An Efficient IP Based Approach for Multicast Routing Optimisation in Multi-homing Environments

N. Wang, G. Pavlou
University of Surrey
Guildford, Surrey, U.K.
{N.Wang, G.Pavlou}@surrey.ac.uk

Abstract- **In this paper we address the optimisation problem of joint intra- and inter-domain multicast routing in multi-homing environments, a topic that has not received much attention until now. Instead of focusing on the traditional Steiner tree based multicast routing optimisation, we consider plain IP based approaches in which routing is controlled through optimised multi-topology-aware IGP/BGP configurations. The benefit is that no dedicated MPLS tunnelling is required for multicast traffic delivery across multiple domains. In this paper we propose a set of heuristic algorithms for the formulated multi-objective optimisation problem considering both intra- and inter-domain operation. Through our simulation experiments, we show that the proposed schemes achieve significant improvement in the relevant performance compared to conventional approaches.**

I. INTRODUCTION

Today, Internet Network Providers (*INPs*) who offer multicast services are facing the challenging task of efficiently delivering multicast traffic both within and across their networks. Traditionally, multicast routing optimisation has been formulated as the well-known Steiner tree problem, with the major objective to deliver multicast traffic with least cost, e.g., by consuming minimum bandwidth resources. Nevertheless, we argue that this problem formulation can be only applied to the *intra-domain* scenario, as computing a global Steiner tree for *inter-domain* multicast traffic is generally neither necessary nor viable in practice. First, computing such a distribution tree requires the knowledge of both the router-level network topology and group membership across multiple domains. Unfortunately, individual *INPs* do not normally release relevant information to their peers who are potential business rivals, due to privacy reasons. Moreover, a global Steiner tree does not always bring benefits to all the involved domains. In Fig. 1(a) a "global optimal" tree in terms of hop counts (shown in thick lines) is shown across three domains, with the source *s* residing in AS1 and three receivers *r* distributed in AS2 and AS3. We notice that, although the multicast tree consumes minimum bandwidth resources (9 network links in total), not all sub-trees within individual domains are optimal in terms of bandwidth conservation. For example, in Fig. 1(b) that shows a "global sub-optimal" tree, the sub-tree in AS2 uses only 3 network links other than 4 in its counterpart constituting the "global optimal" tree. On the other hand, the heterogeneity in the objectives of optimising multicast routing also makes it unnecessary to compute a Steiner tree across multiple domains. If we assume that AS3 aims at minimising end-to-end latency instead of conserving bandwidth resources, then the shortest AS-path routing shown in Figure 1(b) is obviously a better solution. From this example, we can see that there is no incentive for individual network providers to jointly compute a global Steiner tree across multiple domains as this might not meet their own requirements.

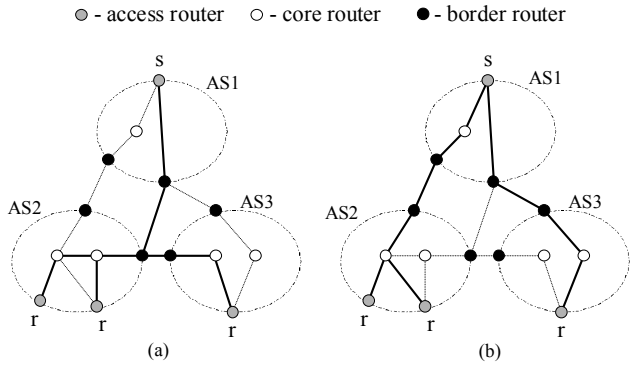

Figure 1 Inter-domain multicast tree construction

In this paper, we focus on multicast routing across multiple domains, not necessarily to compute a global Steiner tree, but to optimise multicast routing for individual *INPs'* traffic engineering (*TE*) objectives. In this paper, we consider two typical objectives: (1) intra-domain bandwidth conservation and (2) load balancing across inter-domain links connecting with adjacent domains. The first objective inherits the traditional property of Steiner trees, and the second one is based on the recent observation that a significant proportion of Internet congestion comes from the hot spots on inter-domain links between *INPs* [3]. This is because many *INPs* tend to overprovision their core networks, while bandwidth resources on inter-domain links are rather scarce.

Multi-homing, which is very popular in today's Internet, offers *INPs* higher robustness and more power for load balancing. By taking one single group as an example, Fig. 2 illustrates how multi-homed domain *R* can perform multicast traffic optimisation through both inter- and intra-domain routing. In the figure, the group source s is located in the remote domain S whose aggregated IP address prefix is identified by P_S. According to

domain R, P_S can be reached through some of its adjacent domains (namely $N_1^S \ldots N_i^S$), specifically via multicast-aware border routers (*M-ASBR*s) b_1 to b_k. In this case, the traffic optimisation for domain R is two fold: (1) To decide which *M-ASBR* to act as the best ingress point for the multicast traffic coming from *s*, and (2) how to configure intra-domain routing to optimally deliver the traffic from that ingress *M-ASBR* to individual receivers. In this paper, we consider small or medium sized *INPs* and also assume that all multicast receivers are static local hosts whose locations are already known. A typical example of this case is stub domains. In our future work we will consider scenarios where multicast tree leaves not only include local group members, but also *M-ASBR*s that lead to remote receivers (e.g., domain N_1^S). In this case, the service agreement between *INPs* may provide information on remote receivers.

