International Journal of Computer Applications (0975 - 8887)
Volume 65 - No. 12, March 2013

Optimal Feasible Green Light Assignment to a Traffic Intersection using Intersection Graph

A. Bharali
Department of Mathematics
Dibrugarh University

A. K. Baruah
Department of Mathematics
Dibrugarh University

ABSTRACT

A feasible green light assignment is an assignment of time period to each traffic stream so that only compatible traffic streams are allowed to recieve overlapping time period. In case of an intersection, it is almost always possible to find several feasible green light assignments. We usually have some goal such as to minimise the total waiting time, to maximise the volume of flow etc., to be achieved. In this paper we will try to find an optimal feasible green light assignment which minimises the waiting time at the same time maximises the volume of flow, using intersection graph.

Keywords:

Compatibility Graph, Intersection Graph, Traffic Network

1. INTRODUCTION

The requirement of phasing traffic lights is to move traffic safely and efficiently. With increasing concern about time and energy use, the latter goal is becoming of incresed importance. Consider a traffic intersection at which we wish to install a new traffic light. Various traffic streams are approaching the traffic intersection. Certain traffic streams are judged to be compatible with each other, in the sense that they can be moving at same time without conflict. The decision about compatibility can be made ahead of time and may be based on estimated volume of traffic in a stream as well as traffic pattern. The compatibility relation can be represented by a graph known as *compatibility graph,* whose vertices are the traffic streams, and two streams are joined by an edge if they are judged to be compatible. In traffic light phasing, we wish to assign a period of time to each stream during which it recieves a green light, and to do it in such a way that only compatible traffic streams can get green lights at the same time. So there is a cycle of green and red lights, and it keeps on repeating again and again. We may think of time during the cycles as being kept on a large clock and the time during which a given traffic stream gets a green light corresponds to an arc on the circumference of the clock circle. Then a *feasible green light assignment* consists of an assignment of an arc of the circle to each traffic stream so that only compatible traffic streams are allowed to recieve overlapping arcs. In terms of compatibility graph, only vertices joined by an edge are allowed to recieve overlaping arcs. Suppose $X = \{A_1, A_2, ..., A_n\}$ is a family of sets, we can associate a graph with X, called the *intersection graph of X [7],* as follows: the vertices of this graph are the sets in X, and there is an edge between two sets A_i and A_j if and only if they have a nonempty intersection. The intersection graph of a family of intervals on the real line is called an *interval graph.* So, an intersection graph corresponding to a feasible green light assignment will be a subgraph of the compatibility graph and it must be a circular arc graph. If we require that no green light time period overlap the starting time i.e. a cycle begins with all red lights, then the intersection graph corresponding to a feasible green light assignment is a spanning subgraph of the compatibility graph which is an interval graph.

2. COMPATIBILITY GRAPH OF A TRAFFIC INTERSECTION

Before introducing compatibility, let us define *conflictness relation of traffic streams.* Also, it is necessary to know the trajectory, the path used by a traffic stream to traverse the intersection to the conflict area. Usually some pairs of traffic streams use, along a part of their trajectories, the same space on the intersection. These are the streams whose trajectories cross or merge. A conflict exists between such streams.

The set of all pairs of traffic streams that creates a conflict between elements of the pair represents the conflictness relation. Thus, the conflictness relation, C_1, can be defined as $C_1 \subset \tau \times \tau$, where τ is the set of traffic streams,

$C_1 = \{(\sigma_i, \sigma_j)|$ the trajectories of σ_i and σ_j cross or merge, $\sigma_i, \sigma_j \in \tau$ $\}$.

Hence, the nonconflict relation is the set of all pairs of traffic streams that are not mutually in conflict, i.e. $C'_2 = (\tau \times \tau) \setminus C_1$.

Also, there is a set of traffic stream pairs, which comprise *conditionally compatible* streams i.e. trajectories cross or merge but can simultaneously get the right-of-way, can be defined as [4]:

$C''_2 = \{(\sigma_i, \sigma_j) \in C_1|$ streams σ_i and σ_j can simulataneously get the right-of-way$\}$.

Now the compatibility relation of traffic stream pairs whose elements can simultanously get the right-of-way is: $C_2 = C'_2 \cup C'''_2$.

