An Integer programming formulation for optimal deployment of ISIS protocol

Aykut Özsoy, *Université Libre de Bruxelles, Département d'Informatique*
Martine Labbé, *Université Libre de Bruxelles, Département d'Informatique*

November 9, 2006

Keywords: ISIS routing protocol, integer programming

1. Introduction

In this paper, we investigate the optimal deployment problem of an Internet routing protocol called Intermediate System - Intermediate System (shortly, ISIS). We introduce for this problem an integer programming formulation which we use in branch-and-cut, and report computational experience.

The ISIS protocol governs the routing[1] activities within the Autonomous Systems (AS).[2] It is based on dividing the set of all routers in an (AS) into subsets, which are called *areas*. Any two routers in an area must be able to reach each other over a path that is completely included in the area. That is, if we represent the AS by a graph where the vertices are the routers and the links are the edges, an area is, in a sense, a connected subgraph.

In the ISIS, each router in an AS belongs to one of three types: L1, L2 or L1/L2. All of the areas but one consist mainly of the L1 routers. The exceptional area consists only of the L2 routers. The areas that consist mainly of the L1 routers are called *L1 areas*, and the area that is composed only of the L2 routers is called the *L2 area*. The L1 routers in an L1 area can communicate directly with each other. They can not communicate directly with the other routers outside the area they belong to. L1 routers in two separate L1 areas communicate with each other via the L2 area. In other words, the L2 area is a backbone area that connects the L1 areas to each other.

Each L1 area has to have special routers that can make the communication between the L1 routers and the L2 area possible. Because, otherwise, L1 routers would not be able to send traffic out of their areas. L1/L2 routers are responsible for carrying out this liaison task, that is, they make the communication between the L1 routers and the L2 area possible. An L1/L2 router is capable of directly communicating with both of L1 and L2 routers. For this reason, they act like the gateways of the L1 areas. From here on, we will use the terms L1/L2 router and *gateway* interchangeably.

The L1/L2 routers can also directly communicate with the other L1/L2 routers in other L1 areas. Hence, even if no L2 area exists in the AS, only the L1/L2 routers can as well provide seamless communication between the L1 areas. In this case, the backbone would be composed only of the L1/L2 routers and the subgraph induced by the L1/L2 routers (i.e., the backbone) would have to be connected. For the sake of simplicity, in this paper we adopt this implementation option of the protocol: we do not consider the L2 routers at all and assume that all inter-area communication is facilitated by the L1/L2 routers.

In this implementation, if an L1 router in an area wants to send packets to an L1 router in another area, it first forwards them to a gateway of its area (i.e., one L1/L2 router located within its area). This gateway router then sends the packets over the backbone towards the area that the sink router belongs to. Finally, gateway L1/L2 router of the sink area receives the packets and sends them to the sink.

In the ISIS, the L1 routers are conservative in their selection of gateways to communicate with outside of their areas. While sending packets out of its area, an L1 router forwards the packets always to a specific,

[1] *Routing* is defined as the process of determining the best route for sending packets toward their destinations in the Internet.
[2] An AS is a network administered by a single authority in the Internet. In other words, an AS is a private network in the Internet.

predetermined gateway (i.e., an L1/L2 router located within the area) even if it is possible to choose a 'better' path via some different one. This is because an L1 router can never see the path that the packets follow outside its area, and it just optimizes the path that the packets follow inside. The gateway router that is chosen by an L1 router is, most of the time, the closest L1/L2 router in the area. In our integer programming formulations, we will allow the routers choose gateways other than the closest one. We will assign each L1 router to one gateway router (i.e., an L1/L2 router within its area), and force them to communicate with outside of their areas through the gateways they are assigned to. Clearly, it is meaningless to assign L1/L2 routers to gateways other than themselves (i.e., each L1/L2 router is assigned to itself for communicating with outside of its area).

This introduction to ISIS is sufficient for our purposes in this paper. Interested reader is referred to [2] for a detailed treatment of ISIS.

