# Inter-domain routing

Bakr Sarakbi

Thèse de doctorat de Télécom SudParis dans le cadre de l'école doctorale S&I en co-accréditation avec l'Université d'Evry-Val d'Essonne

Spécialité
Réseaux - Informatique

Par
**Bakr SARAKBI**

Thèse présentée pour l'obtention du diplôme de Docteur
de Télécom SudParis

Titre

# Routage Inter-Domaine

Soutenue le 10 Février 2011 devant le jury composé de:

| | | |
|---|---|---|
| Timothy G. Griffin | Rapporteur | Université de Cambridge |
| Patrick Sénac | Rapporteur | LAAS-CNRS |
| Khaldoun Al Agha | Examinateur | Université Paris XI |
| Marcelo dias de Amorim | Examinateur | Université Paris VI |
| Ana Cavalli | Directrice de thèse | Telecom SudParis |
| Stéphane Maag | Encadrant | Telecom SudParis |

**Thèse n° 2011TELE0006**

# Acknowledgements

Thank God almighty for providing me with faith, patience, and power to do my researches and to reach this scientific degree.

I would like to thank my dear and lovely parents for their endless love and for taking care of me since the first moment of my life. Without their satisfaction I may not achieve any progress in my life.

I have to thank my wife, the partner who encourages me and stands with me at all times. I would like to thank her for her patience through my busy times. Thanks to her for reviewing my articles and for being the first who read this manuscript.

Thanks to Professor Ana Cavalli for integrating me in her team and for giving me the chance to do my thesis in my favorite subject. And I thank also Stéphane Maag for all what he has done to accomplish this thesis.

I would like to express my gratitude to Professor Timothy Griffin for the time he spent with me, and for the useful discussions we have had together. I am really grateful to him and to Professor Patrick Sénac who honored me by reviewing my dissertation. Undoubtedly, I thank Khaldoun Al Agha and Marcelo dias de Amorim for accepting to be part of my thesis committee.

A special thank to all my colleagues in LOR department, more precisely, Amel, Anis, Iksoon, Mazen and my officemates: Fayssal, Willy, and Felipe, and those who have already graduated: Dario, Bashar, Muneer and Wissam for the useful and nice times we have spent together. A deep thanks also to the administration assistants: Brigitte and Jocelyne for their endless gentleness and support.

Last but not least, I would like to thank all my colleagues at the Higher Institute of Applied Sciences and Technology in Damascus for all types of support they provide during my scientific career.


## Abstract

Internet is the hugest network the humanity has ever known. It provides a large number of various services to more than two billion users. This complex and growing topology lacks stability, which we can notice when a voice call is dropped, when a web page needs to be refreshed, etc. The initiator of this instability is the frequent events around the Internet. This motivates us to unleash a profound study to tackle Internet stability and suggest solutions to overcome this concern.

Internet is divided into two obvious levels: AS (Autonomous System) level and router level. This distinction is reflected on the routing protocols that control the Internet traffic through two protocol types: exterior (inter-AS) and interior (intra-AS). The unique used exterior routing protocol is the external mode of BGP (External Border Gateway Protocol), while there are several used internal routing protocols.

Therefore, stabilizing the Internet is correlated to the routing protocol stability, which directs such efforts to the investigation of routing protocol (BGP) behavior. Studying BGP behaviors results in that its external mode

(eBGP) suffers from long convergence time which is behind the slow response to topology events and, in turn, the traffic loss. Those studies state also that BGP internal mode (iBGP) suffers from several types of routing anomalies that causes its divergence.

Therefore, we propose enhancements for both BGP modes: eBGP and iBGP and try to meet the following objectives: Scalability, safety, robustness, correctness, and backward compatibility with current version of BGP. Our eBGP proposal eliminates the transient disconnectivity caused by slow convergence by suggesting a backup strategy to be used upon the occurrence of a failure. IBGP proposal (skeleton) gives an alternative to the existing internal configurations, that we prove its freeness of anomalies.

Validation methods are essential to prove that the suggested enhancements satisfies the attended objectives. Since we are tackling an interdomain subject, then it is not possible to do validation in the real Internet. We suggested several validation methods to show that our enhancements meet the above objectives. We used simulation environment to implement eBGP backup solution and observe the convergence time and the continuous connectivity. We relied on two tools: brite and rocketfuel to provide us with inter and intra AS topologies respectively. And to ensure the safety of our approaches we employed an algebraic framework and made use of its results.

# Résumé

Internet est le réseau le plus important que l'humanité n'ait jamais connu. Il fournit un nombre large et varié de services à plus de deux milliards d'utilisateurs. Cette topologie grandissanté et complexe manque de stabilité, ce qui peut être noté quand un appel voix est interrompu, quand une page web a besoin d'être rafraichie, etc. Les initiateurs de cette instabilité sont les événements fréquents autour de l'Internet. Cela motive notre étude de l'analyse de la stabilité d'Internet et suggère des solutions à ce défaut.

Internet est divisé en duex niveaux évidents: le niveau AS (Systèmes Autonomes) et le niveau routeur. Cette distinction est reflètée dans les protocoles de routage qui contrle le traffic Internet à travers deux types de protocoles: exterieur (inter-AS) et intérieur (intra-AS). L'unique protocole extérieur utilisé est le mode externe de BGP (External Border Gateway Protocol), alors qu'il existe plusieurs protocoles de routage internes.

De fait, stabiliser l'Internet est corrélé à la stabilité du protocole de routage, ce qui mène de tels efforts à l'investigation autour du comportement du protocole de routage (BGP). L'étude des comportements de BGP en son mode externe (eBGP) souffre de temps de convergence important ce qui est sous la réponse lente des événements topologiques et ainsi la perte du traffic. Ces études établissent aussi que le mode interne de BGP (iBGP) souffre de plusieurs types d'anomalies de routage causant sa divergence.

De fait, nous proposons des améliorations aux deux modes de BGP: eBGP et iBGP et essayons d'atteindre les objectifs suivants: passage à l'échelle, sreté, robustesse et compatibilité avec la version courante de BGP. Notre proposition de eBGP élimine la disconnectivité transitoire causée par une

convergence lente en suggérant une stratégie de backup lors de l'occurence d'une panne. Notre proposition IBGP (skeleton) donne une alternative aux configurations internes existantes, nous prouvons l'absence d'anomalies par son utilisation.

Les méthodes de validation sont essentielles pour prouver que les améliorations suggérées satisfont les objectifs attendus. Puisque nous étudions un sujet interdomain, alors il n'est pas possible d'effectuer la validation dans l'Internet réel. Nous avons suggéré plusieurs méthodes de validation pour montrer que nos améliorations mènent aux objectifs susmentionnés. Nous avons utilisé l'environnement de simulation pour implémenter une solution de backup eBGP et observer le temps de convergence et de la connectivité permanente. Nous nous sommes appuyés sur deux outils: brite et Rocketfuel pour nous fournir des topologies inter et intra AS respectivement. Et pour prouver la sreté de nos approches nous avons utilisé les méthodes algébriques et fait usage de ses résultats.

Ce travail a abordé principalement la stabilité d'Internet. Cette problématique est divisé en deux parties selon la hiérarchie de l'Internet: niveau AS et niveau routeur. La topologie d'Internet, du point de vue du niveau AS, est un ensemble de noeuds, représentant chacun un seul AS. La stabilité de cette topologie est liée à la stabilité du protocole de routage qui est responsable des échanges d'informations de routage entre les noeuds de l'Internet. Du point de vue niveau routeur, chaque AS est une topologie complexe en elle-même, il peut contenir des milliers de routeurs. La stabilité à ce niveau est liée au protocole de routage à l'intérieur de chaque AS. L'instabilité interne influe directement sur la stabilité de l'Internet mondial. Ce qui nous motive à diviser notre travail en deux parties: routage inter-domaine et intra-domaine.

Des propositions efficaces devraient être appliquées à des problématiques bien spécifiées, c'est à dire d'avoir la possibilité de donner une preuve de

correction. Cela nous incite à employer un modèle formel du processus de routage basés sur les algèbres abstraites, puis utiliser ce modèle pour valider les caractéristiques de correction. Les sous-sections suivantes résument les contributions de cette thèse.

# 1. External BGP

La convergence lente est un problème bien connu de BGP. D'une part, rendre ce processus plus rapide peut entraner des comportements indésirables. D'autre part, il n'y a aucune raison pour perdre du trafic depuis que de nombreuses études ont montré que dans la plupart des cas, l'infrastructure sous-jacente est connectée pendant le processus de convergence. Ce qui nous motive autour de ce problème de convergence lente est de proposer une solution à la résolution de perte de trafic.

Outre nos objectifs principaux, à savoir la connectivité continue, nous nous sommes intéressés aux caractéristiques suivantes:

- Absence de boucle pendant et après le processus de convergence.
- limitation de la propagation des pannes.
- Améliorer le temps de convergence.

Afin d'atteindre les objectifs susmentionnés, nous avons proposés quatre mécanismes:

## 1.1. Propagation des informations RC

L'idée de base de ce mécanisme est de créer plus d'informations pour empêcher les autres noeuds de sélectionner des chemins de sauvegarde qui passent par noeud défectueux. Cela pourrait être considéré comme un mécanisme

d'évitement de boucles. Dans certains cas, tous les chemins de sauvegarde (backup) peuvent contenir ces informations. Par conséquent, nous proposons d'ajouter ces informations au niveau du routeur, car le nàeud défaillant pourrait être un ASBR unique tandis que les voisins de l'ASBR du même AS est toujours en cours d'exécution. Donc les informations RC doivent être incluses que dans les messages d'annulation qui sont exportés vers d'autres AS après s'être assuré qu'aucun autre ASBR peut atteindre la destination perdu.

## 1.2. Solution de Backup

Les mécanismes sont divisés en deux phases:
- Calcul du chemin de Backup : il est effectué dans un état stable, chaque noeud construit une liste de chemins de backup et les ordonne selon leur disjonction du chemin principal de chaque préfixe.
- Sélection du chemin de Backup: une fois que le noeud reoit le message d'annulation, il filtre les chemins de backup selon les information RC reus et démarre alors en utilisant le meilleur chemin de backup.

## 1.3. Retrait sélectif

Ce mécanisme consiste à limiter la propagation de la panne. Il stipule que si un noeud reoit un message de retrait (d'annulation) et n'a pas de backup, alors il la diffuse à leurs voisins. Si c'est le cas, alors il le transmet uniquement au prochain noeud du backup si ce dernier était son prédécesseur dans le vieux chemin primaire.

### 1.4. Temporisateur de routes transitoires

Puisque le chemin de backup n'est pas nécessairement la meilleure seconde route, nous avons utilisé un timer supplémentaire (TRT) pour trouver le meilleur nouveau chemin et l'installer dans le Loc-RIB au lieu du backup. Cela garantit que le Loc-RIB contient les meilleurs chemins, qui est normalement l'état stable.

Les résultats des simulations ont montré que la non-connectivité est (relativement) éliminée lors du processus de convergence. Un autre résultat interessant de la simulation a été la limite des pannes qui est le nombre d'AS qui sont touchés par les événements de la topologie. L'algèbre PBR nous a donné un outil de validation pour prouver la sreté de notre solution de backup proposée. En outre, cela nous aide à travers la politique des liens, d'appliquer des conditions de sreté sur le chemin de backup de telle manière que cela élimine les boucles dans la topologie, et ce pendant et après le processus de convergence.

## 2. Internal BGP

La topologie des AS internes est une partie inhérente de l'Internet global, et les événements internes sont un des facteurs qui influent sur la stabilité de l'Internet. Cela rend la stabilité du iBGP essentiel pour une stabilité globale de BGP.

Dans le routage intra-domaine, et selon la configuration interne, le trafic du contrle et des données peut ne pas avoir des topologies semblables. Nous avons compris que cette distinction est derrière la plupart des phénomènes de divergence de routage intra-domaine. Cela nous motive à proposer une alternative aux configurations intra-AS actuelles.

Nous proposons l'approche *Skeleton* comme une alternative aux configurations actuelles comme: full mesh, la réflexion de route, ou confédération. Skeleton est à la fois scalable et sans anomalies. Il assure la simplicité, la correction, et la compatibilité avec les systèmes d'exploitation existants. Skeleton est une solution facile à employer, il peut être déployé en utilisant les dispositifs existants qui supportent la réflexion de routes.

Le modèle topologique proposé donne pour chaque ASBR un DAG attaché à l'ASBR lui-même. Skeleton établit des sessions iBGP sur chaque branche dans l'arbre. Cela garantit que chaque routeur interne reoit les informations de routage grce à son meilleur chemin pour quitter l'AS, et fait en sorte que le trafic de contrle et de données suivent les mêmes chemins. Pour augmenter la redondance de Skeleton, nous avons utilisé une stratégie semblable à celle du backup proposée pour eBGP. Il stipule qu'un routeur interne établit des sessions iBGP supplémentaires (passive) avec son second meilleur prochain saut lié à une autre branche. Nous avons prouvé que cette session supplémentaire garantit la robustesse contre les défaillances.