Again, if the set of streams that passes through the intersection without any conflict is denoted by τ', where $\tau' \subset \tau$, then the set of pairs of traffic streams that can simultaneously get the right-of-way is defined by [4]

$$C_3 = C_2 \setminus \{(\sigma_i, \sigma_j) \,|\, (\sigma_i, \sigma_j) \in C''_2, (\sigma_i \text{ or } \sigma_j \in \tau')\ \}.$$

Hence, the *compatibility relation* can be defined as [4]

$$C_c = C_3 \cup \Delta_S$$

where

$$\Delta_S = \{(\sigma_i, \sigma_i)|\, \sigma_i \in \tau\}$$

The relation C_c is symmetric and reflexive.

The *compatibility graph* of traffic streams is defined as the set of traffic streams, τ, and compatibility relation C_c [4], [9]:

$$G_c = (\tau, C_c)$$

Since set τ is finite, and relation C_c symmetric and reflexive, graph G_c is a finite, nonoriented graph with a loop at each node(may be omitted, if there is no confusion).

3. INTERVAL GRAPH: PROPERTIES AND CHARACTERISATION

Interval graphs arose from purely mathematical considerations *(Hajos(1957))* and independently, from a problem of genetics[1]. It is also easy to see that if G is an interval graph, then every generated subgraph[5] must also be an interval graph. A generated subgraph means a subgraph generated by a subset of the vertex set. However, this is not the case for every subgraph[7]. Thus, if G is an interval graph, it has the property that no graph Z_n, $n \geq 4$, is a generated subgraph. A graph G with this property is called a *rigid circuit graph* or a *triangulated graph [7].* But every rigid circuit graph is not an interval graph. The following theorem due to *Lekkerkerker and Boland* gives an if and only if condition which involves the concept of *asteroidal triple* [7]. Three vertices form an *asteroidal triple* in a graph G if for each two, there exists a path containing those two but no neighbour of the third. [10]

Theorem 3.1 A graph is an interval graph if and only if it is a rigid circuit graph and it has no asteroidal triple [5] [6].
There is yet a second characterisation of interval graphs, which is due to *Gilmore and Hoffman* and it needs the ideas of complement of a graph and transitively orientable graph [9].

Theorem 3.2 A graph G is an interval graph if and only if Z_4 is not a generated subgraph and G^c is transitively orientable [3].
There is another nice characterisation of interval graphs, which is due to *Fulkerson and Gross.* It requires the concepts of dominant clique vetex incidence matrix and consecutive 1's property of a matrix. If G is any graph then its dominant clique-vertex incidence matrix is defined as the matrix whose rows correspond to the dominant cliques(A clique is called dominant if it is maximal), and the columns to the vertices. The $(i, j)^{th}$ entry is 1 if the j^{th} vertex belongs to the i^{th} dominant clique, and it is 0 otherwise [7]. Again, A matrix A of 0's and 1's has the consecutive 1's property if it is posssible to permute the rows such that the 1's in each column appear consecutively [7].

Theorem 3.3 A graph is an interval graph if and only if its dominant clique-vertex incidence matrix has the cosecutive 1's property [2].
Since the identification of largest clique of a graph is itself an NP-hard problem, so it is not generally easy to identify the collection of dominant cliques of a graph.

4. PHASING THE TRAFFIC LIGHTS

Interval graphs and circular arc graphs have wider application in phasing the traffic lights. The traffic light phasing helps the traffic to move safely and at the same time efficiently. The sense of efficiency may be less waiting time(total amount of red light time in a circle), increased the volume of flow, less traffic jam etc. Now if we think about safety then we have to sacrifice efficiency, in some sense(more waiting time) or on the other hand if we think only about efficiency then there may be some dangerous consequences. So we have to choose a feasible green light assignment which is best suited for a given intersection. Now, what makes one feasible green light assignment better than the other ? We usually have some preferences depending on the intersection. For example we may wish to minimise the total amount of waiting time, or to minimise a weighted sum of red light times by weighting more heavily the red light time for heavily travelled traffic streams, or as *Stoffers* [8] points out, we might have some information about expected arrival times of different traffic streams, and we might wish to penalise starting time being far from the traffic stream's expected arrival time and minimise the penalties. Different procedure may be adopted for finding an optimal green light assignment for a given criterion.