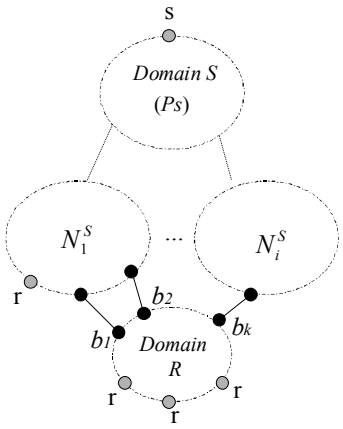

Figure 2 Multicast optimisation in a multi-homing environment

A salient novelty of our proposed scheme is that, solutions based on *direct* multicast path selection heuristics are abandoned, as they need explicit routing functionalities, e.g., using *MPLS*. While *MPLS* is a powerful technology for creating overlay tunnels to support any specific routing strategy, it is also expensive and suffers potentially from scalability problem in terms of *LSP* state maintenance. In this paper, we adopt plain IP routing protocols for enforcing optimal multicast routing without setting up dedicated point-to-multipoint (*P2MP*) *LSPs*. Specifically, multi-topology extensions to *IGPs* (*M-IGP*) [9,10] and multi-protocol *BGP* (*M-BGP*) [2] are manipulated for influencing intra- and inter-domain multicast path selections respectively. These protocol extensions allow us to decouple the multicast path selection from the default unicast routing. Detailed description on this approach will be provided in section III. The advantage of this plain IP based solution is obvious: the high expense and complexity in setting up *P2MP LSPs* across multiple domains can be avoided. Hence, we regard the proposed scheme as an efficient and scalable solution to tackle inter-domain multicast routing optimisation.

II. Background

We start from unicast routing optimisation. In the literature, optimising *IGP* link weights [6,14,19] has been deemed as a scalable and efficient alternative to *MPLS* based approaches for offline intra-domain traffic engineering. This type of plain IP based *TE* paradigm is also applicable to the inter-domain case, where *BGP* routing attributes are manipulated for optimal inter-domain traffic distribution. Some of these *BGP* attributes are used to control outbound traffic (i.e., how unicast flows are delivered out of the local domain towards remote destinations), such as the Local_Preference (local_pref) attribute [3,16], while some others can be applied for influencing inbound traffic, such as the AS_PATH attribute and the Multi_Exit_Discriminator (*MED*) attribute [4,11]. The objectives of these inbound/outbound unicast TE schemes cover a very wide range, from business aspects (e.g. minimise overall monetary cost [16]), to network performance aspects (e.g., bandwidth conservation [3] and load balancing [11,16]). Nevertheless, none of the existing IP based traffic optimisation works have considered multicast flows within the network.

Despite different problem formulations for multicast routing optimisation, most common solutions are to apply direct path selection heuristics to construct the static multicast tree step by step [8,13,15,20]. The most distinct disadvantage of this strategy is that, as the proposed routing algorithms cannot be easily implemented in IP routers, MPLS tunnels are needed to enforce path selections. Inspired from the IP based unicast traffic engineering approach in [6], we proposed an intra-domain multicast TE scheme by optimising *M-IGP* link weights [18]. In this case, optimal intra-domain multicast trees are represented into shortest path trees that can be automatically handled by legacy IP routers.

III. Overview of the Optimisation Framework

Before formulating the joint offline intra- and inter-domain multicast traffic optimisation problem, we first provide a brief review on the current multicast routing semantics, taking Source Specific Multicast (*SSM* [1]) as a typical example. In *SSM*, *PIM-SM* [5] is responsible for constructing a multicast tree across the Internet, from individual group members to the single source for each group. As Designated Routers (*DRs*) for receivers already obtain the IP address of the group source, they are able to send *PIM-SM* join requests directly towards the remote source. Traditionally, path selections of *PIM-SM* join requests follow the underlying unicast routing table that is populated by the conventional *IGP*/*BGP* routing protocols. The advent of multi-topology extensions to these protocols, such as *M-ISIS* [9], *MT-OSPF* [10] and *M-BGP* [2], has made it possible to decouple multicast routing from its unicast counterpart. As a result, dedicated multicast routing optimisation can be achieved through manipulating the routing metrics such as *IGP* link weights and *BGP* route attributes, specifically within the routing plane for multicast traffic.

A. Multi-Topology Routing

The multi-topology extensions to *IS-IS* and *OSPF* provide the original protocols with additional ability of viewing the weight of each link for different logical IP topologies independently. Take *MT-OSPF* as an example, the field of Multi Topology Identifier (*MT-ID*) with value 1 in *MT-OSPF* is dedicated to the multicast routing plane. With this multi-topology capability, network providers are able to perform dedicated *M-IGP* link weight optimisation in the multicast routing plane, without worrying about the unwanted path changes for the unicast traffic due to the adjustment of "shared" link weights.