La caractéristique de correction est obtenue en prouvant que notre approche répond à des conditions suffisantes. Le passage à l'échelle est garanti car le nombre de sessions iBGP dans Skeleton est limitée par le nombre de liens AS physiques internes. La compatibilité ascendante est garantie en utilisant le mappage entre les modèles des réflecteur/client et successeur/prédécésseur. Skeleton surmonte la problème d'asymmetrie en établissant des sessions iBGP entre les voisins IGP seulement. En outre, les oscillations du MED sont traitées en fournissant à tous les noeuds toutes les mises à jour de routage de tous les noeuds de sortie, ce qui garantit une décision stable à prendre par chaque noeud. Enfin nous montrons que la configuration Skeleton est sr en utilisant l'algèbre de routage.

# 3. Méthodes de validation

La validation est une partie intégrante de cette thèse. Le premier défi a été de choisir le meilleur outil de validation qui donne des preuves expérimentales ou théoriques des propriétés alléguées de notre proposition. Nous avons proposé plusieurs approches et tenté de leur faire satisfaire les propriétés suivantes:

- Sreté: absence de boucles.
- Robustesse aux défaillances.
- Passage à l'échelle: consommation limitée des ressources réseaux.
- Compatibilité ascendante (backward): l'interopérabilité avec le BGP conventionnel.
- Correction: atteindre l'état stable de la topologie.

Nous n'avons pas pu trouver un outil de validation qui nous permette de prouver toutes les propriétés susmentionnées. C'est pourquoi nous avons employé plusieurs méthodes pour valider nos approches séparément. Pour notre proposition eBGP, nous comptons sur SSFNet comme un cadre de simulation pour prouver que notre solution atteint la connectivité continue, et de mesurer le temps de convergence ainsi que la propagation des pannes. Nous avons ensuite utilisé l'algèbre de routage pour valider la propriété de sreté de notre approche.

Dans notre proposition iBGP, nous avons utilisé l'approche Rocketfuel pour récupérer des topologies internes de certains grands AS. Nous avons appliqué notre approche sur les topologies et comparer nos résultats avec l'une des meilleures configurations de réflexion de routes. Nous illustrons que nous produisons moins de sessions iBGP. Cela garantit le passage à l'échelle de Skeleton. Nous validons la sreté, la robustesse, et les propriétés de correction en utilisant des conditions de correction suffisantes prises de (17; 20; 30).

# Contents

# List of Figures

## LIST OF FIGURES

# Chapter 1

# Introduction

BGP (Border Gateway Protocol) (64) is the routing protocol that is responsible of calculating and exchanging routes between all the ASes (Autonomous Systems). Once this routing protocol suffers from any type of anomalies, then a traffic loss might occur somewhere in the Internet. A protocol that organizes this huge topology should be careful when taking reactions as a response to various events appeared in that topology. This carefulness makes the Internet more quiet and stops the amplification of routing events. However, this also has bad effect on delaying some necessary reactions.

BGP works in two modes, external and internal. EBGP (External BGP) is responsible of inter-AS routes exchanges, while iBGP (Internal BGP) diffuses externally learned routes inside ASes. Routing events may come from either inter-AS policy change or even from inside some ASes. Therefore, both external and internal stability have large impact on the global Internal stability (6). This creates the needs to deeply study BGP in its two modes and look for solutions to fix possible divergence problems. In this work we propose two solutions to stabilize BGP in its two modes. To validate our proposals, we employ several tools and frameworks. The validation process is dependent on the protocol we are illustrating. EBGP and iBGP are similar in their structures but each one of them has its own concerns. For example, the convergence time is a real concern in eBGP but it has not the same importance in iBGP. In the opposite side, the number of BGP sessions causes a scalability concern in iBGP, where it is not in eBGP. Therefore, the properties we would like to validate for each mode are different. So the first step will be to determine what type of problems we are targeting for each

mode, then we present our solutions to fix those problems. Finally, we show that our proposals satisfy the waited properties.

This motivated us to use several Internet modeling schemes to validate some of the correctness properties of our approach. Internet modeling is a large domain that covers numerous topics, like topology modeling (55), behavior modeling (29), relationship modeling (38), routing modeling (30), etc. The topic we are tackling is modeling the routing process in the Internet, we did this in two directions: topological and functional. Topological model is built according to the fact that BGP is a parallel process that runs simultaneously for each prefix. This helps us in parallelizing the routing process into sub-processes. Each sub-process represents an instance of the routing process that is run in a certain node $n$ to reach certain prefix $\rho$. $n$ may receive signaling messages from all its neighbor, but it selects one of them as a best path to reach $\rho$, unless it employs load balancing technique. Therefore, the traffic that is either generated by $n$ or transiting it follows the selected best path to reach $\rho$. If we look at all the selected paths to reach $\rho$, we notice that they form a tree rooted at the node that announces $\rho$. Each path between a leaf node (end customer AS) and the hub node is a branch in that topology.

Functional model is an algebraic structure $(\Pi, \preceq, L, \oplus, \phi)$ that is composed of a set of composite policies $\Pi$, a set of labels $L$, special additive operation $\oplus$ to calculate a policy of concatenated paths, an order $\preceq$ on $\Pi$ to rank paths according to their policies, and a null policy $\phi$ that represents a non valid path. This algebraic structure represents the PBR (Policy Based Routing) algebra, that has been used to model the routing decision process. Further, PBR algebra's properties allow validating safety and robustness characteristics of our proposed solutions which are summarized in the next two sections.

## 1.1 External BGP

EBGP is the dominant interdomain routing protocol in the Internet. Each router in the Internet has BGP table that contains all Internet prefixes. BGP employs several timers to control the signaling flow, these timers cause slow BGP convergence. BGP convergence is the process that runs after the occurrence of topology change, its role

is to find new paths for every affected prefix. BGP convergence time is the amount of time needed by the convergence process to get back to the stable state.

BGP convergence time might take several minutes during which the topology may suffer from traffic loss while the underline topology is connected. This motivates us to look for a solution for this transient disconnectivity. Two ideas were investigated: First, convergence time reduction by tuning BGP timers and employing trigger updates. Second, using temporary path during the convergence process. We found out that reducing convergence time has several bad effects on network stability. Thus we worked on the second idea, that is to save a backup path for each prefix and use it in case of losing the primary path.

Therefore, the objective of our proposed eBGP enhancements is to eliminate the packet loss during the convergence process without a huge increase the convergence time. The validation of this solution should ensure that there is no packet loss during the re-convergence process. This is done by using the simulation environment SSFNet (3), where the topologies are generated using the topology generators BRITE (60). Validating this proposal shows also that the enhanced eBGP will be free of loops. This is done theoretically using algebraic framework (30).

Our approach to enhance eBGP is based on four mechanisms that can be summarized as follows:

- during the stable state, each node calculates a list of backup paths and rank them according to their disjointness from the primary path.
- Once a failure is detected, the concerned (that are directly connected to the failure) nodes add root cause information to the withdraw messages.
- When a node receives a withdraw message containing root cause information, it filters out the backup paths that contain the root cause node and sends a withdraw message to a selected group of its neighbors.
- The nodes that install backup paths run a new timer, upon its expiration they update their Loc-RIB by replacing the backup path with the best path according to BGP preference.

The obtained simulation results show that we have zero packet loss upon failures occurrence. Further, and due to the selective withdraw, we limited the number of ASes

affected by failures, which decreases the number of nodes that unleash their convergence process.

In addition to the backup path disjointness enforcement, we apply safety conditions to ensure that the transient connectivity respects the inter-AS relationships. This makes the converged, and converging, topology safe according to (30).

## 1.2 Internal BGP

Intra-AS stability has equivalent importance on Internet stability as the Inter-AS one. Three known intra-AS configurations are used currently inside ASes:

1. Full mesh: it states that each ASBR (AS Border Router) has iBGP sessions with all other ASBRs and all internal BGP speakers. The only concern that this solution suffers from is the scalability. Although it is optimal for small autonomous systems, but it does not scale for large networks because it produces a number of iBGP sessions of the order $O(n^2)$, $n$ is the number of ASBRs. This consumes routers' resources and affects badly the network performance.

2. AS confederation: it partitions the autonomous system into sub-ASes, it employs the full mesh solution inside each sub-AS, and employs enhanced version of eBGP between sub-ASes. The problem from which this solution suffers is that it increases the convergence time needed after internal failures.

3. Route reflection: It partitions the autonomous system into clusters, routers insides each cluster are divided into reflectors and clients. Clients establishes iBGP sessions with one or more reflectors. Route reflectors are fully meshed between each other. Clients have no complete visibility of the internal topology, and their decisions are enforced by the reflectors, which causes non-optimal routing decisions. Moreover, signaling and forwarding loops are more likely to happen when there are routers whose signaling paths are different from their best forwarding one.

Our objective is to give a solution that is scalable (limited number of iBGP sessions) and gives complete visibility to all routers. Our proposed solution, the skeleton, is based on three main ideas: First, each internal node establishes an iBGP session with each

best next hop towards each ASBR. Second, iBGP sessions are established between one-hop neighbors only. Third, all skeleton nodes are capable of reflecting routes. Results show that skeleton produces much more less iBGP sessions than any known iBGP configurations. We prove, using routing algebra, that skeleton configuration is safe (free of routing anomalies).

To increase the robustness of skeleton against internal failures, we have suggested additional iBGP sessions to be used as backup ones. Backup selection method is similar to the one used for eBGP and based on our proposed topological model.

## 1.3 Reader's Guide

The remainder of this dissertation is organized as follows:

- **Chapter 2** introduces the Internet topology and talks about its history from the time where it was connecting few sites to the current time where it connects tens of thousands of autonomous systems. This chapter gives also an overview of the routing protocol BGP in its two modes, it shows the different types of routing anomalies that BGP suffers from. We present in this chapter the modeling scheme we will use to validate the proposed solutions. We then end this chapter by talking about the motivations and objectives of our work.
- **Chapter 3** scans some of the related works of enhancing eBGP and details our approach to stabilize eBGP. It presents the four mechanisms used in our proposal to guarantee the absence of packet loss during the convergence process of eBGP.
- **Chapter 4** validates our proposed approach to enhance eBGP. We first prove that our proposal is safe using the routing algebra modeling scheme. Then we show our simulation results using SSFNet, and finally we analyze them and discuss the perfectioning of our approach.
- **Chapter 5** gives an overview of some works related to the enhancements of iBGP configurations. It details the our approach (Skeleton) as a new iBGP configuration that guarantees the safety and stability of AS internal topology.
- **Chapter 6** validates the Skeleton approach by proving the scalability, correctness, safety, and robustness of Skeleton as an intra-AS configuration.

- **Chapter 7** concludes this dissertation and proposes possible future directions to be followed in interdomain routing.

# Chapter 2

# Background

This chapter gives an overview of the Internet from its early history till the current huge topology in its first section. Section II talks in details about the routing protocol BGP (Border Gateway Protocol) along with the routing anomalies that may happen upon topology changes. Section III introduces the topological modeling scheme used for routing protocols. Section IV presents the routing algebra used to validate our proposals. We end this chapter by listing our motivations behind this work as well as the objectives that we try to achieve.

## 2.1 Internet Overview

### 2.1.1 Internet History

In 1957 Soviet Union sent the first unmanned satellite (Sputlink 1) into orbit. In order to secure America's lead in technology, United States found DARPA (Defense Advanced Research Project Agency) in February 1958. DARPA planned a large scale computer networks in order to accelerate knowledge transfer. This network would become the ARPANET. Furthermore, three other concepts were to be developed which are fundamental for history of the Internet:

- The concept of a military network by the RAND corporation of America.

- The commercial network of NPL (National Physics Laboratory) in England. NPL is the founder of packet switching approach.

- The scientific network, Cyclades, in France. Cyclades was composed of small networks connected together. In contrast with ARPANET, during communication between sender and receiver the computers were not to intervene anymore, but simply serve as a transfer node. This was implemented at the physical layer.

Inspired by the Cyclades network and driven by the incompatibility between networks, the international organization for standardization found the OSI (Open Standard Interconnection) reference model. The innovation of the OSI reference model is to standardize the network from its ends. Then TCP assimilated the preference of OSI model and gave the way to TCP/IP protocol family. A standard which guaranteed compatibility between networks and merged them creating the Internet. By February 1990, the ARPANET hardware was removed but the Internet was up and running with less than one thousand FIB (Forwarding Information Base) entries (2).

