A Solution Method for the Time Slot Assignment Problem in SS/TDMA Systems with Intersatellite Links

A SOLUTION METHOD FOR THE TIME SLOT ASSIGNMENT PROBLEM IN SS/TDMA SYSTEMS WITH INTERSATELLITE LINKS

Benny M. NYAMBO
University of Zimbabwe
Mount Pleasant, Harare
Zimbabwe
e-mail: nyambo@science.uz.ac.zw
and
Gerrit K. JANSSENS
Hasselt University
Agoralaan, B-3590 Diepenbeek
Belgium
Email: gerrit.janssens@uhasselt.be

ABSTRACT
The time slot assignment problem (TSAP) is considered in a cluster of satellite-switched time-division multiple-access (SS/TDMA) satellite systems with inter-satellite links. The TSAP is formulated as a graph coloring problem and leads to an integer programming formulation. Efforts are made to solve the problem by solving continuous linear programming methods and by rounding continuous values as integer optimal values. The interactive solution approach is satisfactory for real world problems.

KEY WORDS
SS/TDMA; inter-satellite link; integer programming; work scheduling

1. Introduction

The utilisation of natural resources in satellite communication is optimized by using of multi-beam antennas and satellite-switched time-division multiple-access (SS/TDMA) [1]. SS/TDMA system consists of a satellite with a multi-beam antenna covering several geographical zones and an on-board switch to provide connections between the uplink and downlink beams according to the TDMA frame. As usual in TDM systems communication takes place in a synchronous way and is organised in frames. The frame is composed of time slots. Packet scheduling is usually called time slot assignment (TSA for short). Each time slot represents a particular switching matrix configuration, which transmits a certain number of packets between the connected uplink and downlink beams without conflict [2].

Sometimes stations that need to communicate are in zones not covered by the same satellite. In such a case an SS/TDMA system with more than satellite is required. The satellites are linked by *inter-satellite links* (ISL) forming a satellite network. Stations in zones covered by the same satellite need a transponder at the satellite.

The proper assignment of traffic to time slots to avoid conflicts is very important. The whole idea of the time slot assignment problem (TSAP) is to schedule all the traffic in time slots with minimum duration. It can be stated simply as: Given a traffic matrix, find an assignment of minimum duration which minimizes the number of switching modes. A first algorithm for the SS/TDMA-TSAP has been formulated in [3]. Some of later developed algorithms are discussed in [4], for example [5,6]. This problem has been shown to be NP-complete even for quite restricted ISL patterns and simplified models [7]. For small switching times, the goal is to find the assignments which minimises the delay of packets in the traffic matrix [8].

2. Problem Description

It is conclusive from combinatorial mathematics [8] that, any traffic matrix with maximum line sum *m* can be represented as the sum of *m* switching matrices. Any matrix has an associated bipartite graph (called by the same name as the matrix) which is constructed as follows. Let one set of nodes correspond to rows of the matrix and the other set of nodes correspond to columns. Let there be an edge between any two nodes where the corresponding matrix position has a non-zero element.

Consider a set $S = \{1,2....,s\}$, from a cluster consisting of s satellites. Let satellite p cover a set of disjoint zones Z_p and the set of all zones is denoted by Z. The zones are partitioned into s groups, one for each satellite in the cluster. Traffic requirements among zones are given as an ISL matrix D, which is a |Z| x |Z| matrix with nonnegative integer entries. An entry d_{ij} of D represents the amount of traffic that must be transmitted from zone i to zone j, which are measured in time slot units.

A switching matrix is a traffic matrix with at most one non-zero entry in any line. A line of a matrix is one column or row of that matrix. Thus a switching matrix represents traffic that can be transmitted without conflict. A *time slot assignment* (or schedule) for traffic matrix D is a decomposition of D into switching matrices $D = S_1 + S_2 + ... + S_n$. The length of a switching matrix S_i is the magnitude L_i of the largest entry in S_i. The transmission of a schedule $S_1,S_2,..,S_n$ takes L consecutive time slots, where $L = L_1+L_2+.....+L_n$. A schedule for D is optimal if its length is minimal.