Same as the conventional *ISIS/OSPF* protocols, the original *BGP* only provides a unique routing plane across the Internet. That means, given any IP prefix (indicated in the field of Network Layer Reachability Information, *NLRI*), only one single route is advertised for all types of flows, including IPv4/6, unicast/multicast flows. In M-*BGP*, extensions to *NLRI* are made for advertising incongruent routes for different types of traffic. These new attributes are known as *MP_REACH_NLRI* and *MP_UNREACH_NLRI*. As specified in [2], the Address Family Information (*AFI*) and Sub Address Family Information (*SAFI*) carried in each *M-BGP* advertisement jointly identify the routing plane for different types of flows. For example, an *M-BGP* update message with *AFI* = 1 and *SAFI* = 2 indicates that this advertisement is only carrying IPv4 multicast routes. In this case, existing inter-domain unicast *TE* technologies by tweaking *BGP* route attributes [16] can be applied to multicast as well, provided that the route attributes are configured in the logical network topology for multicast traffic.

B. An Integrated Infrastructure

Fig. 3 illustrates the integrated IPv4 multicast traffic optimisation based on the *M-IGP* and *M-BGP* protocols. Within each resource optimisation cycle, e.g., on weekly or monthly basis, *M-IGP* link weights and *M-BGP* route attributes (e.g., local_pref) are computed/manipulated offline according to the given optimisation objectives. Thereafter, the resulting *M-IGP* link weights and *M-BGP* route attributes are configured in the IPv4 multicast routing plane respectively, using proper *MT-ID* and *AFI/SAFI* values. The IP routers will then populate the multicast routing table (*M-RIB*) for each source prefix. Once a *PIM-SM* join request is received, each router decides, according to its *M-RIB*, the next hop neighbour to send the packet out of the local domain towards the remote group source. In this scenario, the *PIM-SM* join request follows an engineered outgoing path decided by the *M-BGP* attribute (e.g., local_pref) and *MT-OSPF* link weights. As a result, the multicast traffic not only is injected into the local domain via the desired *M-ASBR* (from where the original *PIM-SM* join request packet is delivered outside the local domain), but it also travels along the optimal intra-domain paths that conform to the optimisation requirement. In addition, the multicast forwarding information base (*M-FIB*) is dynamically updated for the incoming interface (iif) and outgoing interface (oif) list of each group.

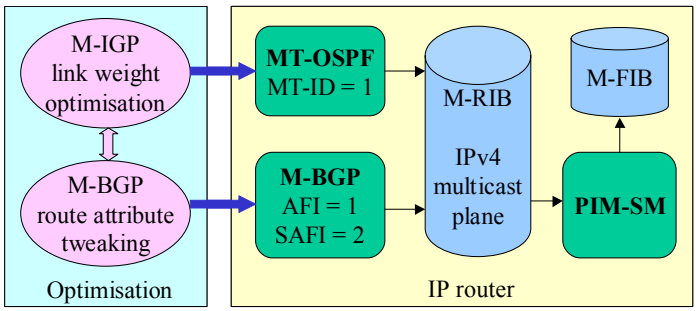

Figure 3 IPv4 multicast traffic optimisation with *M-IGP/M-BGP*

IV. Problem Formulation

As we have mentioned previously, optimisation of intra-domain multicast routing is often formulated into the Steiner tree problem for bandwidth conservation purposes. In a multi-homing environment, bandwidth consumption can be also influenced by the position of the ingress *M-ASBR*. As we consider plain IP based solutions, the optimisation problem is tackled *jointly* through *M-IGP* link weight tuning and *M-ASBR* ingress point selections. Let's take Fig. 4 as an example. We assume that the prefix containing the source *s* for the multicast group can be reached via both *M-ASBR*s *b1* and *b2*. If *b1* is selected as the ingress point, with conventional intra-domain hop-count based shortest path routing, the resulting tree uses 6 network links (Fig. 4(a)). If *b2* is selected, with proper *M-IGP* link weight setting we are able to conserve 50% of intra-domain bandwidth resources, as only 3 network links are used (shown in Fig. 4(b)). From this example we can see that the task of *M-ASBR* selection is to find the optimal root of the multicast tree within the domain, while *M-IGP* link weight tuning is responsible for exploring the best intra-domain paths from the root to individual receivers. Apart from bandwidth conservation, we also consider other objectives such as load balancing across inter-domain links, and this turns the task of integrated multicast routing into a multi-objective optimisation problem.

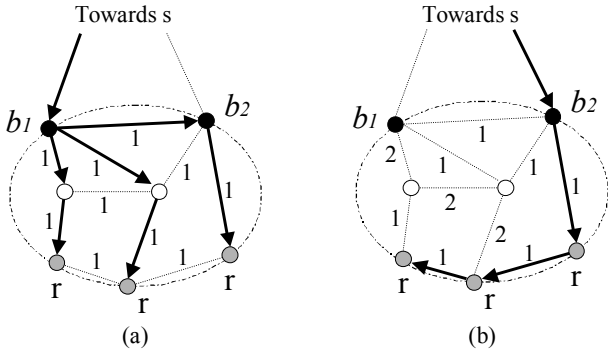

Figure 4 *M-IGP* link weight tuning and *M-ASBR* selection

A network is represented through a directed graph $G=<V,E>$ where V and E denote the node set and the intra-domain link set respectively. The node set V is further categorised into Access router set V_A, Border router (*M-ASBR*) set V_B and Core router set V_C. The nodes in V_A are only attached with static end hosts (receivers), while those in V_B connect other domains (normally provider domains) through inter-domain links. For simplicity, we assume that each *M-ASBR* $b \in V_B$ is attached with only one inter-domain link whose bandwidth capacity is C_b. All the nodes in V are logically connected in full mesh with internal *M-BGP* (*i-M-BGP*) sessions. We consider a set of disjoined unicast address prefixes $P=\{P_1,...,P_k\}$ in the Internet, each of which can be reachable via a distinct subset of V_B. We also assume t multicast group sessions $m_1,...,m_t$. The single source of each multicast group s_i ($0<i<t$) is included in one of the address prefixes P_j ($0<j<k$) under consideration. Each group m_i is associated with a receiver set $R_i \subseteq V_A$ as well as the bandwidth demand D_i, which means that D_i units of bandwidth is consumed on each link of the multicast tree T_i spanning s_i and R_i.