From its beginning, the Internet appeared as a network of networks. Each of those networks can have its own topology and technology, and they are managed autonomously, that is why we call each of them an AS (Autonomous Systems). However, two other components form the Internet: Routing protocol and inter-AS relationships, which we introduce in the following two subsections.

### 2.1.2 Internet Relationships

Understanding the topological structure of the Internet gives the ability to predict routes in the Internet. However, the Internet is composed of large number of ASes resulting in complex interactions, which motivates several works like (15; 16; 33; 63) to propose prediction methods that infer the inter-AS relationships and their types.

Gao (16) was the first to formulate and systematically study the AS relationships inference problem. Gao assumed that every BGP path must comply with the following hierarchical pattern: an uphill segment, followed by zero or one peer-peer link, followed

by a downhill segment. Paths with this hierarchical structure are valley-free or valid. Paths that do not follow this hierarchical structure are called invalid and may result from BGP misconfigurations. Following this definition of valid paths, Gao proposed an inference heuristic that identified top providers and peering links based on AS degrees and valid paths.

Following (16) Subramanian *et al.* (33) developed a mathematical formulation of the inference problem. They cast the inference of AS relationships into the Type of Relationship (ToR) combinatorial optimization problem. The authors speculated that ToR is NP-complete and developed a heuristic solution that takes as input the BGP tables collected at different vantage points and computes a rank for every AS. The heuristic then infers AS relationships by comparing ranks of adjacent ASs.

Di Battista et al. (15) showed that ToR is indeed NP-complete, developed mathematically rigorous approximate solutions to the problem and proved that it is impossible to infer peer-peer relationships under the ToR formulation framework. For this reason, their solutions infer customer-peer relationships only and ignore peer-peer and sibling-sibling relationships.

The relationships among ASs in the Internet represent the outcome of policy decisions governed by technical and business factors of the global Internet economy. Precise knowledge of these relationships is therefore an essential building block needed for any reliable and effective analysis of technical and economic aspects of the global Internet, its structure, and its growth. Links between ASes are controlled financially depending on which AS provides service to which AS (25; 53; 63). Thus, ASes can be divided into levels according to their positions in the Internet hierarchy. Top level ASes provide Internet service to all their neighbors and do not buy service from any AS, while bottom level ASes buy service from several ASes and do not provide service to any AS. This gives a label for each link according to the relationship type between the two end ASes. We mean by an AS relationship the established logical session between two AS peers, which may be carried over one or more physical links, it can be categorized either per prefix or per peer. A per peer AS relationship can have one of the following types:

- *customer-provider*: is a link between two ASes one of them, the provider, provides (sells) Internet service to the other, the customer. From the provider point of view, this link is labeled as a *customer* link, and is labeled as *provider* link from customer's point of view.

- *peer-peer*: is a voluntary interconnection of administratively separate Internet networks for the purpose of exchanging traffic between the customers of each network. In such relationship, neither party pays the other for the exchanged traffic; instead, each derives revenue from its own customers [1].

A per prefix relationship comes from the complex commercial agreements between AS nodes, where we can find an inter-AS link between two ASes $a$ and $b$ representing customer relationship in both directions. This happens when we look at this relationship from prefix point of view, so it can be seen as a customer link for a subset of prefixes and a provider one for another subset. In other words, $a$ is a provider of $b$ for a set of prefixes $S_1$ and $b$ is a provider of $a$ for a set of prefixes $S_2 \neq S_1$ and peers elsewhere.

Inter-AS relationships play an important role in routing decision nowadays. They are implemented in a per router basis using the optional transitive communities attribute (11; 48). However, this attribute does not give the Internet nodes the sufficient information of all the inter-AS relationships along the received path. In this dissertation, we propose a new attribute that has the same format of AS_Path attribute and carries the relationships types between all the paths' routers.

### 2.1.3 Interdomain Routing

We have seen in the previous subsection that the Internet is a set of connected networks (ASes or domains), paths between these ASes are calculated using EGP (Exterior Gateway Protocol). EGP is responsible of diffusing IP prefixes between ASes and

[1] Another types like backup or sibling may exist in the Internet, they can be considered as special type of *customer-provider* or peering links.

IGP (Interior Gateway Protocol) is responsible of exchanging routes inside each AS. Interdomain routing is the process of finding best paths between different domains or ASes.

The rapid increase in IP prefixes makes the needed number of interdomain paths huge, and the increase in Internet nodes (ASes) increases the number of possible paths for each IP prefix. This complicates the Internet topology and arises the stability and scalability concerns for the protocol that manages it, that is the routing protocol. These challenges causes the EGP protocols to diverge or to have slow convergence. Since BGP (Border Gateway Protocol) is the dominant interdomain routing protocol in the Internet, this makes it a target for several research subjects to resolve those challenges. Next section gives an overview of BGP in its two modes eBGP (External BGP) and iBGP (Internal BGP).

## 2.2 Border Gateway Protocol

BGP is a path vector protocol used to carry routing information between ASes. The term path vector comes from the fact that BGP routing information carries a sequence of AS numbers that identifies the path of ASes that a network prefix has traversed. The path information associated with the prefix is used to enable loop prevention. BGP is a policy based routing protocol due to its metric, which is based on preference of several attributes.

BGP uses TCP (Transport Control Protocol) as its transport protocol (port 179). This ensures that all the transport reliability is taken care of by TCP and does not need to be implemented in BGP, thereby simplifying the complexity associated with designing reliability into the protocol itself.

Routers that run a BGP routing process are often referred to as BGP speakers. Two BGP speakers that form a TCP connection between one another for the purpose of exchanging routing information are referred to as neighbors or peers.

Most of BGP's implementations employ two RIBs (64) (Routing Information Base),

the first one is the Adj-RIB-In in which the router stores the NLRIs (Network Layer Reachability Information) learned by its neighbors. The second one is the Loc-RIB which is a subset of the Adj-RIB-In, it contains only the best route for each NLRI. This best route is to be announced to the peers.

### 2.2.1 BGP Messages

- OPEN message: to open a BGP session between peers.
- UPDATE message: to transfer reachability information among peers. This message is used to either advertise a feasible route to a peer or withdraw infeasible routes. The UPDATE message is usually referred to as a BGP advertisement.
- NOTIFICATION message: sent when an error condition is detected. The BGP session is immediately shut down after this message is sent.
- KEEPALIVE message: periodically exchanged to verify that the peer is still reachable.

### 2.2.2 BGP Attributes

A variable-length sequence of path attributes is present in every UPDATE message, except for an UPDATE message that carries only the withdrawn routes. Each path attribute is a triple (attribute type, attribute length, attribute value) of variable length. BGP attributes are divided into four categories:

1. Well-known mandatory: An attribute that has to exist in the BGP UPDATE packet. It must be recognized by all BGP implementations. If a well-known attribute is missing, a NOTIFICATION error is generated, and the session is closed. This is to make sure that all BGP implementations agree on a standard set of attributes. An example of a well-known mandatory attribute is the AS_PATH attribute.

2. Well-known discretionary: An attribute that is recognized by all BGP implementations but that might or might not be sent in the BGP UPDATE message. An example of a well-known discretionary attribute is LOCAL_PREF.

3. Optional transitive: An attribute that is not required to be supported by all BGP implementations. If an optional attribute is not recognized by the BGP implementation, that implementation looks for a transitive flag to see whether it is set for that particular attribute. If the flag is set, which indicates that the attribute is transitive, the BGP implementation should accept the attribute and pass it along to other BGP speakers.

4. Optional nontransitive: When an optional attribute is not recognized and the transitive flag is not set, which means that the attribute is nontransitive, the attribute should be quietly ignored and not passed along to other BGP peers.

BGP path attributes, as defined in (64), are as follows:

- ORIGIN: is a well-known mandatory attribute that defines the origin of the path information.

- AS_PATH: is a well-known mandatory attribute that is composed of a sequence of AS path segments.

- Next_Hop: is a well-known mandatory attribute that defines the (unicast) IP address of the router that should be used as the next hop to the destinations listed in the NLRI field of the UPDATE message.

- MULTI_EXIT_DISC: is an optional non-transitive attribute. The value of this attribute may be used by a BGP speaker's decision process to discriminate among multiple entry points to a neighboring AS.

- LOCAL_PREF: is a well-known attribute. A BGP speaker uses it to inform its other internal peers of the advertising speaker's degree of preference for an advertised route.

- ATOMIC_AGGREGATE: is a well-known discretionary attribute and used when a BGP speaker aggregates several routes for the purpose of advertisement to a particular peer.
- AGGREGATOR: is an optional transitive attribute, which may be included in updates that are formed by aggregation.

The adaptability of BGP has come from its attributes and the ability to add new ones as optional attributes, without raising any compatibility problem with another implementation.

Routing protocols calculate a numeric value for each route and then compare between them to select the best one. Apart from those routing protocols, when a BGP speaker receives more than one route towards certain destination, it does not calculate one value for each route, instead, it compares its attributes one by one using the following order:

1. It prefers the highest local preference value.
2. If equal, it prefers the shortest AS path.
3. If equal, it prefers the lowest origin code.
4. If equal, it prefers the lowest MED.
5. If equal, it prefers the closest IGP neighbor.
6. If equal, it prefers the lowest next hop BGP ID (tie breaking).

### 2.2.3 BGP Timers

BGP implements several timers to control the flow of its messages and to limit its control message overhead. Those timers increase the time needed for new changes in the topology to propagate, and hence increase the time needed for this topology to converge, which leads to a notable disconnectivity. But at the same time, those timers

have an important role in stabilizing the Internet and in protecting from the flapping routes. BGP employs five timers:

1. ConnectRetry Timer: once expires, it drops the TCP connection and releases all BGP resources.

2. Hold Timer: its expiration indicates that the BGP connection has completed waiting for the back-off period to prevent BGP peer oscillation.

3. Keepalive Timer: it determine the interval between two successive KEEPALIVE messages.

4. MAOI (Min AS Origination Interval) Timer: it determines the minimum amount of time that must elapse between successive advertisements of UPDATE messages that report changes within the advertising BGP speaker's own autonomous systems.

5. MRAI (Min Route Advertisement Interval) Timer: it determines the minimum amount of time that must elapse between an advertisement and/or withdrawal of routes to a particular destination by a BGP speaker to a peer. This rate limiting procedure applies on a per-destination basis, although its value is set on a per BGP peer basis.

### 2.2.4 BGP Finite State Machine

1. Idle State: This is the first stage of the connection. BGP is waiting for a Start event, which is initiated by an operator or the BGP system. An administrator establishing a BGP session through router configuration or resetting an already existing session usually causes a Start event. After the Start event, BGP initializes its resources, resets a ConnectRetry timer, initiates a TCP transport connection, and starts listening for a connection that may be initiated by a remote peer. BGP then transitions to a Connect state. In case of errors, BGP falls back to the Idle state.