Having set the grounds, we are now ready to introduce the optimal deployment problem of ISIS. It consists of

1. partitioning the AS under consideration into areas (i.e., partitioning the set of routers in the AS into subsets),
2. deciding which router type each router should belong to (i.e., L1 or L1/L2)
3. routing the traffic within the AS (i.e., deciding the path of traffic flow for each source-sink pair given)
4. with the objective of minimizing the total cost of routing.

This problem retains features from graph partitioning problems, location problems and multi-commodity flow problems. In this paper, we introduce an integer programming formulation for solving this problem.

2. Notation

Let digraph $D = (V, A)$ represent an AS (V stands for the set of routers in the AS, and A stands for the set of links between the routers). We assume that the set A is symmetric throughout (i.e., $(j, i) \in A$ for all $(i, j) \in A$). Let the set T contain all source-sink pairs (u, v) which have traffic in-between, where u is the source and v is the sink. Let $d_{u,v}$ be the amount of required traffic flow between each source-sink pair $(u, v) \in T$. Let $c^{u,v}_{i,j}$ denote the cost incurred when unit flow associated with the pair $(u, v) \in T$ passes through the arc $(i, j) \in A$. The node-arc incidence matrix of D is denoted by $N_{|V|\times|A|}$; $(Nf^{u,v})$ is the product of the matrix N by a flow vector $f^{u,v} \in \mathcal{R}^{|A|}$, and $(Nf^{u,v})_i$ is the ith row of this product which corresponds to a node $i \in V$. The elements of the matrix N are defined as follows:

$$\forall u \in V,\, a \in A,\, N_{u,a} = \begin{cases} 1, & \text{if } a \text{ is an outgoing arc of } u, \\ -1 & \text{if } a \text{ is an ingoing arc of } u, \\ 0 & \text{if } a \text{ is not adjacent to } u. \end{cases}$$

The ith element of the vector $(Nf)_i$ gives the net flow (i.e., the total outgoing flow minus total ingoing flow) for node i.

3. The Integer Programming Formulation

Partitioning and location constraints:

Variables: We start with the variables. The first set of variables are the partitioning variables: $w_{u,v} \in \{0, 1\}$ for all $u, v \in V$ such that $u \neq v$, where,

$$w_{u,v} = \begin{cases} 1, & \text{if } u \text{ and } v \text{ are in the same area;} \\ 0, & \text{otherwise.} \end{cases}$$

The second set of common variables used in the formulation are $x_{u,v}$ variables ($x_{u,v} \in \{0,1\}\ \forall u,v \in V$):

$$x_{u,v} = \begin{cases} 1, & \text{if router } u \text{ is assigned to the L1/L2 router } v \text{ for communicating} \\ & \text{with outside of its area (i.e., if router } u \text{ uses router } v \text{ as the gateway} \\ & \text{router to the outside of the area);} \\ 0, & \text{otherwise.} \end{cases}$$

Setting $x_{u,u} = 1$ or $x_{u,u} = 0$ implies configuring the router u as an L1/L2 router or an L1 router, respectively.

Constraints: We first introduce the following sets of constraints:

$$\begin{aligned} w_{u,v} + w_{u,t} - w_{v,t} &\leq 1 \qquad \forall u,v,t \in V : u \neq v, u \neq t, v \neq t && (1) \\ \sum_{v \in V - \{u\}} w_{u,v} &\leq F_U - 1 \qquad \forall u \in V, && (2) \\ \sum_{v \in V - \{u\}} w_{u,v} &\geq F_L - 1 \qquad \forall u \in V, && (3) \\ \sum_{v \in V} x_{u,v} &= 1 \qquad \forall u \in V, && (4) \\ x_{u,v} &\leq x_{v,v} \qquad \forall u,v \in V : u \neq v, && (5) \\ \sum_{u \in V} x_{u,u} &\leq Y, && (6) \\ x_{u,v} + x_{v,u} &\leq w_{u,v} \qquad \forall u,v \in V : u < v. && (7) \end{aligned}$$