Given the network and multicast group information, the joint optimisation task is to (i) for each prefix P_j ($0<j<k$), to select the best *M-ASBR* $\overline{b_j}$ as the ingress point, and (ii) to assign a proper *M-IGP* weight w_{uv} for each intra-domain link $(u,v) \in E$, based on which shortest path routing is performed for computing *all* multicast trees. As we have mentioned before, the optimisation objective of the task above is to construct a multicast tree T_i for each group m_i with the purpose of (i) intra-domain bandwidth conservation, i.e.

$$\text{minimise } l^{\text{intra}} = \sum_{i=1}^{t} \sum_{(u,v)\in E} D_i \times x_{uv}^i \tag{1}$$

$$\text{where } x_{uv}^i = \begin{cases} 1 & if\ (u,v) \in T_i \\ 0 & otherwise \end{cases} \tag{2}$$

and (ii) inter-domain load balancing, i.e.,

$$\text{minimise } \max(u_b^{\text{inter}} = \frac{\sum_{i=1}^{t} D_i \times y_b^i}{C_b}) \text{ for each } b \in V_B \tag{3}$$

where

$$y_b^i = \begin{cases} 1 & if\ b \text{ is selected as the ingress of } m_i \text{ (i.e. root of } T_i) \\ 0 & otherwise \end{cases} \tag{4}$$

It is worth emphasising that, if the sources of multiple groups belong to the same prefix (a special case is that multiple groups share the same source), all these groups should select the same *M-ASBR* as their common ingress point. In other words, the binary variable of y_b^i in (3) and (4) should have the same value for all groups whose sources belong to the same prefix. The reason for this is because *M-BGP* route attributes are only associated with aggregated unicast address prefixes without considering multicast group information. Hence, *M-ASBR* ingress selection is based on prefixes other than group specific.

V. Proposed Algorithms

The optimisation problem formulated above is *NP-hard*, as it is a combination of three other *NP-hard* problems: namely Steiner tree [8,15], shortest path representability [19] and the Generalised Assignment Problem (*GAP*) [3]. In this section we propose a set of efficient schemes for solving the problem, which can be further classified into Single Ingress Selection (*SIS*) and Multiple Ingress Selection (*MIS*) algorithms.

A. Single Ingress Selection (SIS)

We first introduce the most straightforward algorithm named Single ingress selection with Hop-Count intra-domain routing (*Single-HC*). In this approach, intra-domain routing is based on the metric of hop-counts, which means that the *M-IGP* weight is set to 1 for all intra-domain links. On the other hand, *M-ASBR* ingress selection is based on a greedy search algorithm solely for the purpose of conserving bandwidth resources (objective (i)). That is, to select a single *M-ASBR* for each prefix that results in the least intra-domain bandwidth consumption l^{intra} with the constraint of bandwidth capacity of inter-domain links. Specifically, we sort individual prefixes in descending order according to the sum of the bandwidth demand from the groups whose source is in that prefix. After that, we assign sequentially these prefixes to a specific *M-ASBR* (with sufficient residual bandwidth on the inter-domain link) with least intra-domain bandwidth consumption.

Next we consider joint intra- and inter-domain routing by taking into account *M-IGP* link weight optimisation. In order to obtain the optimal value of each link weight as well as the best ingress candidate for each prefix, we introduce a Genetic Algorithm (*GA*) based approach – the *Single-GA* algorithm. Genetic Algorithms can be described as follows. First, a series of random solutions are obtained as the initial generation of chromosomes in the population. Thereafter, improved offsprings evolve iteratively from the parents by calculating their fitness. Chromosomes with higher fitness have higher probabilities of being inherited by the next generation. In each iteration, a new generation of chromosomes is created through the process of parent selection and reproduction. This is specifically achieved through genetic operators such as crossover and mutation on genes constituting each chromosome. After a predefined number of generations, or when fitness performance has converged, the chromosome with the best fitness is selected as the final solution.

Procedure *Single-GA-fitness*
Begin
Set the *M-IGP* weight of each intra-domain link in the network according to the chromosome;
For each prefix P_j
Aggregate group bandwidth demand according to P_j, i.e.
$AD_j^{\text{inter}} = \sum_{i=1}^{t} D_i$ for $s_i \in P_j$;
End for;
Sort the prefix list *P* in descending order according to AD_j^{inter} ($0<j<k$);
For each prefix P_j in the ordered list *P*
Assign an *M-ASBR* $\bar{b} \in V_B$ reachable to P_j such that
(1) Intra-domain bandwidth consumption l_j^{intra} is minimised for the groups whose source $s_i \in P_j$ and
(2) *M-ASBR* $\bar{b}$ has sufficient residual bandwidth for the aggregated demand AD_j^{inter};
Update inter-domain link utilisation on $\bar{b}$, i.e.,
$u_{\bar{b}}^{\text{inter}} = u_{\bar{b}}^{\text{inter}} + AD_{\bar{b}}^{\text{inter}} / C_{\bar{b}}$;
End for;
$l^{\text{intra}} = \sum_{j=1}^{k} l_j^{\text{intra}}$; /* Sum up total intra-domain bandwidth consumption for all prefixes */
$fitness = \frac{\alpha}{l^{\text{intra}} + \alpha \times \max(u^{\text{inter}})}$;
End