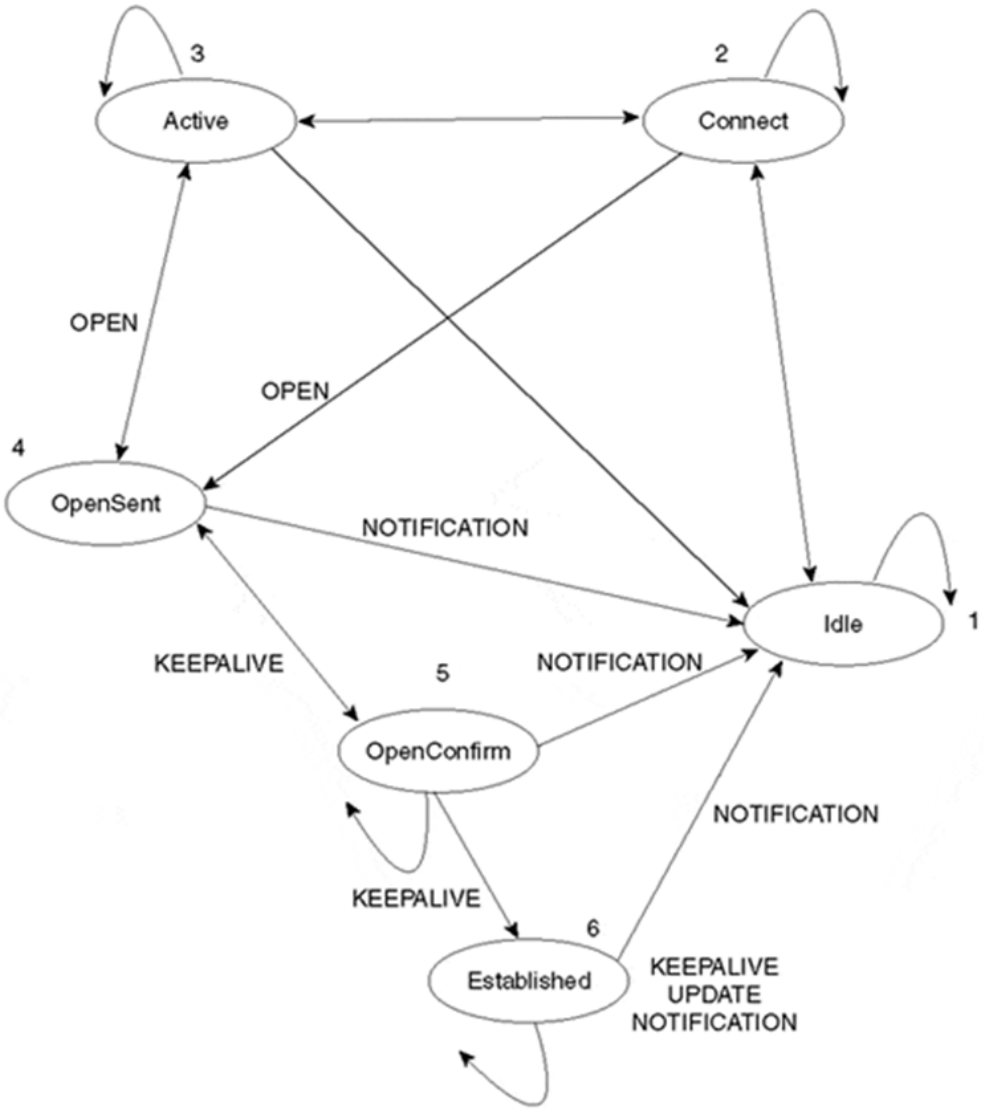


**Figure 2.1:** BGP Finite State Machine (24)

2. Connect State: BGP is waiting for the transport protocol connection to be completed. If the TCP transport connection is successful, the state transitions to OpenSent (this is where the OPEN message is sent). If the transport connection is unsuccessful, the state transitions to Active. If the ConnectRetry timer expires, the state remains in the Connect stage, the timer is reset, and a transport connection is initiated. In case of any other event (initiated by system or operator), the state goes back to Idle.

3. Active State: BGP tries to acquire a peer by initiating a transport protocol connection. If the transport connection is established, it transitions to OpenSent (an OPEN message is sent). If the ConnectRetry timer expires, BGP restarts the ConnectRetry timer and falls back to the Connect state. In addition, BGP continues to listen for a connection that might be initiated from another peer. The state might go back to Idle in case of other events, such as a Stop event initiated by the system or the operator.

4. OpenSent State: BGP is waiting for an OPEN message from its peer. The OPEN message is checked for correctness. In case of errors, such as a bad version number or an unacceptable AS, the system sends an error NOTIFICATION message and goes back to Idle. If there are no errors, BGP starts sending KEEPALIVE messages and resets the KEEPALIVE timer. At this stage, the hold time is negotiated, and the smaller value is taken. In case the negotiated hold time is 0, the Hold Timer and the KEEPALIVE timer are not restarted.

   At the OpenSent state, the BGP recognizes, by comparing its AS number to the AS number of its peer, whether the peer belongs to the same AS (Internal BGP) or to a different AS (External BGP).

   When a TCP transport disconnect is detected, the state falls back to the Active state. For any other errors, such as an expiration of the Hold Timer, the BGP sends a NOTIFICATION message with the corresponding error code and falls back to the Idle state. Also, in response to a stop event initiated by the system

or the operator, the state falls back to the Idle state.

5. OpenConfirm State: BGP waits for a KEEPALIVE message. If a KEEPALIVE is received, the state goes to Established, and the neighbor negotiation is complete. If the system receives a KEEPALIVE message, it restarts the Hold Timer (assuming that the negotiated Hold Time is not 0). If a NOTIFICATION message is received, the state falls back to the Idle state. The system sends periodic KEEPALIVE messages at the rate set by the KEEPALIVE timer. In case of any transport disconnect notification or in response to any stop event (initiated by the system or the operator), the state falls back to Idle. In response to any other event, the system sends a NOTIFICATION message and returns to the Idle state.

6. Established State: This is the final stage in the neighbor negotiation. At this stage, BGP starts exchanging UPDATE packets with its peers. Assuming that it is nonzero, the Hold Timer restarts at the receipt of an UPDATE or KEEPALIVE message. If the system receives any NOTIFICATION message (if an error has occurred), the state falls back to Idle.

   The UPDATE messages are checked for errors, such as missing attributes, duplicate attributes, and so on. If errors are found, a NOTIFICATION message is sent to the peer, and the state falls back to Idle. If the Hold Timer expires, or a disconnect notification is received from the transport protocol, the system falls back to the Idle state.

### 2.2.5 BGP Convergence

BGP sends updates only upon topology changes. Once a topology change (physical link failure, router failure, policy change in either originating AS or one of the transit ASes, or addition/deletion of network prefixes) takes place, BGP speakers react by starting their decision process and calculate the new routes according to the new topology. The time between the instability event and the stable state is referred to as convergence time.

BGP convergence time can be divided into two levels: Internet-wide level and node level. Internet-wide convergence is a theoretic definition that refers to the global stable state of the Internet, which states that each node for each of its RIB's prefixes has reached the stable state. However, the global state can not be measured or predicted since there is no single authority that manages this huge topology. The node level convergence time is calculated on a per-prefix basis. In our work, as stated in chapter 3, we divide the convergence process into a number of parallel processes equal to the number of prefixes. Therefore, the convergence time we are tackling in our study is $\Delta t_n^\rho = t_n^2 - t_n^1$, where: $\Delta t_n^\rho$ is the convergence time needed by node $n$ to re-converge to a stable state concerning the prefix $\rho$. $t_n^1$ is the instance at which node $n$ lost the prefix $\rho$, $t_n^2$ is the instance at which node $n$ has a stable path towards $\rho$.

The delay caused by BGP convergence time is composed of two intervals: the delay needed by BGP timers and UPDATE messages to propagate between peers, and the delay needed by the decision process. This latter consists of three phases:

1. Phase I: Each route received from a BGP speaker in a neighboring AS is analyzed and assigned a preference level. The routes are then ranked according to preference, and the best one for each network advertised to other BGP speakers within the autonomous system.

2. Phase II: The best route for each destination is selected from the incoming data based on preference levels, and used to update the Loc-RIB.

3. Phase III: Routes in the Loc-RIB are selected to be sent to neighboring BGP speakers in other ASes.

BGP convergence is the process of calculating best paths (highest policy) towards each Internet prefix. A converged network is the one in which each node has available path towards each Internet prefix. The importance of BGP convergence appears when failures take place somewhere in the Internet. Topology's reaction between the failure and the converged state is called the convergence process. Optimal convergence should

satisfy three characteristics:

**C1.** Best path selection: Each Internet node should receive enough information to be able to select the best path.

**C2.** Robustness: Recovery after failures should be immediate.

**C3.** Safety: Safe topology is the one that has neither oscillating paths nor routing loops.

Conventional BGP convergence process violates C1 because the only information available along BGP path is the AS_PATH attribute, which is not sufficient to select paths that fit the needs of ASes. To reach the robustness, a failover mechanism should exist which is not the case with conventional BGP. Therefore, it violates C2. BGP employs the MRAI (Minimum Router Advertisement Interval) timer for route flap dampening, and uses the AS_PATH attribute to avoid routing loops. However, these mechanisms are not sufficient to ensure topology safeness (9; 58).

### 2.2.6 Internal BGP

EBGP is the mode that control BGP sessions between two routers related to two different ASes. IBGP is the mode that control BGP sessions between routers in the same AS. EBGP sessions are established only between two BGP speakers that are physically connected. However, iBGP may establish sessions between two speakers that are not directly connected, that is because iBGP depends on an IGP (Interior Gateway Protocol) routing protocol to calculate the path between the two end peers.

Routers that have both eBGP and iBGP neighbors are called ASBRs (AS Border Routers). They are the responsible of diffusing externally learned routes within their own AS. (64) states that an iBGP speaker is not able to re-advertise internally learned routes from its iBGP peers, that is because iBGP has no loop detection mechanism. Therefore, full mesh topology should be maintained among the ASBRs, thereby, each internal node should establish iBGP sessions with all ASBRs to guarantee complete visibility. However, iBGP full mesh has scalability concerns:

- Number of iBGP sessions: For an AS that has $n$ internal router and $a$ ASBR,

the needed number of iBGP sessions in a full mesh topology is: $na + \frac{a(a-1)}{2}$. This may cause extra overhead on ASBRs.

- Administration: The configuration management of this huge number of sessions is tough.

- BGP table size: Increased number of neighbors increases the number of paths for each route, which leads to memory consumption.

To resolve the scalability concerns of full mesh iBGP, two alternative configurations have been appeared:

#### 2.2.6.1 AS Confederation

AS confederation (47) divides the AS into smaller ASes. Each sub-AS has a private AS number and contains a set of routers. IBGP topology inside each sub-AS is full mesh. BGP sessions between routers from different sub-ASes are managed by an enhanced version of eBGP. This new protocol that might be called as eiBGP (External iBGP), has enhanced AS_PATH attribute to include sub-AS information. It has also enhancements related to the handling of MED, Loc_Pref, and Next_Hop attributes.

AS confederation needs attentive design method. Increasing the number of sub-ASes decreases the number of routers inside each sub-AS and, in turns, decreases the number of needed iBGP sessions, which is a positive point. At the same time, this increases the number of eiBGP sessions, which creates scalability concern. Further, increased number of eiBGP sessions causes inherited divergence problem since eiBGP is an enhanced version of eBGP, which is known to converge slowly.

Vice versa, Decreasing the number of sub-ASes increases the number of routers inside each sub-AS and, in turns, increases the number of iBGP sessions, which arises a scalability concern. However, the positive point is that it simplifies the eiBGP topology.

#### 2.2.6.2 Route Reflection

RRS (Route Reflections Solutions) (57) was motivated by the scalability problem of FMS (Full Mesh Solution). RRS successfully resolves scalability concerns by reducing the needed number of iBGP sessions. However, it causes new types of divergence problems and configuration dependent routing loops and oscillations.

BGP route reflection is based on reducing the number of iBGP sessions by giving certain nodes (Route Reflectors) the ability of redistributing the received routes. RRS divides the AS into clusters, each cluster has one or more RR and the rest of nodes are their clients. RRs establish iBGP sessions with their clients and their non-client peers. Top level RRs should be fully meshed. When RR receives a route from an iBGP peer, it selects the best path based on its path selection rule. If the path was received from a non-client peer then it reflects it to all its clients. If the path was received from a client, it reflects it to all its clients and non-clients peers. Clients' paths are limited to those advertised by their RRs. For the sake of redundancy, clients may establish iBGP sessions with RRs either in the same or different cluster.

Employing RRS may lead to the following signaling and forwarding anomalies:

1. Non complete visibility:

   Using RRS does not guarantee that each node would have a complete visibility. That is because internal nodes do not receive routing updates from all the advertising ASBRs, which leads to an incomplete information and, in turn, to a non-optimal decision of the best BGP next hop (ASBR). Consider the topology in figure 2.2, continuous lines indicate physical links and dash ones indicate iBGP sessions. The numbers next to the continuous lines are the IGP costs.

   Figure 2.2 shows a simple scenario in which an internal node's traffic quit the AS from a non optimal egress node (if hot potato is implemented). C2 gets all the routes from RR, it does not receive any route from C1 or C3. Thus C2 will never know about paths from C1. If a prefix $\rho$ is advertised by both C1 and C3, the traffic from C2 will quit the AS from C3 where the best path is via C1.

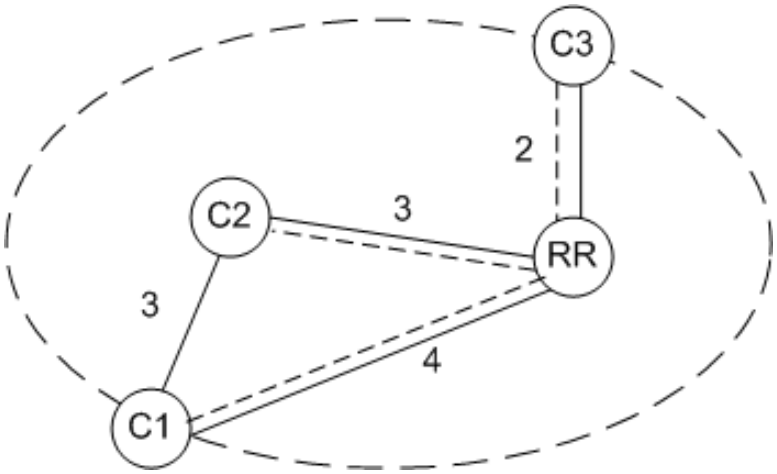


**Figure 2.2:** Non complete visibility

2. Route Oscillation:

   Route oscillation occurs when the topology is no longer able to converge to a stable state. This happens when a subset of routers within an AS exchange routing information forever without being able to settle on a stable routing configuration. Misconfiguration is the main reason behind the route oscillation. When internal clusters contain nodes whose preferred egress points are not in the same cluster, then an oscillation might occur. For example, consider the configuration in figure 2.3 [1] where it is not possible to settle on a stable state. The preferred egress point of the route reflector of each cluster is in the neighboring cluster because $b_i < a_i$. $RR_i$ oscillates between two states. 1) It selects $\alpha_i$ as preferred exit point. 2) It selects $\alpha_{i+1}$ as preferred exit point and withdraws $\alpha_i$.