6							
	3						**3**
		6					
			3		**3**		
			3				**3**
						6	
	3				**3**		
				6			

(a) An ISL traffic matrix D

3							
	3						
		3					
			3				
							3
						3	
					3		
				3			

(b) Switching matrix S_1

3							
		3					
					3		
						3	
	3						
				3			

(c) Switching matrix S_2

							3
			3				

(d) Switching matrix S_3

Fig. 1: Example of an ISL traffic matrix and its three switching matrices

In Figure 1, an example, taken from [1, figure 1] is shown, which has two satellites, each covering four disjoint zones. The traffic requirements among the zones are represented by an 8x8 ISL traffic matrix D. The traffic matrix is then decomposed into three switching matrices S_1, S_2 and S_3, and the length of the schedule is 9.

3. Formulation as a graph towards an optimisation problem

Lee and Park [1] interpret the TSAP as a coloring problem on a bipartite multigraph G, which is constructed from the ISL traffic matrix D as follows:

1. Let zone i be represented by two nodes. One is called the source node i and the other the destination node i.
2. If there are traffic requirements from zone i to zone j, source node i and destination node j are joined by edges. The number of edges (multiplicity) between two nodes represent the number of traffic requirements from node i to node j (denoted as d_{ij}).

The resulting graph $G=(V,E)$ is a bipartite multigraph, where V is the node set and E is the edge set of G. As both satellites cover four disjoint zones, there are eight source nodes and eight destination nodes. The first and third zone of satellite A show only intra-zonal requirements. All other zones show inter-zonal requirements. The fourth zone of satellite B has only requirements for a zone covered by satellite A (its first zone). The other zones show requirements for or from zones from the other satellite.

The TSAP is formulated as an optimisation problem on G. The TSAP is a decomposition of D into switching matrices, transmitting without conflicts, which means that traffic represented by two adjacent edges in G cannot take part in the same switching matrix. In graph-theoretical terminology, the decomposition of D is represented by a coloring on G. An edge coloring of a graph is an assignment of colors to the edges with the property that no adjacent edges have the same color. A minimum coloring is a coloring which uses as few colors as possible. The number of colors represents the number of needed time slots in the corresponding schedule for D.

In an edge coloring solution, a set of edges can have the same color if and only if the set is a matching, with a matching of a graph *G* as a subset of edges such that no two edges in the subset are adjacent.

Lee and Park [1] define a switching configuration on a graph *G* as a switching matrix such that all entries have the value 1 and has at most one non-zero entry in each row and column. A switching configuration can be represented as a matching on *G*, but not all matchings are feasible due to a limited number of transponders in ISL available. An acceptable matching is defined as a matching which represents a switching configuration. Therefore, TSAP is equivalent to finding a minimum coloring of *G* such that each color forms an acceptable marking.

In [1] TSAP is formulated as an integer programming problem. They consider a weighted graph *G' = (V,E')* obtained from the multigraph *G = (V,E)* by representing multiple edges between two nodes in *G* as an aggregate edge. The weight w_e of an edge *e* in *G'* is the multiplicity of the edges between two end nodes of *e* in *G*. A matching of *G* is also a matching of *G'*.

Let x_k be the decision variable denoting the number of colors to be assigned for the acceptable matching *k*. The decision variable has the meaning of the number of time slots which are required in the switching configuration, represented in the acceptable matching *k* in a schedule for *D*. Let the coefficient a_{ek} be 1 if and only if edge *e* is contained in the acceptable matching *k*; and let it be 0 otherwise. With these definitions the TSAP can be formulated as:

$$min \sum_{k \in M} x_k$$

subject to

$$\sum_{k \in M} a_{ek} . x_k \geq w_e \quad \forall e \in E' , \qquad (1)$$

$$x_k \in Z^+ \qquad \forall k \in M ,$$

where *M* is the set of all acceptable matchings.

4. Solution of the TSAP optimisation problem

The programming problem (1) can be solved by a general integer linear programming algorithm or by a specific branch-and-bound method developed by Henderson and Berry [10]. One should be aware that in programming problem (1) the number of variables is growing very fast with increasing problem size, which means that the use of an integer linear programming algorithm might become prohibitive. Also, obviously, a restriction on the total number of time slots cannot be introduced.