Figure 5 Algorithm for computing fitness (*Single-GA*)

Our strategy is to let *GA* search for optimal *M-IGP* link weights during the evolution process of the population. In this case, the genes in each chromosome are the *M-IGP* weights for individual intra-domain links. In addition, we design a simple heuristic for *M-ASBR* ingress selection for each chromosome, i.e., a dedicated set of link weights. This algorithm is similar to *Single-HC*, except that intra-domain routing is based on the underlying *M-IGP* weights obtained from each chromosome. The most important issue in *GA* based solutions is how to design the fitness so as to drive the whole population towards the optimal result. In the proposed *Single-GA* approach, the fitness reflects both objectives of intra-domain bandwidth conservation and inter-domain load balancing. To achieve this, we define the fitness of each chromosome as follows:

$$fitness = f(l^{\text{intra}}, \max(u^{\text{inter}})) = \frac{\alpha}{l^{\text{intra}} + \alpha \times \max(u^{\text{inter}})}$$

where α is a constant and α is a tuneable parameter for controlling the trade-off between the two objectives, namely minimising l^{intra} and minimising $\max(u^{\text{inter}})$. If we use the notation in the problem formulation, the fitness is:

$$fitness = \frac{\omega}{\sum_{i=1}^{t} \sum_{(u,v) \in E} D_i \times x_{uv}^i + \alpha \times \max_{b \in V_B} ((\sum_{i=1}^{t} D_i \times y_b^i) / C_b)}$$

With this definition, chromosomes with higher fitness, i.e., lower overall intra-domain bandwidth consumption and lower maximum inter-domain link utilisation, have higher possibilities to survive in the next generation. Fig. 5 provides description of computing fitness in the *Single-GA* approach, including the embedded heuristic of *M-ASBR* ingress router selection. The time complexity of the *Single-GA* algorithm is $O(|E|+t+k\log k+k|V_B|)$: Set the weight of each link takes $O(|E|)$. The task of aggregating group bandwidth can be done through visiting individual groups with the complexity of $O(t)$, and sorting of the source prefix list takes $O(k\log k)$. Finally, the ingress point selection has complexity of $O(k|V_B|)$.

B. Multiple Ingress Selection (MIS)

In the unicast scenario, Hot Potato Routing (*HPR*) is very popular for delivering inter-domain traffic towards a specific remote prefix via multiple egress points. In *HPR*, if multiple routes are available for a given prefix whose route attributes with higher priorities (e.g., local_pref, AS_PATH) are "equally good", each router will then select its own closest *ASBR* (decided by the *IGP* link weight) to deliver the unicast traffic out of the local domain. The benefit of *HPR* for unicast traffic is two folded. First, it helps to minimise intra-domain bandwidth consumption, and second, the delay of traffic delivery can be also reduced as each flow is sent out from the local domain as quickly as possible.

In contrast to its popularity in unicast routing, *HPR* is seldom considered in multicast traffic delivery. Apart from the lack of multicast deployment in practice, there exist two major theoretical reasons for this. First, *HPR* is always able to achieve minimum intra-domain bandwidth consumption for unicast routing (hop-count based) if the capacity constraint on inter-domain links is ignored [3]. However, this is not the case for multicast routing, where it is possible to consume still lower bandwidth resources by making multiple receivers share a common intra-domain path than allowing them to find their own closest ingress points (see Fig. 6). The second issue is bandwidth consumption on inter-domain links. In unicast routing, the *total* bandwidth resources consumed on inter-domain links is not influenced by egress router selections including *HPR*. Again, multicast routing does not obey this rule: For each particular group, the overall bandwidth consumption on inter-domain links is proportional to the number of ingress points selected in *HPR*. This means that, using *HPR* for intra-domain bandwidth conservation might result in undesired high utilisation on inter-domain links.

