown in *Fig.5*. We examine the interpolation method for the total minimized delays explained above; that is, the interpolation of $TD^*(T_j \,/\, T_{j-1})$ from the minimized total delays at every two and four time boundaries (m = 2 and *4*). The both interpolations seem appropriate, since not only the optimal switching timings but also the signal parameters for programs are very close

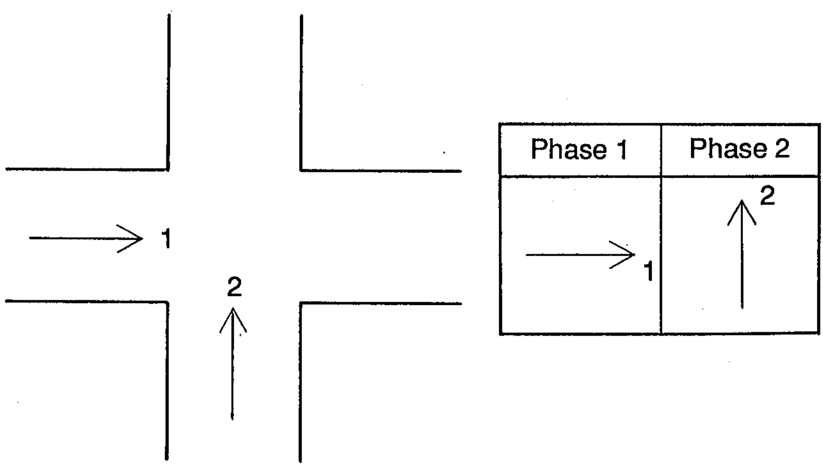

Fig. 4 An Intersection of Two One-way Streets.

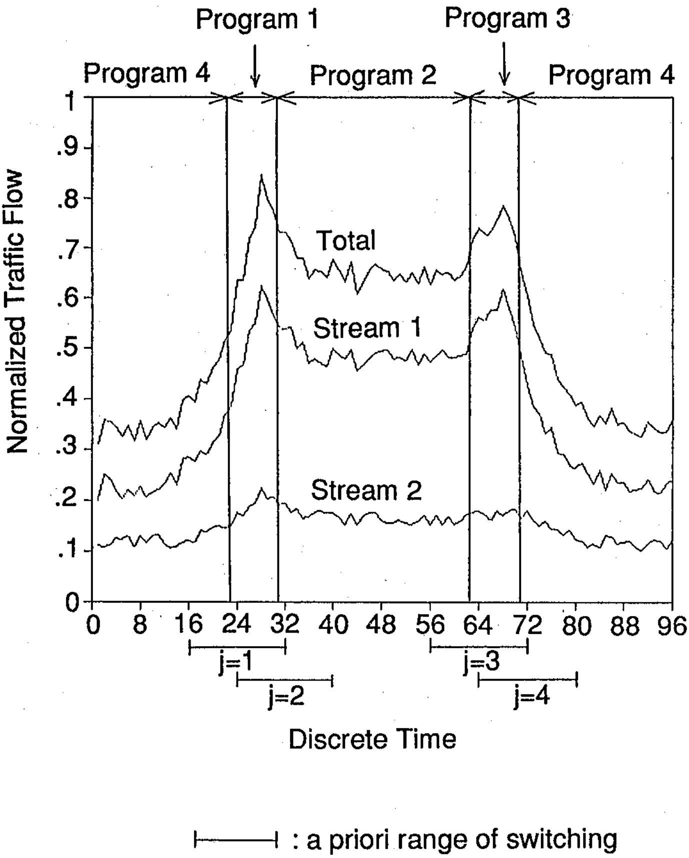

Fig. 5. Time-Dependent Demand Flows

to the optimum's obtained without using the interpolation. However, since the result apparently depends on the time-dependent demand flow variations and the signal phase plan, our conclusion cannot be firm but must be checked further.

All possible combinations of switching timings within the *a priori* ranges are counted *64,009* in this example and *Figure 6* illustrates distributions of the minimized total delays associated with these *64,009* combinations. We see that the distributions are getting smoother when the interpolation interval, m, increases from two to three and four. For the delays obtained without the interpolation, the minimum delay of *114.5 [hours]* is given by the timings shown in *Table* 2 but the maximum delay is about *127.5 [hours]*. Although we can roughly anticipate the ranges of switching timings from our experiences, we may have a chance to select switching timings which yield the larger delay by at most about *12 [%]*.