   The $l$ clusters form a signaling loop which causes the route oscillation. Similar scenarios of oscillations might take place due to MED attribute (9) or path asymmetry (5). The main reason behind such signaling anomalies is the lack of visibility.

[1] Route reflectors are supposed to be fully meshed, but to avoid excessive clutter in the figure, we did not show all the iBGP sessions (dashed lines).

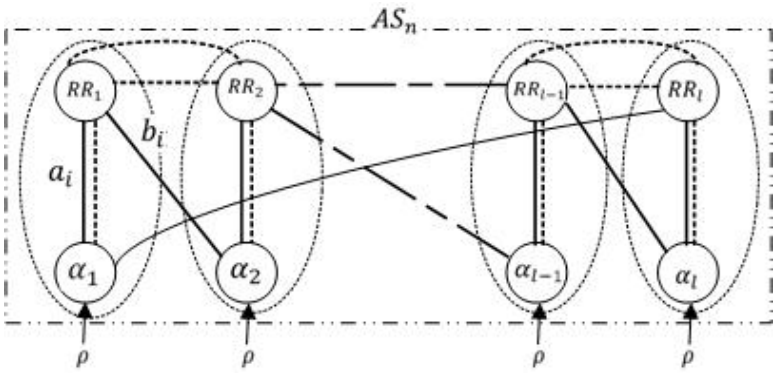


**Figure 2.3:** Route Oscillation

3. Routing Loops:

Routing loop is the passage of the routed traffic by the generating node more than one time. The reported reason behind iBGP forwarding loops is that the path between an internal node and its exit point contains at least one node that has different exit point. RRS topologies might cause routing loops either inside the AS or even outside (20). We illustrate only the internal routing loops. An example of a routing loop is shown in figure 2.4 where we have: $b_i > b_{i+1} + a_{i+1}$ .

Dispute wheel (20) is one of the configurations that causes forwarding loops. In figure 2.4, $n_i$ has an iBGP session with $RR_i$ from which it gets all the routing information. Its path to its egress node $RR_i$ passes by another node $n_{i+1}$. $n_{i+1}$ has a different egress node $RR_{i+1}$ and the path between them passes by another node and so on. If $n_i$ has an iBGP session with its preferred egress node $RR_{i-1}$ the forwarding loop will not occur. In other words, if internal nodes establish iBGP sessions based on the selection of preferred egress points, then we can have loop free topologies. This is the main idea of Skeleton as we will see later.

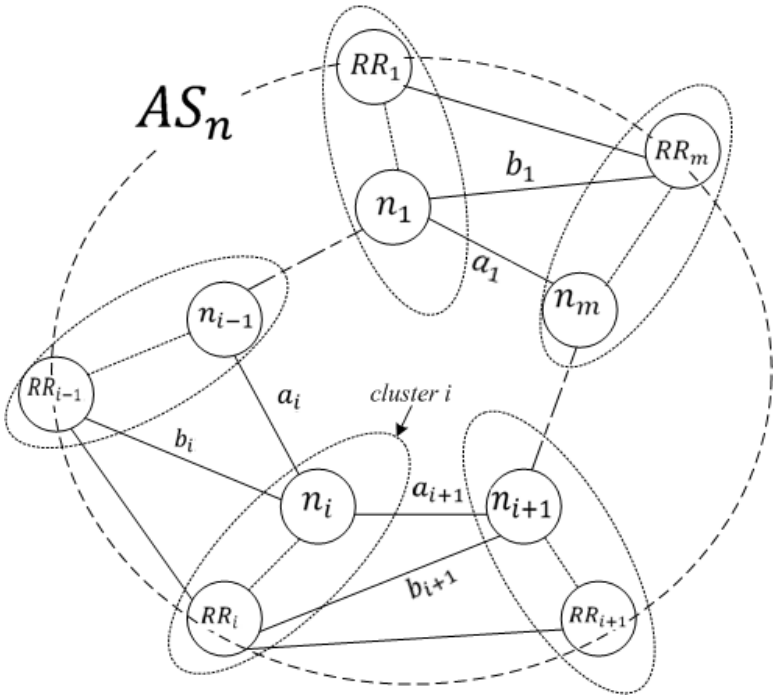


**Figure 2.4:** Forwarding loop (Dispute Wheel)

## 2.3 Internet Modeling

Internet topology is a graph of nodes, where nodes represent ASes (Autonomous Systems) and edges represent inter-AS links. This is a deterministic model that can be obtained in the stable state, as proposed in (23; 61). However, Internet is a changing topology which makes any fixed representation inaccurate. Topology changes, in terms of Internet routing, form routing events that may disrupt the stability of the routing protocol. Moreover, and because of that Internet is a huge topology, it takes routing protocol long time to reconverge after a routing event. The convergence of the Internet routing protocol is a challenging problem that can not be eliminated, but it can be enhanced.

Internet routing is a complex process, it runs between tens of thousands of ASes, carrying hundreds of thousands of prefixes (2). This makes it hard to be described formally. However, that process can be divided into parallel sub-processes that run

simultaneously for each separate prefix.

This section presents a model of Internet routing that describes both inter/intra AS topologies. In addition, it models the decision process of Internet routing protocol BGP using routing algebra. We first present the topological model of both inter/intra AS. We denote prefix-based Internet model for inter-AS topology, and ASBR-based model for intra-AS one. We then present the routing algebra of (30), which states that of certain properties are satisfied then the inter/intra AS topology converges to prefeix/ASBR based model.

### 2.3.1 Topological Modeling

The Internet is a network of networks, its representation depends on the level of its hierarchy we are illustrating. In our study we propose modeling schemes for both AS level and router level. We use the AS level model when studying an EGP protocol, and we use the router level when we study an IGP protocol. We show that when the external/internal topology is safe, then it converges to prefix/ASBR based model.

The topological modeling approach employs the graph theory in representing the topology we study. This gives a simple model on which we can easily understand and apply the functional model. Next subsection gives a quick overview of graph principles that we will use in our modeling scheme. Second subsection presents the prefix based model that looks at the Internet from AS level point of view. Third subsection apply the prefix based model on the intra-AS topology.

#### 2.3.1.1 Overview

A graph $G(V, E)$ is a representation of a set of vertices $V$ and a set of edges $E$ where each edge connects pair of vertices from $V$. In interdomain routing, $V$ represents the set of ASes, and each edge in $E$ represents an inter-AS physical link between two neighboring ASes. Some times there might be several physical links connecting two ASes. In such cases we represent them by a single edge. Our representation does not take into consideration the logical connections that might be established between

two far ASes through IP (Internet Protocol) tunnels. Graphs that have directed edges are called directed (or asymmetric) graphs and those with undirected edges are called undirected (or symmetric) graphs. In intra AS topology, the set $V$ represents the set of routers, and $E$ represents the physical links between routers. Routers that are not BGP speakers (i.e. they use static or default routes to exit the AS) are transparent.

A tree is an undirected graph in which any two vertices are connected by exactly one simple path. In other words, any connected graph without cycles is a tree. This definition fits converged networks and models the data plane. However, Internet nodes may have more than one path to reach each others. Moreover, a path from node $a$ to node $b$ might be different form the path from node $b$ to node $a$. This leads to that the represented graph should be directed and loop free. This modeling graph called DAG (Directed Acyclic Graph), which is a directed graph with no directed cycles. Our topological model uses DAGs to represents both the inter and intra AS topologies.

#### 2.3.1.2 Prefix-based Internet model

In network routing we distinguish between two planes: control plane and data plane. Control plane represents the signaling flow of the routing messages, the graph the models it is called the routing graph. Data plane represents the traffic flow according to the routing decision, the graph that models it is called forwarding graph. This distinction leads to different representation for routing and forwarding.

Network topology can be modeled as a graph, its nodes are either ASes or routers, its edges are physical links. Network convergence is a parallel process that is run for each prefix separately. This parallel process can be divided into a number of single processes equal to the number of prefixes.

Unless there exists a BGP prefix hijack, an Internet prefix may not be announced by more than one AS node. However, in some cases where a customer connects to several providers without being part of any of them, then each of those providers is considered as an announcing AS node.

The network can be represented as a DAG rooted at the prefix ($\rho$) announcing node

$d_\rho$. The resulted topology is a subgraph with the same set of vertices and a subset of edges, we call it $\rho$-converged subgraph $C_\rho$. This subgraph consists of branches $B_\rho$ terminated at hub node $d_\rho$, the number of these branches is limited by the number of directly connected neighbors of $d_\rho$.

Figure 2.5.a shows a sample topology, the paths to reach prefixes announced by nodes *f,d,b* form subgraphs as shown in figures 2.5.b,c,d respectively.

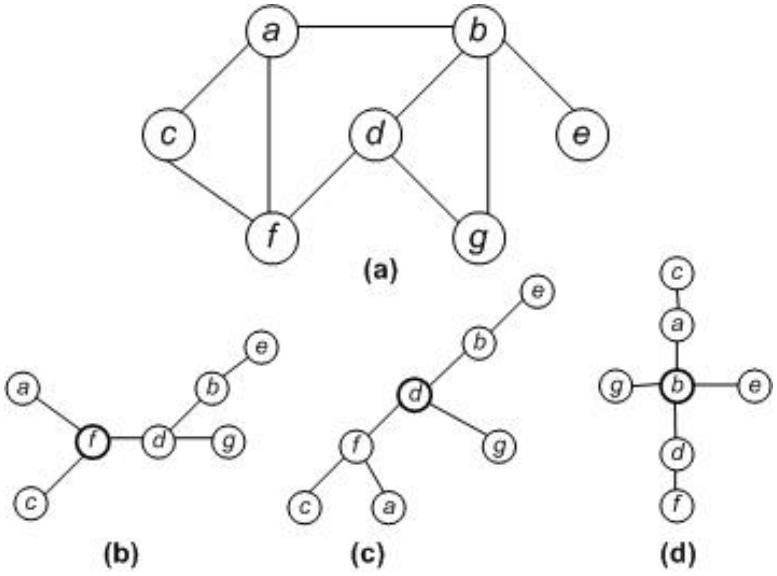


**Figure 2.5:** Prefix-based Model

DAG is a loop free graph, however, in case of failures, the topology will change and loops may take place. In figure 2.5.b, if node *d* fails, and both nodes *b,g* select each other to reach prefixes announced by *f*, then a forwarding loop will take place between *b* and *g*. To ensure the network stability after the occurrence of a failure, it should be guaranteed that the network keeps its structure as a DAG during and after the convergence process. This means that the topology type before and after the failure should follow the DAG model.

#### 2.3.1.3 ASBR-based AS model

ASBR-based model is a subgraph of the internal AS physical graph with the same set of nodes and a subset of intra-AS links. These subgraphs form DAGs rooted at the ASBRs. Branches of this topology are the best IGP (Interior Gateway Protocol) paths calculated between leaf routers and ASBRs. Figure 2.6 shows an example of AS topology with corresponding ASBR-based AS model, where A,B,C are ASBRs and 1,2,3,4 are internal routers. For the sake of simplicity, we suppose that all inter-router links have IGP metric value equal to 1.