In the following paragraphs it is aimed to reformulate the optimisation problem in the case where the total number of time slots is fixed. In case such a less complex formulation may be found, the optimisation program can be run by the user for several trial values of the total number of time slots, specified in advance by the expert user. Inspiration for the development of this section is based on [10]. So the key constraint, involving a fixed number *p* of time slots, is:

$$\sum_{k \in M} x_k = p .$$

The optimisation problem can be formulated as:

$$min \ Z = \sum_{e \in E'} U_e - H \, Y$$

subject to

$$\sum_{k \in M} a_{ek} . x_k + U_e \geq w_e + Y \quad \forall e \in E' \qquad (2)$$

$$\sum_{k \in M} x_k = p$$

$$x_k \in Z^+ \qquad \forall k \in M$$

$$Y \in Z$$

$$U_e \ boolean \qquad \forall e \in E' .$$

In this formulation some new variables have been introduced. The variable U_e has the meaning of the number of time slots which would need to be added in the matching *e* in order to meet the multiplicity w_e. The variable *H* denotes a very large positive number in order to force the variable *Y* to take value 0. The variable *Y* has been introduced to generate a suitable scheduling pattern for any value of the fixed number *p* of time slots. The solution of this problem is not necessary a solution to the original problem, but a solution can be created from it. If *Y* = 0, the Boolean variable might take a 1-value, which means one (and only one) time slot needs to be added. In case *Y* = -1 in the optimal solution, more than one slot needs to be added, what we try to avoid. In case *Y* = +1 too many time slots for whole schedule are not allowed.

In terms of computational complexity, however, the model is not easy because of the extra 0-1 (Boolean) variables U_e which are added. If fast response is required to such an optimisation problem, it might be that an integer linear programming software (making use of branch-and-bound techniques) does not find the optimal solution in time. Therefore, a simpler, alternative model is suggested:

$$max\ \ Y$$

subject to

$$\sum_{k\in M} a_{ek}.x_k + w_e \geq Y \quad \forall e \in E^{'} \qquad (3)$$

$$\sum_{k\in M} x_k = p$$

$$x_k \in Z^{+} \qquad \forall k \in M\ .$$

Problem (3) is a maximin problem as it maximises some smallest difference between a supply and a demand. When the optimal value of *Y* takes the value +1, all schedule requirements are met with at least one extra slot. The model is simple and contains hardly more variables to be decided upon then model (1). The variable *Y* which should take values either -1, 0 or 1 can be defined to be integer in problem (3). If, for some reason, this might be impossible or due to time constraints, the variable *Y* can be relaxed to be a continuous variable. If the value of the continuous variable is larger than 1, the number of time slots in most practical situations may be the one specified in advance.

3. Conclusion

The time slot assignment problem can be formulated as an integer programming problem. By fixing the number of time slots the problem seems to become harder instead of simpler. An integer formulation is given for the case of a fixed number of time slots. This models however does not always determine the best switching pattern. Reallocation of surpluses or shortages is sometimes required, but this can be done quite easily making use of a continuous linear programming problem.

References

[1] T. Lee, S. Park, An integer programming approach to the time slot assignment problem in SS/TDMA systems with intersatellite links, *European Journal of Operational Research, 135,* 2001, 57-66.

[2] S. Kim, S.-H. Kim, Time slot assignment in a heterogeneous environment of a SS/TDMA system, *International Journal of Satellite Communications, 15*, 1997,197-203.

[3] T. Inukai, An efficient SS/TDMA time slot assignment algorithm, *IEEE Transactions on Communications, COM-27 (10),* 1979, 1449-1455.

[4] A. Ganz ,Y. Gao, Efficient algorithms for SS/TDMA scheduling, *IEEE Transactions on Communications, COM-40,* 1992, 1367-1374.

[5] C.A. Pomalaza-Raez, A note of efficient SS/TDMA assignment algorithms, *IEEE Transactions on Communications, COM-36(9),* 1078-1082.

[6] M. Dell'Amico, F. Maffioli, M. Trubian, New bounds on optimum traffic assignment in satellite communication, *Computers and Operations Research, 25,* 1998, 729-743.

[7] I. Gopal, C. Wong, Minimizing the number of switchings in an SS/TDMA system, *IEEE Transactions on Communications*, *COM-33*, 1985, 497-501.

[8] T. Weller, B. Hajek, Scheduling non-uniform traffic in a packet switching system with small propagation delay, *IEEE/ACM Transactions on Networking, 5(6),* 1997, 813-823.

[9] W.B. Henderson, W.L. Berry, Heuristic methods for telephone operator shift scheduling: an experimental analysis, *Management Science, 22,* 1976, 1372-1380.

[10] D. L. van Oudheusden, W. Wen-Jenq, Telephone operator scheduling with a fixed number of operators, *European Journal of Operational Research*, *11*, 1982, 55-59.