The areas are partitioned into areas by means of the constraints (1), the so-called triangle inequalities. These constraints are first used by Grötschel and Wakabayashi [1] for the clique partitioning problem. Consider three different routers u, v and t. In the triangle inequalities, if $w_{u,v} = 1$ (i.e., if u and v are in the same area) and if $w_{u,t} = 1$ (i.e., if u and t are in the same area) we ensure that $w_{v,t}$ is set to 1, too. The constraints (2) and (3) impose upper and lower bounds F_U and F_L respectively on the sizes of the areas. In constraints (4) we ensure that each router is assigned to a gateway to communicate with routers outside their areas. In these constraints, when $x_{u,u} = 1$ for a router u, then we do not assign it to any other router (i.e., $x_{u,v} = 0$ for $v \in V - u$); but, when $x_{u,u} = 0$, we are forced to assign it to a gateway. The constraints (5) prevents an L1 router from being assigned to a non-gateway router (i.e., an L1 router). The constraint (6) imposes an upper bound Y on the number of L1/L2 routers that can be used in the implementation. Finally, the constraints (7) ensures that each L1 router is assigned to a gateway that lies within the same areas as itself.

These constraints retain graph partitioning and location features. The multi-commodity flow aspect of the problem is incorporated into the model by means of the sets of constraints introduced in the next section.

Multi-commodity flow constraints: We exploit the fact that the inter-area flows can be analyzed in three portions: 1. from the source node to the gateway (L1/L2 router) it is assigned to (i.e., the flow within the source area); 2. from the gateway of the source area to the gateway in the sink area to which the sink router is assigned to (i.e., the flow on the backbone of L1/L2 routers); 3. the flow from the sink router's gateway to the sink router.

This formulation approach employs the idea of replicating the network in three layers as shown in Figure 1. The sets of nodes and the arcs in the three layers are identical. In addition to the arcs in the three layers, we also have arcs between identical nodes of the subsequent layers. Namely, there is an arc from the node i of Layer 1 to the node i of Layer 2 for each node in V. Also, there is an arc from the node i of Layer 2 to the node i of Layer 3 for each node in V. Note that, unlike the arcs within the layers, these arcs are all in one direction. It is only possible to send flow from Layer 1 to Layer 2 and from Layer 2 to Layer 3, not in the opposite directions. These arcs are defined to be costless.

If the source and the sink routers, say (u,v), lie in different areas, then the flow starts at the source node u in the Layer 1, passes through the Layer 2 and ends at the sink node v of the Layer 3. If u and v are in the same area, then the flow starts and ends at u and v nodes of Layer 1. Layer 2 accommodates the backbone and is utilized only in flows between areas. In other words, Layer 1 hosts the first portions of the inter-area flows and all the intra-area flows, Layer 2 hosts the second portions of the inter-area flows and Layer 3 hosts the third portions of the inter-area flows.

Variables: In this formulation, besides the ones introduced in Section 3., we use five sets of flow variables to represent the flows within the layers and between the layers. The variables $f_{i,j}^{u,v}$, $g_{i,j}^{u,v}$ and $h_{i,j}^{u,v}$ represent the flows within Layer 1, Layer 2 and Layer 3, respectively. The $t_i^{1,u,v}$ and $t_i^{2,u,v}$ variables are employed for the inter-area flows to pass from Layer 1 to Layer 2, and from Layer 2 to Layer 3, respectively. Formal definition of the $f_{i,j}^{u,v}$ variables are given below. This definition also applies to $g_{i,j}^{u,v}$ and $h_{i,j}^{u,v}$, if 'Layer 1' is replaced with the appropriate layer label.
$f_{i,j}^{u,v} \in \{0,1\} \qquad \forall (u,v) \in T, \forall (i,j) \in A;$

$$f_{i,j}^{u,v} = \begin{cases} 1, & \text{if flow from the source } u \text{ to the sink } v \text{ passes through the arc } (i,j) \text{ in Layer 1;} \\ 0, & \text{otherwise.} \end{cases}$$