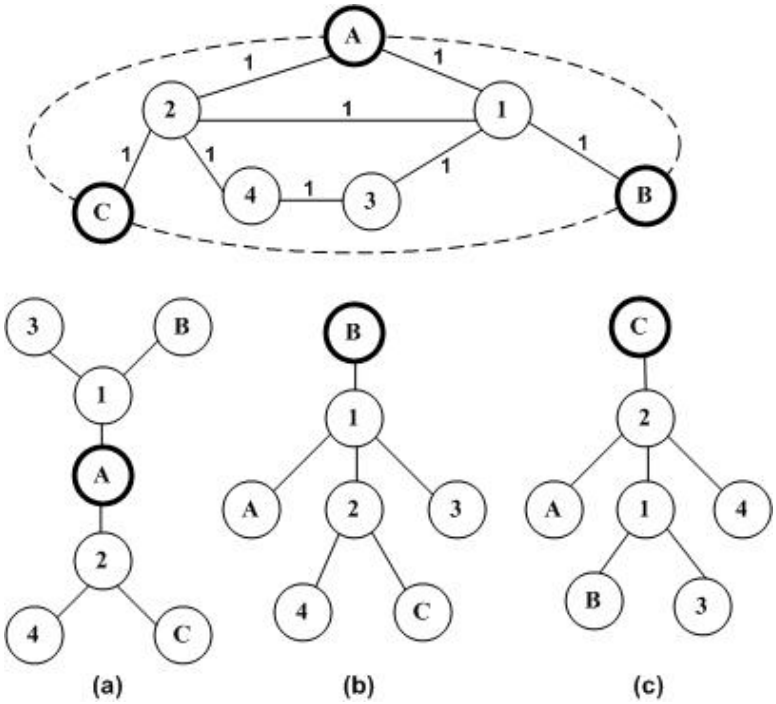


**Figure 2.6:** ASBR-based model

### 2.3.2 Functional Modeling

Modeling BGP convergence has been tackled by few works. Griffin and Sobrinho (22) proposed a high level declarative language based on the routing algebra framework of Sobrinho (30). The proposed language, RAML (Routing Algebra Meta Language), has

the key property that correctness conditions can be derived easily. It has the ability to express any routing protocol while retaining the ability to derive algebraic properties that is related directly to the convergence ones. Sobrinho (30) has proposed an algebra for routing based on walks and signatures. His work forms a good start in applying algebraic theory on network routing. Sobrinho proposed an algebra for routing as an ordered septet $(W, \preceq, L, \Sigma, \phi, \oplus, f)$ where:

- $W$ is a set of weights.
- $L$ is a set of labels.
- $\Sigma$ is a set of signatures.
- $\preceq$ is a total order on $W$.
- $\oplus$ is a binary operation that maps pairs with a label and a signature into a signature.
- $f$ is a function that maps signatures into weights.
- $\phi$ is a special signature.

This generic definition gives the possibility to choose the above sets and function according to the tackled subject. Once this algebra is determined according to a routing protocol, then all what we need is to prove the strict monotonic property of this algebra in order to show that it converges to an optimal state. This guarantees the safety and robustness characteristics of that protocol.

According to Sobrinho, the algebra is monotone if for all $l \in L$ and $\sigma \in \Sigma, f(\sigma) \preceq f(l \oplus \sigma)$, while the strict monotonicity condition is: for all $l \in L$ and $\sigma \in \Sigma, f(\sigma) \prec f(l \oplus \sigma)$. The strict monotonicity has an essential advantage over the monotonicity, for example, in next hop forwarding, the path cost from the next hop should be (strictly) less than the path cost of the current node. This is indispensable to avoid loops.

PBR (Policy Based Routing) protocol selects paths by preferring those with highest policies. Internet paths should be selected by calculating their policies, and then

selecting the compliant ones. Operations between policies need to be done in a well defined space.

We remind that the objective behind employing the routing algebra in our work is to ensure that our proposed solutions lead to optimal convergence; that is the convergence towards a DAG topology as defined in the previous section.

We reform the routing algebra to make it easier and simpler to be applied without affecting its generality. We define a composite set of path policies $\Pi$ which represents the of signatures $\Sigma$ in the routing algebra. We define the function $f$ is the *identity* function. This gives that the set of weights $W$ is identical to $\Pi$. Therefore, the routing algebra that we will work on, which we will call the PBR algebra, has the form: $(\Pi, \preceq, L, \oplus, \phi)$, where:

- $\Pi$ is a set of composite path policies.
- $\oplus$ is a binary operation on $\Pi$.
- $L$ is a set of labels.
- $\preceq$ is a total preorder on $\Pi$. Note that we relax the antisymmetric condition since it may not be true in all cases. For example, if we take the path policy represented by the AS_Path attribute, then we can have two path policies $\wp_1 = \{AS_a, AS_b, AS_c\}, \wp_2 = \{AS_\alpha, AS_\beta, AS_\gamma\}$. For some prefixes, $\wp_1$ might be preferred over $\wp_2$ and the opposite for other prefixes. However, $\wp_1 \neq \wp_2$.
- $\phi$ is the policy of a none valid path.

To guarantee the correctness of a protocol, it is sufficient to prove the following properties:

1. Order preservation: $\forall \ell \in L, \wp_b, \wp_c \in \Pi - \{\phi\}, \wp_b \preceq \wp_c : \ell \oplus \wp_b \preceq \ell \oplus \wp_c$.
2. Monotonicity: $\forall \wp_a \in \Pi - \{\phi\}, \forall \ell \in L : \wp_a \preceq \ell \oplus \wp_a$.
3. Strict monotonicity: $\forall \wp_a \in \Pi - \{\phi\}, \forall \ell \in L : \wp_a \prec \ell \oplus \wp_a$.

In chapters 4 and 6 we employ this algebra to prove the strict monotonicity property and thereby the safety and robustness of our proposed approaches.

## 2.4 Motivations and Objectives

The increasing number of AS nodes in the Internet, which exceeds nowadays 35,000 ASes, see figure 2.7, along with the complex relationships between them create severe challenges on the routing process. Moreover, the tremendous number of announced prefixes, see figure 2.8, increases the load on this process. With such a complex topology, BGP suffers from relatively long convergence time and is not guaranteed to converge (21). Moreover, its divergence may cause undesired packet loss (42).

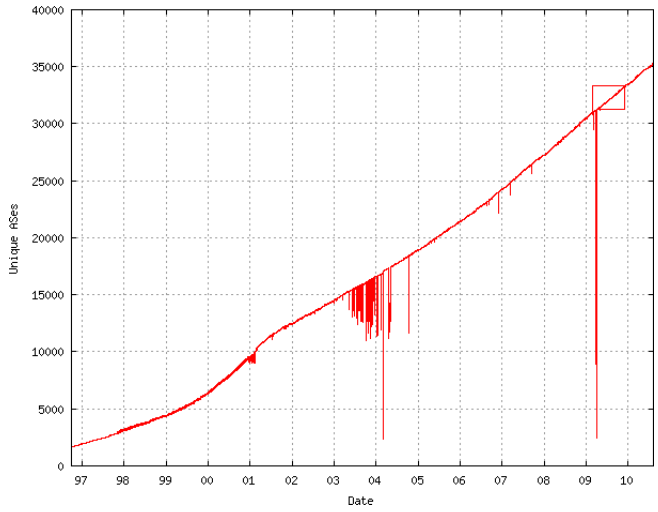


**Figure 2.7:** AS Counts (2)

However, stabilizing BGP comes on the cost of its convergence time. In other words, Maintaining continuous connectivity to reduce packet loss means increasing BGP stability, which leads to longer convergence time. And reducing the convergence time affects (badly) BGP stability and, in turns, increases packet loss.

In this work we believe in that the connectivity is the main objective of the Internet, connectivity means packet delivery. So having slightly longer convergence time

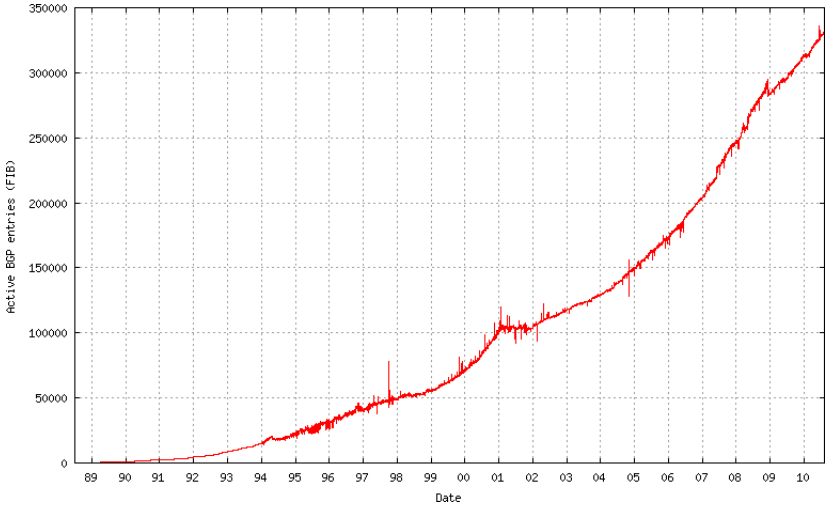


**Figure 2.8:** Forwarding Information Base 1990-2010 (2)

while ensuring continuous connectivity is better than losing traffic for the sake of fast convergence. This motivates us to propose a backup solution for BGP increases its stability and guarantees continuous connectivity but with slightly slower convergence process.

Interdomain events (topology change, policy change, etc) are not the only affecting factor on eBGP convergence time, but it can be affected by the intra-AS events (6). This makes studying BGP stability is attached to its both modes (external and internal). Therefore, Internal anomalies listed in subsection 2.2.6.2 may disrupt the external connectivity, which leads to more events in the external topology. Thus the eBGP convergence process will run more times and then more possibility of interdomain packet loss. This motivates us to propose a new iBGP solution that ensures internal stability and, in turn, improve the global stability.

The main mission of any routing protocol is to build a logical topology that guarantees data connectivity. So any new approach in this domain must hold several properties to be accepted as an applicable and useful protocol. In this dissertation we tackle the following properties, which we think are sufficient to have happy topology:

1. Safety: a safe topology is the topology that is free of loops (routing and forward-

ing).

2. Scalability: scalable protocol is the one that does not highly consume topology resources like: link utilization (with high number of control packets), router's resources (CPU, memory) that nay be caused due to huge number of iBGP sessions, etc.

3. Correctness: correct configuration is the one that reach the topology to the intended stable state, which is the state that respects the designed routing policy.

4. Robustness: is the ability to resist all types of events (topology or policy changes).

5. Backward compatibility: it states that the enhanced protocol should be operable with the conventional one. This property is essential when we deal with a heterogeneous topology like the Internet.

We have employed several tools (rocketfuel, brite, SSFNet) and frameworks (routing algebra, topology modeling) to validate each of the above properties in chapters 4 and 6.

# Chapter 3

# External BGP

## 3.1 Related Works

Today's Internet is the interconnection of more than 30,000 ASes with continuous increase, where the BGP table size has exceeded 300,000 entries and expected to jump over 350,000 by the end of 2010 (2). These numbers have direct impact on the scalability of BGP, since they affect both the number of BGP messages and its convergence time (8). Measurements and experimental studies (6; 14) have shown empirical results on the impact of routing events, either inside or between ASes. BGP is not guaranteed to converge (21), the huger the Internet, the lesser this guarantee is. It was observed that BGP can be behind more than 50% of the performance problems of voice calls because of its slow convergence time (42). Several contributions (8; 19; 39; 41; 42) have agreed on that BGP may take more than 30 minutes to converge where the underlying infrastructure is connected. Several proposals (10; 32; 41; 44) were appeared to illustrate the BGP convergence time problem but none of them talked about the employment of their solution in the real world Internet. Further, they have not mentioned how to fix the backward compatibility problem of their solution. However, the added value of our proposal is the backward compatibility with current BGP. This is reflected on the design of our enhancements to be practical and usable in the current Internet.

Obradovic (45) has proposed a real-time model for BGP with which he could put

an upper bound of the convergence time which is estimated by $O(n)$, where $n$ is the number of AS nodes in the network. Sun *et al.* (62) reduced this bound to the order of the diameter of the network by using the root cause notification. These proposals have tried to clarify the convergence time problem and modeling it, but they have never talked about the disconnectivity caused by BGP re-convergence process, which is the real concern in this domain.

Katabi *et al.* (42) have done measurements on VOIP calls through the Internet. They claimed that most of the previous works have concentrated on documenting the voice performance problem with an assumption of that all VOIP's performance problems are caused by congestion. They claim also that more than of 50% of performance problems are caused by BGP. This shows that BGP is a serious concern for the real time applications.

Katabi *et al.* (41) stated that it is impossible to design an Inter-domain routing protocol that converges fast enough for real-time applications. They have suggested a new protocol R-BGP (Resilient BGP), which is based on the pre-calculation of backup path. The backup path is then to be advertised to the next-hop of the primary one. However, R-BGP has no loop avoidance mechanism, and it does not limit the failure propagation. This latter has a direct impact on BGP convergence time. R-BGP has no flushing mechanism to reorder the paths in Loc-RIB, hence non-optimal paths might exist whereas the optimal ones are existed in Adj-RIB-In.

Pei *et al.* (10) improve the convergence time of BGP by limiting it with the order of the network diameter, as well as reducing the number of BGP messages exchanged during the convergence process. This is achieved by adding to each BGP message an identifier (root-cause) indicating the cause of the BGP message. BGP-RCN (10) is mainly based on loop avoidance to improve the convergence time, but it does not illustrate the main problem which is the transient disconnectivity. Yannuzzi *et al.* (39) have evaluated the proposed solution in (10) and claimed that it suffers from the problem of identifying (accurately) the root cause of a failure. They have presented three objectives of the Inter-domain routing: reducing the number of BGP messages

exchanged during convergence, scalability, and reducing the BGP convergence time. They stated that it is difficult to exceed some of the inter-domain routing limitations due to the worldwide deployment and the autonomousity of the entities through the Internet.