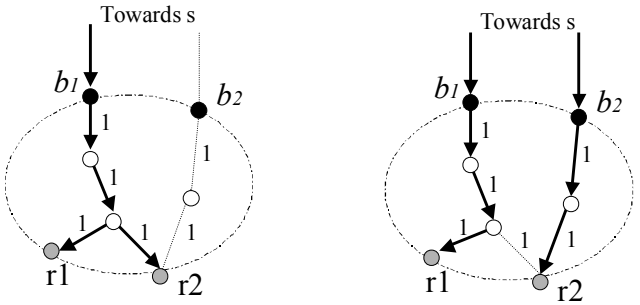

Figure 6 Inefficiency in using HPR in multicast routing

In this section, we explore the feasibility of applying *controlled HPR* for multicast routing optimisation. Our *MIS* strategy is that, additional ingress points are selected only if they are able to achieve *significant* intra-domain bandwidth conservation compared to *SIS*. One approach to enabling *MIS* in practice is as follows. For all *M-ASBRs* that can reach the source prefix, first to set higher (but equal) values of local_pref for the *M-ASBR*s selected by the optimisation procedure, and (2) to configure equally good attributes (before reaching the attribute of *M-IGP* link weight) for these selected *M-ASBRs*, such that each *i-M-BGP* speaker can decide its nearest ingress point according to *M-IGP* link weights only from these selected *M-ASBRs*. As a result, *HPR* can be only performed among the *M-ASBRs* with higher local_pref. Again, we emphasise that, *HPR* can be only adopted on per prefix basis, and it is not possible for individual groups to select their own ingress points. We extend *Single-GA* into a new algorithm named Controlled *GA*-based *HPR* (*C-HPR-GA*). First of all, we assume that the selection of the primary ingress router $\bar{b}$ for prefix P_j within each chromosome in the *Single-GA* algorithm results in overall intra-domain bandwidth consumption of $l_j^{\text{intra}}(\{\bar{b}\})$. After that, we consider the scenario of adding another potential ingress $b' \in V_B \setminus \{\bar{b}\}$. We try all *M-ASBRs* that can reach P_j, except $\bar{b}$ itself, and find the one, namely *b'*, that (jointly with $\bar{b}$) results in least intra-domain bandwidth consumption and also has sufficient bandwidth on its inter-domain link. If the new overall intra-domain bandwidth consumption $l_j^{\text{intra}}(\{\bar{b}\} \cup \{b'\})$ is below $\lambda \times l_j^{\text{intra}}(\{\bar{b}\})$, where $0<\lambda<1$ (λ is set to 0.5 in our algorithm), then *b'* will be selected as the secondary ingress for P_j. We repeat this procedure until a pre-defined maximum number of ingresses for each prefix, namely B_m, is reached. Fig. 7 shows the detailed algorithm for *MIS* during the computing of fitness for each chromosome. The time complexity of the *C-HPR-GA* algorithm is $O(|E|+t+klogk+k|V_B|^2)$, assuming that the maximum value of B_m is $|V_B|$.

Procedure *C-HPR-GA-fitness*
Begin
Set the *M-IGP* weight of each intra-domain link in the network according to the chromosome;
For each prefix P_j
Aggregate group bandwidth demand according to P_j, i.e.,
$AD_j^{\text{inter}} = \sum_{i=1}^{t} D_i$ for $s_i \in P_j$;
End for;
Sort the prefix list *P* in descending order according to AD_j^{inter} ($0<j<k$);
For each prefix P_j in the ordered list *P*
Assign an *M-ASBR* $\bar{b} \in V_B$ reachable to P_j such that
(1) Intra-domain bandwidth consumption $l_j^{\text{intra}}(\{\bar{b}\})$ is minimised for the groups whose source $s_i \in P_j$ and
(2) *M-ASBR* $\bar{b}$ has sufficient residual bandwidth for the aggregated demand AD_j^{inter};
Update inter-domain link utilisation on $\bar{b}$, i.e.,
$u_{\bar{b}}^{\text{inter}} = u_{\bar{b}}^{\text{inter}} + AD_{\bar{b}}^{\text{inter}} / C_{\bar{b}}$;
$B_j = \{\bar{b}\}$;
$|B_j| = 1$; /* Find additional ingresses for P_j */
While $|B_j| < B_m$
Find $b' \in V_B \setminus B_j$ reachable to P_j such that
(1) Intra-domain bandwidth consumption $l_j^{\text{intra}}(B_j \cup \{b'\})$ is minimised and
(2) *M-ASBR* b' has sufficient residual bandwidth for the aggregated demand AD_j^{inter};
if $l_j^{\text{intra}}(B_j + \{b'\}) < \lambda \times l_j^{\text{intra}}(B_j)$
$B_j = B_j \cup \{b'\}$; $|B_j| = |B_j| + 1$;
Update inter-domain link utilisation on b', i.e.,
$u_{b'}^{\text{inter}} = u_{b'}^{\text{inter}} + AD_{b'}^{\text{inter}} / C_{b'}$;
end if;
End while;
End for;
$l^{\text{intra}} = \sum_{j=1}^{k} l_j^{\text{intra}}(B_j)$; /* Sum up total intra-domain bandwidth consumption for all prefixes*/
$$fitness = \frac{\alpha}{l^{\text{intra}} + \alpha \times \max(u^{\text{inter}})};$$
End
Figure 7 Algorithm for computing fitness (C-HPR-*GA*)

VI. PERFORMANCE ANALYSIS

In our simulation experiments, we used the GEANT [7] network topology that contains 23 nodes and 76 unidirectional links (two links with opposite directions between any pair of adjacent nodes). The scaled bandwidth capacity of each link is set to 10^4 units. We consider 100 multicast groups whose sources are distributed randomly in 50 remote prefixes. We repeat this random distribution for ten times for each instance of test configuration, and those prefixes that do not contain any source for these 100 groups are not considered. We assume that each remote prefix can be reached via maximum 50% of the *M-ASBRs* of the network. In order to obtain the best GA performance, we test different values for manipulating the chromosomes in each generation, and we set the probability of crossover and mutation to 0.3 and 0.001 respectively. The number of chromosomes included in each generation is set to 100. The maximum generation for each instance of GA calculation is 500. Apart from the GEANT network topology, we also used random topologies created by *GT-ITM* for testing our proposed algorithms, and we found that the performances are very similar among these topologies.