Table 2. Optimal Signal Parameters and Switching Timings

Program No.	Switching Timings Start	Switching Timings End	Cycle Time [sec]	Green Time Stream1 [sec]	Green Time Stream2 [sec]	Total Delay/ Program [hours]	Cumulative Total Delay [hours]
	23(5:45)	31(7:45)	108	72	26	21.43	21.43
1	22(5:30)	32(8:00)	106	70	26	24.94	24.94
	24(6:00)	32(8:00)	108	72	26	22.12	22.12
	31(7:45)	62(15:30)	69	43	16	47.13	68.56
2	32(8:00)	62(15:30)	66	41	15	44.80	69.74
	32(8:00)	64(16:00)	70	44	16	49.29	71.41
	62(15:30)	71(17:45)	93	63	20	20.36	88.92
3	62(15:30)	72(18:00)	91	61	20	21.94	91.68
	64(16:00)	72(18:00)	93	63	20	17.61	89.02
	71(17:45)	23(5:45)	41	21	10	25.55	114.47
4	72(18:00)	22(5:30)	39	19	10	23.12	114.80
	72(18:00)	24(6:00)	42	22	10	25.99	115.01

Upper figures : Without using the interpolation.
Middle figures : Interpolation of the total delays at every two time boundaries.
Lower figures : Interpolation of the total delays at every four time boundaries.
Lost time per cycle = 10 [sec]

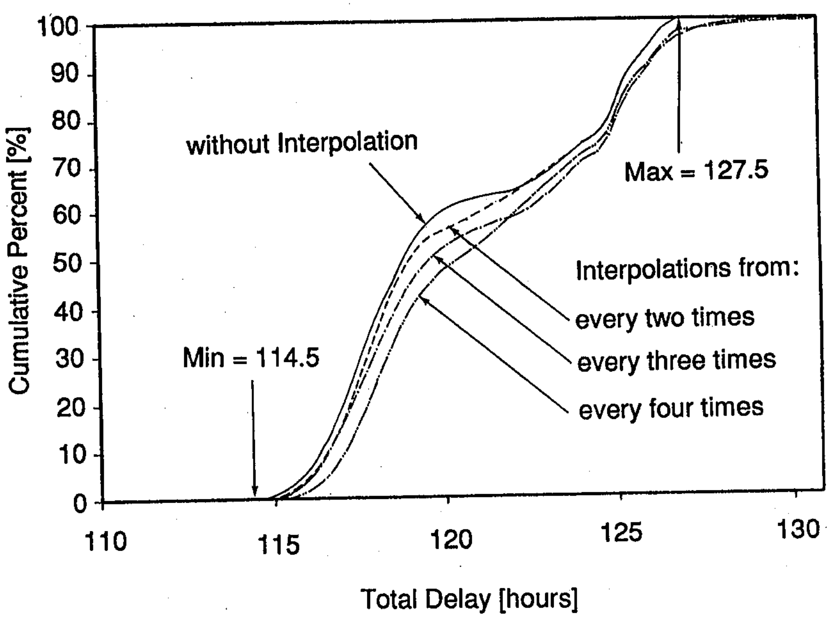

Fig. 6. Distributions of the Total Delays by Several Interpolations

3.4. Discussion on the Extension

Based on the dynamic formulation, let us briefly discuss the following two problems of the same structure as the above.

(1) The traffic engineers sometimes use the same program in two or more different time periods in practice, although we have assumed that a program is applied to a unique time period. We assume that at most J programs are available but can be switched S times in a day $(S > J)$. This is also a combination problem but the total number of all possible combinations is much larger than our original problem. Suppose that we have arbitrarily chosen S time boundaries of switchings on the circular time axis, and then want to allocate J programs to S different time periods provided that every

program must be used at least once. The total number of possible allocations is ${}_SC_J \cdot J^{(S-J)}$. In the original problem, this value is clearly *1* because S is equal to J. But if the number of switchings, S, gets increasing, we could expect much larger number of combinations than the original problem: approximately *1014* combinations for $K=96$, $J=5$, and $S=8$.