Although, all previous solutions have admitted that it is impossible to reduce the convergence time to zero, most of them concentrated on reducing it and forgot about the disconnectivity caused during this time. We also have tried to reduce the convergence time, but we were more interested in keeping the connectivity during the path recalculation phase. This gives our proposal additional advantages, no matter how much we could reduce the convergence time since we gain the continuous connectivity. To our knowledge, there is no proposal that has tackled BGP convergence time while illustrating the following points:

- Transient disconnectivity elimination.
- Loop avoidance.
- Limiting failure propagation.
- Ensuring intended stable state.

In the next chapter, we show that our proposal meets the above points through different mechanisms: first, we combine the root cause (10) with the backup strategy (41) in our design, second, we send the backup information in a new optional attribute instead of proposing a new protocol, third, we employ selective withdraw mechanism to limit the propagation of failure, forth, we added a new timer TRT (Transient Route Timer) to ensure that the Loc-RIB contains the best path, this is referred to as the intended stable state. The validation of our proposal is in the next chapter.

## 3.2 Stabilizing EBGP

This section presents our approach that illustrates the transient disconnectivity of BGP during its re-convergence process. Although reducing the convergence time is not our

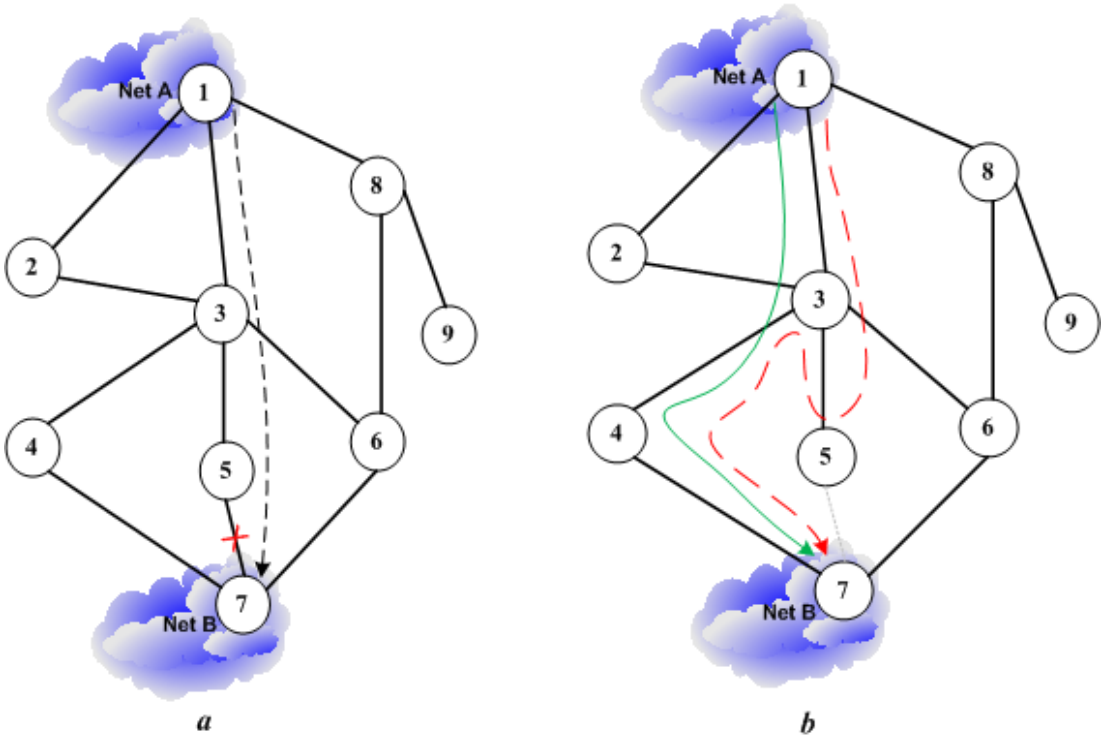


**Figure 3.1:** *a.* traffic between A and B before link failure between ASes 5 and 7
*b.* Dashed line shows the traffic flow in the temporary phase, continuous line shows the flow in the converged state

main objective in this work, but when designing our solution we were aware of keeping that time as short as possible.

Once a topology suffers from a link failure, the time needed to re-converge is divided into two parts: the first one is the failure propagation time, and the second one is the period of the recalculation of alternative path. During this time the topology may suffer from transient disconnectivity, and hence packets are dropped.

In the rest of this chapter we assume that the node we are studying has another possible choice (path) to the lost NLRI (Network Layer Reachability Information), in other words, the underlying infrastructure is connected. If this is not the case, nothing can be done to prevent the disconnectivity (as in figure 3.1.*a*, if the link between AS9 and AS8 is lost, then AS9 can do nothing).

Let us consider the scenario in figure 3.1, where the primary path between the two networks A and B is AS1, AS3, AS5, AS7. If the link between AS5 and AS7 fails, then AS5 has no other choice because AS3 has not informed it about the other two possible paths. Therefore, packets that were passing through AS5 will be dropped until AS3 moves to AS4 or AS6, or until AS1 turns to AS8.

On one hand, BGP timers play an important role in reducing the number of UPDATE messages and eliminating route flapping by creating time intervals between BGP messages. On the other hand, they calm the reactions against failure events which, in turn, slow the re-convergence of BGP. This fact makes reducing the convergence time an inherent trade-off. Griffin *et al.* (19) showed that we cannot expect a net gain in convergence time by reducing the MRAI initial value below the default value (30 second).

In fact, the key problem behind BGP slow convergence is the possible traffic loss during the convergence process, which is referred to as transient disconnectivity. Instead of working on reducing the convergence time and causing undesirable results, we have chosen to admit the slow convergence and tackle its key problem, that is the transient disconnectivity.

Our approach as shown in figure 3.2 states that each node, in the stable state, runs the backup algorithm and builds a list of backup paths ranked according to their disjointness with the primary one. Once a failure occurs, the closest nodes to that failure add the root cause information and disseminate withdraw messages to a subset of their neighbors. Each node that receives a withdraw message select a backup path with respect to the received root cause information and then switch to it while running the TRT timer. Once the TRT expires, the node updates its Loc-RIB by installing the new best route instead of the backup one. This last step ensures the intended stable state because the standard convergence process runs in parallel with our proposed mechanisms. next subsections details the four mechanisms that form our solution.

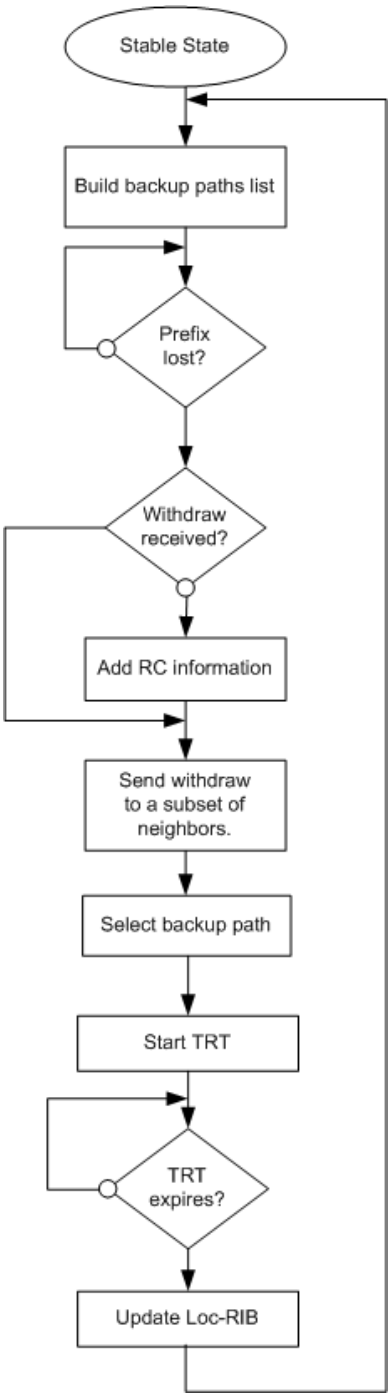


**Figure 3.2:** eBGP proposal

### 3.2.1 Propagating RC Information

This is a loop avoidance mechanism, the node that loses a peer adds its own AS number to the withdraw message, so the receiving nodes avoid any path that passes again by this node. Previous works, like (10; 50), refer to this information as *Root Cause Information*.

The RC info is filled in the BNH/RC (Backup Next Hop/Root Cause) Optional Non-transitive attribute. To clarify this procedure we apply it on the scenario in figure 3.1.

- In the stable state the BPSA (Backup Path Selection Algorithm) is applied to build the backup database for each AS. AS3 receives UPDATE messages containing network B from its neighbors and selects AS5 as its primary next-hop. Upon applying the BPSA, AS3 finds that the most disjoint path from the primary one is via AS4. AS5 receives the advertisements from AS3 and AS7. AS7 is its primary next-hop and cannot be its BNH since this violates condition (1) of BPSA. AS5 cannot also choose AS3 since this violates condition (2). Therefore, AS5 looks for network B in its neighbors' backup information, i.e. the BNH attribute, AS7 has no backup to network B, while AS3 has its backup via AS4. The result is that AS5 installs a backup path to B which is [AS3, AS4, AS7].

- In the Non-Stable state we consider two possible failures:

  - Case I: Failure in the link between AS5 and AS7. Once the failure happens, AS5 clears all the NLRIs learned by AS7. Then, it looks for the BNH of this path which is AS3, after that, AS5 adds itself as an RC info and sends trigger withdrawal to AS3. Finally, AS5 installs the backup info directly into its Loc-RIB and starts sending and forwarding traffic immediately. At this point, the traffic coming from net A to B follows the dashed line in figure 3.1.

  - Case II: Failure between AS3 and AS5. AS3 detects that its peer AS5 (its primary next-hop towards network B) has been lost. It verifies its backup

path and installs the one via AS4 instead of the old one in the Loc-RIB, and stops the failure at this limit till TRT expiration.

RC info provides the Internet nodes with additional information concerning the topology event, which helps in making more stable decisions. However, if all received paths includes the RC node, then the received node has no choice rather than selecting a path that passes by an RC node. A situation like this can be illustrated when we look at the AS node as a set of routers. Therefore, this concern can be resolved if we add the following enforcement:

*RC information is added only by the ASBRs that send withdraw messages to external peers.*

Following this enforcement leads to that the ASBR that loses an external peer should not add the RC info to the UPDATE messages destined to the internal peers. The ASBR that receives a withdrawn route from an internal peer knows (thanks to iBGP) whether the lost prefix is accessible via another ASBR or not. If so, it does not add RC info, else, it adds it.

### 3.2.2 Backup Solution

The backup path selection algorithm is divided into two separate phases: first, building backup paths list during the stable state and rank them according to their disjointness with the primary one, second, selecting backup path upon receiving the withdraw message that contains the root cause information.

Our proposed solution includes enhancements for the network performance in two states:

- Stable State: when there is no failure in the topology, BGP speakers scan their routing table and calculate the best backup paths using BPSA, and then disseminate them through the new attribute BNH to a selected group of their neighbors, as we will see in the selective withdraw mechanism.

- Non-stable State: when a failure takes place, each node faces one of two possible scenarios:
  - It loses one of its peers: it formulates an update message to withdraw all the paths whose next hop is the lost peer. It includes in this update message the RC information through the new attribute BNH/RC. If it has a backup path, then it installs it immediately in its Loc-RIB and runs the decision process to select the new best path.
  - It receives an UPDATE message containing a withdrawn route: it reads the RC info to avoid the selection of any path that passes by the node that originated this message, then it withdraws that route from its neighbors being used it as next-hop to the withdrawn prefix.

The (BPSA) Backup Path Selection Algorithm respects the following conditions:

1. Avoids the backup path whose BNH is the same as the lost primary one.
2. Avoids the backup path which passes again by the same node.
3. Avoids the backup path that contains the RC information.
4. Considers the inter-AS relationships.

It works as follows:

- It looks for an alternative neighbor that can route the traffic to the lost prefix. If it finds a neighbor that can play the role of a BNH and respects the above conditions, then it selects it.
- It fills the backup info in an "Optional Non-transitive" attribute (BNH) and sends it in a BACKUP UPDATE message to its primary next hop. That is because all the other neighbors receive an update message containing its primary path. Therefore, those neighbors are able to select the announced primary path as their backup. An example that justifies this choice is in figure 3.1.a, let AS8 be

the primary next-hop of AS1 to reach network B, and let AS3 its backup next-hop. Both AS2 and AS3 do not need to know about the backup next-hop of AS1. However, AS8 needs this backup information, thereby selects AS1 as BNH, whereas AS2 selects AS1's primary path as backup one.