5. A PROBLEM

Let us try to find an optimal feasible green light assignment which will minimise the waiting time at the same time maximise the flow of traffic. To find such an optimal feasible green light assignment, we must identify each interval graph(or circular arc graph) which is a spanning subgraph of the compatibility graph. For each of these interval graphs, find all the different consecutive ordering of dominant cliques and for each such ordering we must find an optimal solution of pahse durations. Then consider all these together to find an optimal solution for the entire graph. Let us illustrate this by considering the following intersection:

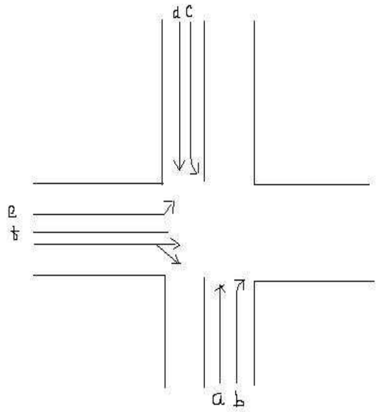
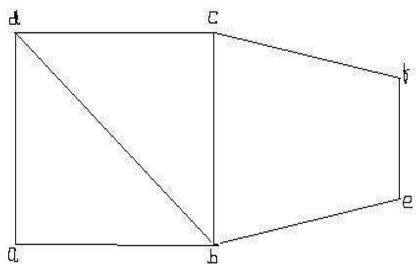


Fig 1. An Intersection

The corresponding compatibility graph G is as follows:



And some of the corresponding intersection graphs are:

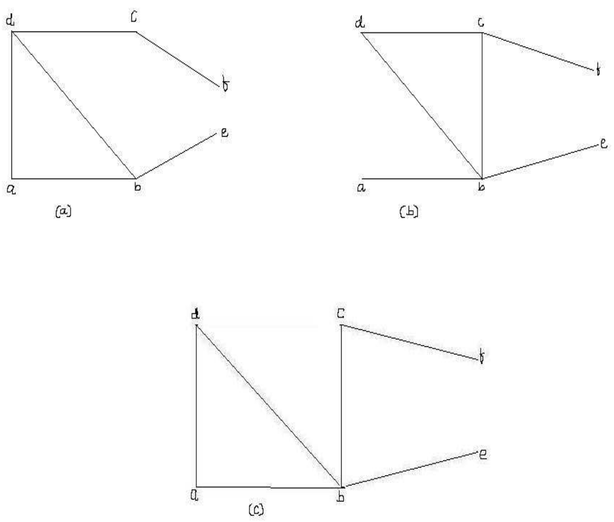


Fig 2. Some Intersection Graphs

Now, let us consider the first intersection graph in Fig. 2(a). A feasible green light assignment corresponding to this intersection graph is shown in Fig. 3:

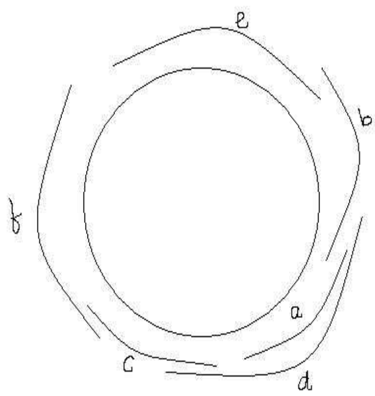


Fig 3. A feasible green light assignment

For concreteness, we handle the interval graph case. And by consecutiveness of the ordering, $K_1 = \{e, b\}$, $K_2 = \{b, a, d\}$, $K_3 = \{d, c\}$, $K_4 = \{c, f\}$ is one consecutive ordering of dominant cliques. Thus there are four phases. In phase 1, traffic streams e and b get green lights, then in pahse 2, streams b, a, d get green lights and so on. Suppose to each clique K_i, we assign a duration d_i. Now our aim is to find the values of these d_i's so that total red light times is as minimum as possible at the same time the flow of traffic is as maximum as possible. Let v_1, v_2, v_3, v_4, v_5, v_6 be the volumes(pcu/s)[4] of steams a, b, c, d, e, f respectively. The answer is obtained by observing the following: a gets red light during phases K_1, K_3 and K_4, so a's total red light time is $d_1 + d_3 + d_4$. Similarly b's, c's, d's, e's and f's red light times are $d_3 + d_4$, $d_1 + d_2$, $d_1 + d_4$, $d_2 + d_3 + d_4$ and $d_1 + d_2 + d_3$ respectively. And the volumes(pcu/s) of traffic that will be blocked during these red light times are $v_1 + v_2 + v_4$, $v_1 + v_2 + v_4 + v_5$, $v_3 + v_4 + v_6$, $v_1 + v_2 + v_3 + v_4$, $v_2 + v_5$ and $v_3 + v_6$ respectively. (When the stream a is blocked, all the phases i.e. K_i's containing a will be showing red light and so on.) So, we have to minimise
$Z = (d_1 + d_3 + d_4)(v_1 + v_2 + v_4) + (d_3 + d_4)(v_1 + v_2 + v_4 + v_5) + (d_1 + d_2)(v_3 + v_4 + v_6) + (d_1 + d_4)(v_1 + v_2 + v_3 + v_4) + (d_2 + d_3 + d_4)(v_2 + v_5) + (d_1 + d_2 + d_3)(v_3 + v_6)$
Here, all the values of v_i's are expected to be known. We also have to make the assumptions that