This problem cannot be formulated in the dynamic programming in the same manner as the original problem which uses time T_j as the decision variable. For instance, when we start the calculation from the beginning, $j=1$, the minimized total delay associated with program *1* cannot be determined because program *1* may be used twice or more in the different time periods.

However, this problem has also the property of holding *the Bellman's optimality principle.* Suppose that we have arbitrarily allocated programs *1* through j-*1* on the time axis, where one program is used several times just as shown in *Fig.7*. The optimal signal parameters and the associated total delays of programs *1* to j-*1* can be determined, since all time periods for these programs have been decided. We have to further allocate the remaining programs to the rest of the time periods and determine the total *future* delay associated with the remaining programs, but the *future* delay is clearly independent of what we have done, given time periods of programs *1* to j-*1*. Thus, this property can be again written in the recursive relationship below using the variable P_j which means a set of time periods for program j, instead of T_j as in the original problem:

$$F_j(P_j) = \underset{P_{j-1}}{Min.} \left[TD^*(P_j \,/\, P_{j-1}) + F_{j-1}(P_{j-1}) \right], \qquad (3.3)$$

where $F_j(P_j)$ = the minimized total delay of programs *1* through j,

$TD^*(P_j \,/\, P_{j-1})$ = the minimized total delay of program j which is used in time period P_j, given time periods of P_1 to P_{j-1}.

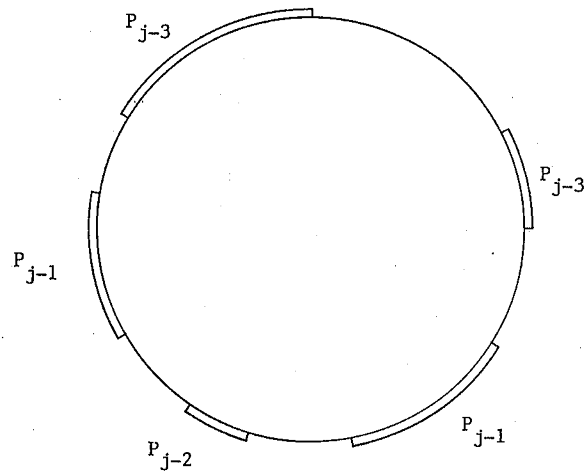

Fig.7. Time Periods of Program Operations on the Circular Time Axis

(2) In the pretime multi-program signal control, both the number of programs and switching timings are constrained as we have seen so far. However, we have a different type of signal control, the responsive program selection control, in which programs can be switched as many times as necessary depending on time-dependent traffic demand, although the number of available programs is limited. For this control, we do not have to consider the switching timings but we must determine to what demand flows each of the programs should be applied. Therefore, we must classify the demand flows so that each of the programs corresponds to the cluster of demand flows. For example, the time-dependent demand flows of streams *1* and *2* shown in *Fig.5* are described in the two-dimensional demand space as illustrated in *Fig.8*. The demand space should be divided into J clusters for J available programs.

This clustering problem is also written in the recursive form just as in *Eq.(3.3)*, since the *future* delay of $TD^*(P_j \,/\, P_{j-1})$ is again independent of the *past* delay of $F_{j-1}(P_{j-1})$, given P_1 to P_{j-1}. (Here, the P_j is considered as a set of demand flows controlled by program *j*.)

We recognize that the above two problems have the same structure as our original one. However, they likely require enormous calculations even if the dynamic programming is applied, since we must minimize the total delay for a huge number of different P_j's. Practically speaking, we may, therefore, utilize some strong constraints just like *a priori* ranges of switchings in the original problem to reduce the calculation in the optimization process of the dynamic programming.