Apart from the *GA* based schemes, we also implemented three other approaches for comparison purposes, namely, Single ingress selection with hop count based intra-domain routing (*Single-HC*), uncontrolled *HPR* with hop count based intra-domain routing (*U-HPR-HC*), and controlled *HPR* with hop count based intra-domain routing (*C-HPR_HC*). In order to evaluate intra-domain bandwidth conservation capabilities, we use the performance of *Single-HC* as the baseline, and define Bandwidth Conservation Ratio (*BC-ratio*) for the rest algorithms. Specifically, the *BC-ratio* of a specific algorithm is the ratio of its intra-domain bandwidth consumption over that of the *Single-HC* algorithm. The maximum bandwidth demand for individual multicast groups Max D_i is used as the x-axis in the following figures. The range of Max D_i for each test is from 200 to 1200 units.

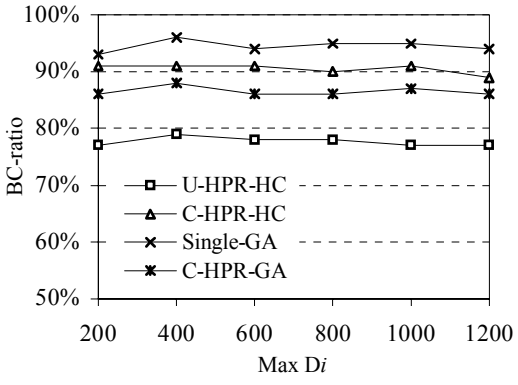

Figure 8 *BC-ratio* vs. Max D_i (α =10^3)

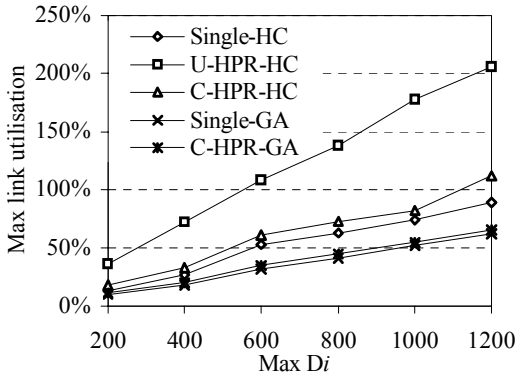

Figure 9 Maximum inter-domain link utilisation vs. Max D_i (α =10^3)

Fig. 8 and Fig. 9 illustrate roughly "equal-split efforts" on the two objectives by setting the value of α to 10^3. From Fig. 8 we can see that all the proposed algorithms outperform *Single-HC* in terms of intra-domain bandwidth conservation. Specifically, uncontrolled *HPR* is able to achieve the *BC-ratio* of 78%, which means up to 28.2% of the bandwidth resources within the network can be saved. By limiting the total number of ingress routers, the *C-HPR-GA* algorithm has higher *BC-ratio* (87%), meaning that it is able to conserve 14.9% of intra-domain bandwidth compared to *Single-HC*. On the other hand, by using controlled *HPR* with hop count intra-domain routing, and optimising *M-IGP* link weights with single ingress selection, *C-HPR-HC* and *Single-GA* have the *BC-ratio* of 91% and 94% respectively. Fig. 9 shows the maximum link utilisation across inter-domain links. As we expected, uncontrolled *HPR* has much higher utilisation on inter-domain links than all the other algorithms, which indicates that normally it is not a proper solution in practice. Another notable result is that, both *Single-GA* and *C-HPR-GA* achieve the best performance in load balancing across inter-domain links. Moreover, *C-HPR-GA* does not exhibit much higher utilisation on inter-domain links. This is because we have tightly controlled maximum inter-domain link utilisation in the fitness calculation function.

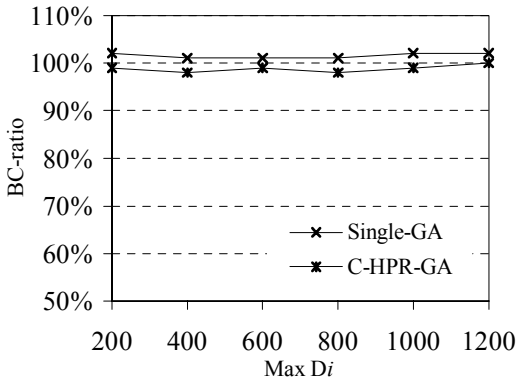

Figure 10 BC-ratio vs. Max D_i (α =10^4)

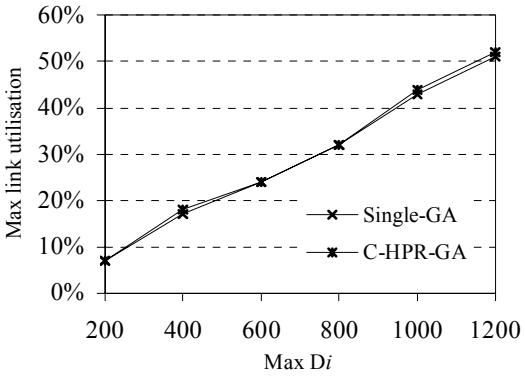

Figure 11 Maximum inter-domain link utilisation vs. Max D_i (α =10^4)

Fig. 10 and Fig. 11 show the performance of the *GA*-based algorithms when more efforts are put towards load balancing across inter-domain links. We achieve this by increasing the value of α from 10^3 to 10^4. From now on we don't include other algorithms because they are not capable of tuning efforts between the two objectives through α. From Fig. 10 we notice that neither *C-HPR-GA* nor *Single-GA* have any gain on intra-domain bandwidth consumption. However, Fig. 11 shows that the two algorithms are able to further decrease maximum inter-domain link utilisation (by 7% compared to Fig. 9). Similar to the previous scenario, the gap between the two GA-based schemes in inter-domain load balancing is very small in Fig. 10.