(1) green light time duration for each stream is of some minimum length, say t.
(2) The duration of a complete cycle is fixed, say T.

Since a's total red light time is $d_1 + d_3 + d_4$, so a's total green light time is d_2 and so on. So, the problem reduces to minimise Z subject to

$$\begin{aligned} d_2 &\geq t, \\ d_1 + d_2 &\geq t, \\ d_3 + d_4 &\geq t, \\ d_2 + d_3 &\geq t, \\ d_1 &\geq t, \\ d_4 &\geq t, \\ d_1 + d_2 + d_3 + d_4 &= T, \end{aligned}$$

and obviously each $d_i \geq lb$ where lb is some fixed lower bound of the time duration. Since all the values of v_i, t and T are known, so it won't be difficult to find the optimal solution of this LPP. Similarly for a different feasible green light assignment, we can construct a similar LPP and get another solution. From all these solution we can have an optimal feasible green light assignmnt. Then putting all these optimal solutions for different possible interval graphs of G which arises from different consecutive orderings of their dominant cliques, together to find an optimal solution i.e. optimal feasible green light assignment for the entire graph G and that will be the best suited green light assignment for the intersection shown in Fig. 1 satisfying our goal.

6. CONCLUSION

For relatively small graphs, this procedure seems to be very vital as it reduces a somewhat critical optimisation problem to a simple LPP. In case of larger graphs this process will also work but it will be costly as far as the time is concerned, as even identifying all the dominant cliques invloves a lengthy computation for larger graphs. However, in most of the real cases the graphs are relatively small(i.e. less number of streams). In this paper we consider a problem to find an optimal feasible green light assignment which will minimise the waiting time as well as maximise the flow of traffic. Since Z is a function of d_i's with the weights v_i's, for different purpose we may suitably choose the weights and a similar formulation of LPP can be obtained whose solution will give the required optimal assignment for that purpose.

7. ACKNOWLEDGEMENT

This Research work is funded by grants from the UGC, New Delhi, India as a Major Research Project awarded to A. K. Baruah.

8. REFERENCES

[1] S. Benzer, *On the topology of the genetic fine structure,* Proc. Nat. Acad. Sci. USA, 45, pp. 1607-1620, 1959.

[2] D. R. Fulkerson and O. A. Gross, *Incidence matrices and interval graphs,* Pacific J. Math., 15, pp. 835-855, 1965.

[3] P. C. Gilmore, A. J. Hoffman, *A characterization of comparability graphs and of interval graphs,* Canad. J. Math., 16, pp. 539-548, 1964.

[4] S. Guberinic, G. Senborn, B. Lazic, *Optimal Traffic Control: Urban Intersections,* CRC Press 2008.

[5] F. Harary, *Topics in Graph Theory,* New York Academy of Sciences, 1979.

[6] C. B. Lekkerkerker, J. C. Boland, *Representation of a finite graph by a set of intervals on the real line,* Fund. Math., 51, pp. 45-64.

[7] Fred S. Roberts *Graph Theory and Its Applications to Problems of Society,* SIAM, Philadelphia, 1978.

[8] K. E. Stoffers, *Scheduling of traffic lights-A new approach,* Transportation Research, 2, pp. 199-234, 1968.

[9] D. B. West, *Introduction to Graph Theory,* Prentice Hall, 1996.

[10] F. R. McMorris, Terry A. McKee, *Topics in Intersection Graph Theory,* SIAM, 1987.