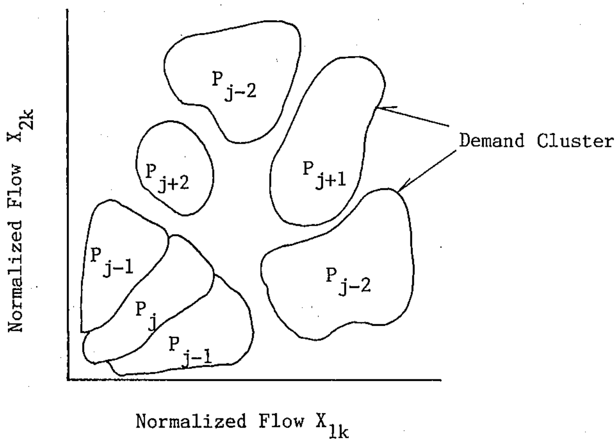

Fig.8. Flow Clusterings on the Two-Dimensional Demand Space

4. Summary

We deal with a pretime multi-program signal control at an isolated undersaturated intersection and discuss *(1)* what should be the optimal cycle time and green splits of each program, given a time period to which the program is applied, and *(2)* when one program should be switched to another. The major remarks are summarized below:

(1) The optimum signal parameters of a program are determined so as to minimize the total delay assuming that a time period of the program

operation has been decided. Following the previous studies, this problem is formulated in a minimization program which is convex in green splits and an inverse of the cycle time.

(2) We discuss how to determine the time periods of different programs so as to minimize the total delay of all programs throughout a day, given the number of programs available. For the simplicity, a program is assumed to be used only one time period. This problem is essentially the combination problem of selecting the switching timings on the time axis. We show that the problem can be reduced to the dynamic programming based on *the Bellman's optimality principle* and solved systematically in a short computation time. Furthermore, if we utilize *a priori* knowledge on times of switching; *i.e.*, a program for peak traffic would be applied for approximately from *6:20* until *8:20*, the computation time can be saved so much that a personal computer can handle the calculation.

(3) To apply the dynamic programming, however, the minimized total delays must be evaluated many times for various time periods of signal program operations by solving the convex programming. For the reduction of the computation time, we examine the interpolation of the total delays in which the total delay value of a time period is interpolated from the total delay values of the neighboring time periods. An examination of the interpolation through a simple example shows that the switching timings and signal parameters based on the interpolated delay values give a good agreement with the optimum's obtained without using the interpolation.

(4) For the future extensions, we recognize that our problem still has the same structure and could be reduced to the dynamic programming in principle, even if we relax the assumption mentioned above so that a program can be used two or more different time periods. We finally discuss the traffic responsive program selection control where the number of programs is limited but a program can be switched to another at any time. Namely, there is no constraint on the number of switchings. For this control, we must cluster the demand flows to which each of the programs is applied. This clustering problem is also shown to have the same structure.

References

1. Allsop R.E.:Estimating the Traffic Capacity of a Signalized Road Junction, *Transportation Research*, Vol.6, pp.245-255, 1972.

2. Allsop R.E. : Effects of Errors in Lost Times on the Delay to Traffic at an Isolated Road Junction Controlled by Fixed-Time Signals,*Transportation Research*, Vol.7, pp.145-162, 1973.

3. Zuzarte Tully I.M. : Synthesis of Sequences for Traffic Signal Controllers using Techniques of the Theory of Graphs, Ph.D Thesis, *University of Oxford Engineering Laboratory Report OUEL* 1189/77, 1977.

4. Heydecker B.G. and I.W.Dudgeon : Calculation of Signal Settings to Minimize Delay at a Junction, *Proceedings of 10 th International Symposium on Transportation and Traffic Theory*, pp.159-178, Elsevier, Boston, July, 1987.

5. Gallivan S. and B.G.Heydecker : Optimizing the Control Performance of Traffic Signals at a Single Junction, *Transportation Research B*, Vol.22B, No.5, pp.357-370, 1988.

6. Improta G. and G.E.Cantarella : Control System Design for an Individual Signalized Junction, *Transportation Research B*, Vol.18B, No.2,pp.147-167, 1984.

7. Cantarella G.E. and G.Improta : Capacity Factor or Cycle Time Optimization for Signalized Junctions: A Graph Theory Approach, *Transportation Research B*, Vol.22B, No.1, pp.1-23, 1988.

8. Moller K.: Calculations of Optimum Fixed-Time Signal Programs, *Proceedings 10 th International Symposium on Transportation and Traffic Theory*, pp.179-198, Elsevier, Boston, July, 1987.

9. Heydecker B.G.:A Note on the Interpretation of the Dual Variables in Estimating Traffic Capacity at Signal-Controlled Road Junctions, *Transportation Research B*, Vol.17B, No.5, pp.359-364, 1983.