VII. SUMMARY

In this paper, we investigated offline optimisation on joint intra- and inter-domain multicast routing in multi-homing environments. First of all, we demonstrated why traditional Steiner tree based problem formulations do not generally apply to the inter-domain case. Following that, we proposed a generic optimisation framework using plain IP based routing paradigms. Specifically, link weights of *M-IGP* and route attributes of *M-BGP* are jointly manipulated for optimised intra-domain bandwidth consumption and load balancing across inter-domain links. Simulation results have shown that the proposed solution is able to achieve significantly better performance than conventional routing configurations. Especially, if *HPR* is properly controlled, intra-domain bandwidth can be further conserved without any expense of increasing inter-domain link utilisation. This IP based approach achieves high simplicity and scalability in the control plane, as there is no need for setting up dedicated *P2MP MPLS* tunnels across multiple domains.

As this work is the very first step towards inter-domain multicast traffic optimisation, there still exist many issues that need to be considered. First, *BGP* has been often blamed for causing potentially Internet routing instability and slow convergence problems. When *M-BGP*, which is an incremental extension to *BGP*, is used for inter-domain multicast routing, it is not surprising that these problems do occur in the multicast routing plane. Hence, our next step is to investigate stable inter-domain multicast routing optimisation, taking into account some existing research works on both unicast and multicast routing stability, e.g., [12,17]. Finally, routing robustness in case of both intra- and inter-domain link failures is yet another topic to be addressed in our future research.

REFERENCES

[1] S. Bhattacharyya, "An Overview of Source Specific Multicast (SSM)", RFC 3569, Jul. 2003

[2] T. Bates et al, "Multiprotocol Extensions for BGP-4", RFC 2858

[3] T. C. Bressoud et al, "Optimal Configuration for BGP Route Selection", IEEE INFOCOM, 2003.

[4] R.K.C. Chang and M. Lo, "Inbound Traffic Engineering for Multihomed ASs Using AS Path Prepending", IEEE Network Magazine, March/April 2005

[5] B. Fenner et al, "Protocol Independent Multicast - Sparse Mode (PIM-SM): Protocol Specification (Revised)", Internet draft, draft-ietf-pim-sm-v2-new-11.txt, Oct. 2004, work in progress

[6] B. Fortz et al, "Internet Traffic Engineering by Optimising OSPF Weights", IEEE INFOCOM, 2000, pp. 519-528

[7] The GEANT network topology, available online at: http://www.geant.net/upload/pdf/Topology_Oct_2004.pdf

[8] L. Kou et al, "A Fast Algorithm for Steiner Trees", Journal of Acta Informatica, 15, pp. 141-145, 1981

[9] T. Przygienda et al, "M-ISIS: Multi Topology (MT) Routing in IS-IS", Internet Draft, draft-ietf-isis-wg-multi-topology-09.txt, March. 2005, work in progress

[10] P. Psenak et al, "Multi-Topology (MT) Routing in OSPF" Internet Draft, draft-ietf-ospf-mt-04.txt Apr. 2005

[11] B. Quoitin et al, "A performance evaluation of BGP-based traffic engineering", International Journal of Network Management, 15(3), May-June 2005.

[12] P. Rajvaidya, K. Almeroth, "Multicast Routing Instabilities", IEEE Internet Computing, Vol. 8, Issue 5, 2004, pp. 42-49

[13] G. N. Rouskas et al, "Multicast Routing with End-to-end Delay and Delay Variation Constraints", IEEE JSAC Vol. 15, No. 3, pp. 346-356, Apr. 1997

[14] A. Sridharan et al, "Achieving Near-Optimal Traffic Engineering Solutions for Current OSPF/IS-IS Networks", IEEE INFOCOM, pp. 1167-1177, Apr. 2003

[15] H. Takahashi, A. Matsuyama, "An Approximate Solution for the Steiner Problem in Graphs", Math. Japonica 6, pp533-577

[16] S. Uhlig and O. Bonaventure, "Designing BGP-based outbound traffic engineering techniques for stub ASes", ACM SIGCOMM Computer Communications Review, October 2004.

[17] H. Wang et al, "Stable Route Selection for Interdomain Traffic Engineering", IEEE Network Magazine, December, 2005

[18] N. Wang, G. Pavlou, "Bandwidth Constrained IP Multicast Traffic Engineering without MPLS Overlay", IEEE/IFIP MMNS, 2004

[19] Y. Wang et al, "Internet Traffic Engineering without Full Mesh Overlaying", Proc. IEEE INFOCOM, Vol. 1, pp. 565-571, 2001

[20] Q. Zhu et al, "A Source Based Algorithm for Delay-constrained Minimum-cost Multicasting", Proc. on IEEE INFOCOM, Vol. 1, pp. 377-385, 1995