Now, we give the definition for $t_i^{1,u,v}$ variables. Similarly, this definition can be used for $t_i^{2,u,v}$, too, if the layer labels are changed accordingly.
$t_i^{1,u,v} \in \{0,1\} \qquad \forall (u,v) \in T, \forall i \in V;$

$$t_i^{1,u,v} = \begin{cases} 1, & \text{if flow from the source } u \text{ to the sink } v \text{ passes from Layer 1 to Layer 2 over node } i \\ & \text{(i.e., through arc } (i,i) \text{ between the two layers);} \\ 0, & \text{otherwise.} \end{cases}$$

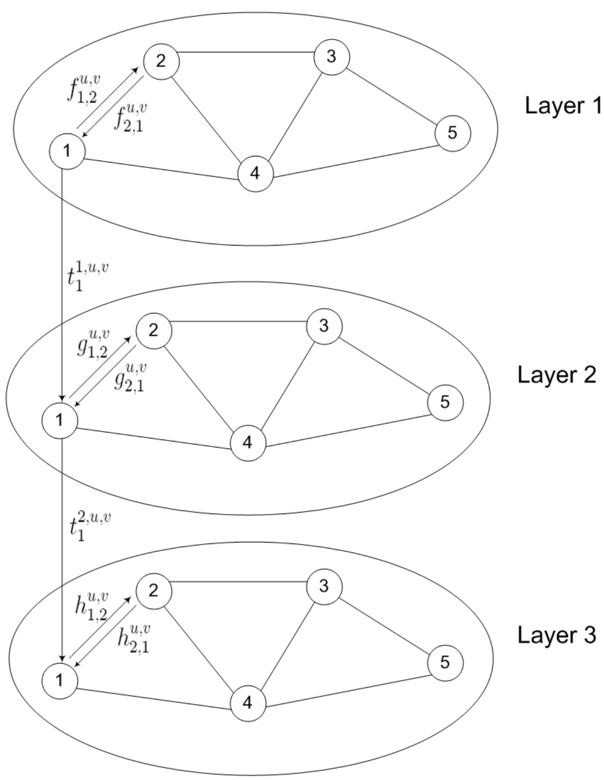

Figure 1: The variables of the Formulation 1 are illustrated on an example graph. The variables $f_{i,j}^{u,v}$, $g_{i,j}^{u,v}$ and $h_{i,j}^{u,v}$ are illustrated for the edge $(1,2)$. The variables $t_i^{1,u,v}$ and $t_i^{2,u,v}$ are illustrated for the node 1.

We now present the MIP formulation we introduce for the problem. Multi-commodity aspect of the problem is taken into consideration by means of the constraints (8)-(14).

$$
\begin{aligned}
\min \quad & \sum_{(u,v)\in T} \sum_{(i,j)\in A} c_{i,j}^{u,v} d_{u,v} (f_{i,j}^{u,v} + g_{i,j}^{u,v} + h_{i,j}^{u,v}) \\
\text{subject to} \quad & \text{the constraints } (1) - (7),
\end{aligned}
$$

$$
(Nf^{u,v})_i + t_i^{1,u,v} = \begin{cases} 1 & \text{if } i = u \\ 0 & \text{if } i \in V - \{u,v\} \\ -w_{u,v} & \text{if } i = v \end{cases} \quad \forall (u,v) \in T \tag{8}
$$

$$
(Ng^{u,v})_i + t_i^{2,u,v} - t_i^{1,u,v} = 0 \qquad \forall i \in V, \forall (u,v) \in T \tag{9}
$$

$$
(Nh^{u,v})_i - t_i^{2,u,v} = \begin{cases} w_{u,v} - 1 & \text{if } i = v \\ 0 & \text{if } i \in V - \{v\} \end{cases} \quad \forall (u,v) \in T \tag{10}
$$

$$
t_i^{1,u,v} \leq x_{u,i} \qquad \forall (u,v) \in T, \forall i \in V \tag{11}
$$

$$
t_i^{2,u,v} \leq x_{v,i} \qquad \forall (u,v) \in T, \forall i \in V \tag{12}
$$

$$
\sum_{j\in V:(i,j)\in A} g_{i,j}^{u,v} \leq x_{i,i} \qquad \forall (u,v) \in T, \forall i \in V \tag{13}
$$

$$
f_{i,j}^{u,v} + f_{j,i}^{u,v} + h_{i,j}^{u,v} + h_{j,i}^{u,v} \leq w_{i,j} \qquad \forall (u,v) \in T, \forall (i,j) \in E \tag{14}
$$

The constraints (8) are the flow conservation constraints for Layer 1: for each source-sink pair (u, v) it is ensured that the flow going out of the source node u in Layer 1 is 1; and for the intermediate nodes in Layer 1 it is ensured that the ingoing flow and the outgoing flow are equal. If u and v are in the same partition, then the right hand side value of the last constraint in (8) (i.e., the one with $i = v$) takes value -1, and this means the sink v in Layer 1 will absorb the flow sent by the source u. However, if u and v are in different partitions, then the right hand side value of this constraint will be equal to 0 and the sink v in Layer 1 will not accept the flow sent by u. Note that if u and v are in different partitions, the flow sent by u is meant for the sink node v in Layer 3.

The constraint (9) are the flow-balance equations of Layer 2. Since Layer 2 represents the backbone, all the ingoing flow and the outgoing flow must be equal for all nodes of Layer 2. The constraints (10) stand for the flow-balance equations of Layer 3. In these constraints, for each source-sink pair (u, v), all the nodes that are different from the sink v has 0 flow-balance. If u and v are in different areas, then the right hand side of the last constraint in (10) (i.e., the one with $i = v$) takes value -1 and this means the sink node v in Layer 3 accepts one unit flow sent by the source in Layer 1. Otherwise, the right hand side value is equal to 0, thus the node v in Layer 3 does not accept the flow sent by the source in Layer 1. Note that, the constraints (8) ensure that if the source u and the sink v are in the same area then the sink in Layer 1 accepts the flow; if they are in different areas then the sink v in Layer 3 accepts the flow.

While sending flow from a source u to a sink v, we must ensure that the flow gets onto the backbone on the L1/L2 router that the source is assigned to, and gets off the backbone on the L1/L2 router that the sink is assigned to. The constraints (11) and (12) stand for this requirement. The flow from u to v is allowed to pass from Layer 1 to Layer 2 over the node i only if it is a gateway and the source u is assigned to it. Similarly, the flow is allowed to pass from Layer 2 to Layer 3 over the node i only if it is a gateway and the sink v is assigned to it.

The constraints (13) guarantee that the backbone is composed of the L1/L2 routers. More specifically, in (13), if a router is not an L1/L2 router then it is going to receive no flow from the other routers in the Layer 2.

The constraints (14) say that all the flow in Layer 1 and Layer 3 must be realized between routers that are in the same area.

4. Conclusion

In this long abstract, our aim has been to present a short introduction to the ISIS deployment problem and an Integer Programming formulation we have developed for it. This formulation has 14 sets of constraints, which are all necessary to accommodate all features of this complex problem. Although complex, this model also retain some nice properties, such as, even if we relax the integrality restrictions of all the flow variables and the $x_{u,v}$ variables for $u \neq v$, we always end up with an integral optimal solution. In the paper we elaborate on this formulation, and use it in a branch-and-cut scheme to solve the problem to optimality.

References

[1] M. Grötschel, Y. Wakabayashi, "Facets of the Clique Partitioning Polytope", *Mathematical Programming* 47, pp. 367–387, 1990.

[2] R. White, A. Retana, *IS-IS: Deployment in IP Networks*, Addison-Wesley, Boston, 2003.