- If it could not find that neighbor, it searches the advertised neighbors' paths to find the possible backup paths through the value of BNH attribute.
- If more than one possible backup path exist, it selects the most disjoint path, which is the one that has the minimum number of common nodes with the primary one. And if there is more than one disjoint path, then the standard BGP path preference procedure is applied.

### 3.2.3 Limiting Failure Propagation

The objective behind this filter is to limit the propagation of failures in the network, and to stop them at the node that has backup path toward the lost NLRI. The node, which receives a withdraw message and has no backup path for the lost NLRI, disseminates the withdraw message. However, if it has a backup path, we distinguish between two cases:

- Its backup next-hop is its predecessor in the primary path: In this case the withdraw message should be forwarded only to that node. For example, in figure 3.1, AS3 is the backup next-hop of AS5 to reach network B, where AS5 is AS3's primary next-hop to reach the same network. In this case AS5 must send a withdraw message to AS3.
- Its backup next-hop does not relate to the primary path. In this case the withdraw message should be stopped at that node. In figure 3.1, AS3 does not need to propagate the withdraw message received from AS5 since its backup next-hop is AS4.

The node that installs a backup path in its Loc-RIB instead of the primary one should inform its neighbors that are using it as next hop to reach $\rho$. However, and

for the sake of continuous connectivity, we propose that the node will put off such announcement until it converges to the best path. That is because the backup path, as we will see in the next sections, is not necessarily the second best one. This amount of time to wait is related to the TRT timer. Once this the TRT timer expires, it disseminates an UPDATE massage carrying both the withdrawn path (unfeasible) and the backup path (feasible).

This has the following advantages:

- The connectivity is assured during the inter-advertisements interval (MRAI).
- Neighboring nodes receive one update message and execute immediately the phase II of the decision process, which notably reduces their convergence time. That is because this latter will be independent of MRAI timer.
- Increases stability.

### 3.2.4 Transient Route Timer

TRT (Transient Route Timer) is a timer used to ensure the intended stable state. Upon its expiration, it forces the node's BGP instance to run its preference calculation to look for the policy compliant path, and then move it to the Loc-RIB. The motivation behind the employment of this timer is that the backup path is not necessarily the second best path, and since it is installed in the Loc-RIB, this latter need to be refreshed to ensure that it contains the best policy compliant paths towards all prefixes. The initial value of this timer should cover the duration of the transient instability of the topology, through which the convergence process is run.

Recall the example in figure 3.1, AS1 has three paths in its Adj-RIB-In towards network B. We assume that it selects AS3 as it best path to B. Let AS2 be the second best. Upon applying the BPSA it finds that AS8 is the best backup path (most disjoint path). If the link between AS1 and AS3 fails, AS1 moves directly to its backup path, which is via AS8. There is no event that will trigger AS1 to move to AS2, which is

preferred over AS8. To change such unintended state we introduce the TRT to trigger AS1 to move to its BGP policy best route after the loss of the route via AS3.

Failure propagation, using our proposal, looks like a damped wave whose amplitude (amplitude represents the number of affected nodes) decreases when the failure goes farther from the root cause. It starts with its maximum amplitude, because a node failure affects all its neighbors. Once that event reaches a node that has a backup path, it stops it. That is why the amplitude decreases. TRT represents approximately the wave period and acts as a pause before the nodes replace backup paths with the new best ones.

## 3.3 BPSA vs DUAL

One of the algorithms that can be discussed when looking for a solution to stabilize BGP is DUAL (Diffusion Update Algorithm) (12; 18). In addition to the routing table, DUAL uses two other tables: Topology and Neighbor tables. Neighbor table contains all the directly connected routers. Topology table contains all the routing information sent by the neighbors listed in neighbor table. DUAL enforces each router to select a successor and a feasible successor. Successor router is the neighbor that has the minimum calculated cost. Feasible successor is the one that meets the feasibility conditions, which states that the advertised distance of the feasible successor must be less than the calculated distance of the successor. This condition ensures that the backup path (feasible successor) is loop free.

Several aspects can be discussed about DUAL:

- One of the main concerns of DUAL is that the diffusion calculations need a (relatively) long time to be done in a huge topology like the Internet. DUAL adapts quickly to routing changes in medium scale networks (7; 28), but its behavior cannot be expected in very large scale networks like the Internet.

- BGP uses path vector routing and it is not possible to push it to employ DUAL due to that DUAL is not interoperable with path vector routing. Moreover,

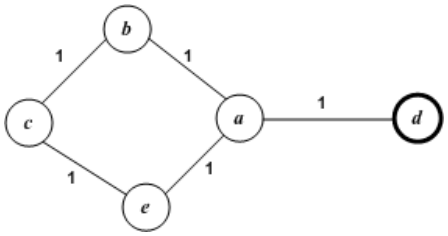


**Figure 3.3:** A simple scenario to compare the feasible successor of DUAL with disjointness backup path

applying DUAL's feasibility condition on BGP metric needs further investigation since BGP has no metric value that can be compared.

- Comparing the feasibility condition with our disjointness condition, we figure out that the disjointness is an easier condition than the feasibility one and at the same time it ensures freeness of loops. Figure 3.3 shows a simple scenario, if we apply DUAL and look from $b$'s stand point, we notice that its successor is node $a$ to reach destination $d$, and it has no valid feasible successor because the advertised distance from $c$ is larger than best distance to $d$. However, if we apply our BPSA we notice that $b$ accepts $c$ as its backup next hop. Therefore, if the link between $a$ and $b$ fails, and DUAL is employed, then node $b$ enters the active state and calculate a new successor. However, and using BPSA, node $b$ moves directly to its backup path.

- Although our approach care about traffic loss more than slow convergence of BGP, but we are still aware that BGP convergence time should not be very long, and our results show a slight increase in the convergence time. However, DUAL will cause BGP to have a very slow convergence time.

No previous works have analyzed DUAL in a large scale network, so a good future work might be to employ DUAL in Internet-like simulation framework and observe its behavior. Then we may judge if it can be used in interdomain routing.

# Chapter 4

# Validation and Analysis

In this section we present the experimental validation, and then we analyze the obtained results. We prove that our proposal ensures the following characteristics:

- Limiting failure propagation.
- Robustness (continuous connectivity).
- Safety (no routing loops).
- Intended stable state.
- Backward Compatibility.

## 4.1 Safety

Our stabilizing solution tackles the transient disconnectivity that follows a failure in the network. It ensures continuous connectivity while the network is converging to its stable state. It depends on backup paths calculated during the stable state. We prove now that those backup paths leads to safe state.

The algebra that describes the backup path selection process depends on two principal parameters carried in two attributes: AS_Path of the primary path and RC informations included in the received withdraw messages. Based on the routing algebra,

we define an algebraic structure that describes how our solution selects backup paths, then we prove that this choice is safe.

The algebra of backup path selection is $(\Pi, \preceq, L, \oplus, \phi)$ where the set of policies $\Pi$ is the standard BGP attributes plus the addition optional attribute that is the inter-AS relationship type. However, we will illustrate only the AS_Path attribute, which plays the principal role in selecting the backup path. Therefore, $\Pi = \{0,1\} \times \mathbb{N} \times \{\pi, c, v, p, \phi\} \times ap$, where $\pi$ is the policy of a trivial path, $ap$ is the path vector carried in the AS_Path attribute, and $c, v, p$ are the relationship types: customer, provider, and peer respectively. The positive integer parameter reflects is the avoidance level of the backup path. Its initial value is zero, it augments when the number of common nodes increase. The 0/1 parameter takes the value 1 if the RC node is found in the advertised AS path and 0 elsewhere.

The set of labels $L = \{c, v, p\} \times \{ap_p\} \times \{rc\}$ where $ap_p$ is the AS path of the primary path and $rc$ is the root cause AS node sent by the withdraw messages. The comparator $\preceq$ is the lexicographic $\leq$. The operation $\oplus$ is defined in the following table.

| $\oplus$ | $(b, d, c, ap)$ | $(b, d, v, ap)$ | $(b, d, p, ap)$ |
|---|---|---|---|
| $(c, ap_p, rc)$ | $(b, d, c, ap)$ | $(b, d, \phi, ap)$ | $(b, d, \phi, ap)$ |
| $(v, ap_p, rc)$ | $(b, d, v, ap)$ | $(b, d, v, ap)$ | $(b, d, v, ap)$ |
| $(p, ap_p, rc)$ | $(b, d, p, ap)$ | $(b, d, \phi, ap)$ | $(b, d, \phi, ap)$ |

$b = 1$ if $rc \in ap$ and 0 elsewhere, and $d$ is the number of common nodes with the primary path. The less these two members are the better the backup path is. The RC information is determined by the sender node and may not be changed by transient nodes, thereby the value of $b$ will not be changed after we apply any label to received path. The same is for $d$, because the number of common nodes is not changed after applying the local label since the current node is not considered in the calculation of $d$. Therefore, the monotonic property of this algebra is guaranteed by the third and fourth members of $\Pi$ as proven in (30).

Note that the operation $\oplus$ states that, for example, extending a provider path by a peer or customer one leads to impossible path $\phi$. However, and since backup paths are used temporarily, we may accept to violate inter-AS relationship guidelines to have more possible backup paths. This case might be useful in the case where all the paths (the respect the AS relationships safety guidelines) contain the RC node. In other words, a *valley* backup path is better than a path that contains the RC node.

## 4.2 Experimental Validation

This section presents the simulation results upon applying our approach on topologies that were built using BRITE (60). Our measurements were done in three phases: First, we collect statistical data from real ISPs (Internet Service Providers), second, we run the simulation using the standard BGP, and finally, we re-run the simulation on the same topologies of phase II using the Enhanced BGP. We start this section by introducing the auxiliary tools we have used: Brite and SSFNet.

### 4.2.1 Brite

Brite (60) is a topology model, its goal is to produce a topology generation framework which improves the state of the art and is based on design principles which include representativeness, inclusiveness, and interoperability. Representativeness leads to synthetic topologies that accurately reflect many aspects of the actual Internet topology (e.g. hierarchical structure, degree distribution, etc.). Inclusiveness combines the strengths of as many generation models as possible in a single generation tool. Interoperability provides interfaces to widely-used simulation applications such as ns, SSFNet and OmNet++ as well as visualization applications.

While rocketfuel was used for intra-AS topology model, brite is used as topology generator for inter-AS topologies. The generated topologies were introduced to SSFNet where we apply our proposed eBGP enhancements.

### 4.2.2 SSFNet

SSFNet (3) is a simulation environment that is capable of running network routing protocols on imported topologies. This open source tool allows to modify the implementation network protocols, which makes it suitable for us since we need to modify the BGP implementation. We have used SSFNet simulator to apply the eBGP backup solution on topologies generated by Brite tool. It is divided into two levels. The core level includes protocol implementations using java language. The second level is the configuration one, where the network topologies are designed and the protocol models are selected using DML (Domain Modeling Language). The implemented Models are: NIC (link layer implementation), IP, TCP, UDP, OSPF, BGP, HTTP, sockets, and NetFlow collectors.

#### 4.2.2.1 Points of Interests

Simulation in SSFNet starts by the configuration phase, in which we design the desired topology. The problem we have faced in SSFNet is that it is not an interactive tool that allows inserting changes in order to do the needed measurements to calculate the convergence time. Thus, we have had to introduce additional DML functions to fit our needs.

SSFNet was run on topologies generated by brite (60) tool. We then inserted topology changes (addition and deletion) to trigger BGP convergence process and measures its delay.

Setting up the simulation environment starts by converting SSFNet into a real-time interactive tool, the second task was to insert measurement functions to give us the value of the convergence time, and finally we implement our enhancements and integrate them into the BGP implementation.

#### 4.2.2.2 Detailed Modifications

- Calibrating the simulation time: This step is necessary to give us a correct estimation time. We have built a new class (SSF.OS.RTSim.class) that converts the original

simulation ticks into the real-time ones, all our DML files refer to this class in order to have real-time simulation results.

- Adding events: We add additional DML functions to be parsed by the simulator. The role of them is to cause link failures or create new links in the illustrated topology.

- Java threads: Processing the new DML functions needs additional threads, methods, and classes to be built in the core implementation of SSFNet, as well as new functions in all the classes that implements TCP/IP layers reaching to the BGP class.

- Change listener: SSFNet uses change listener classes to get notifications upon any changes in the routing table, we added our own listeners to get notifications upon link failures, we also create new functions in the existed listeners to run the backup process upon the changes in the routing table.

- BGP class: We integrate our own functions in this class, the stable state enhancements were added in the main function, where the non-stable state ones were added through a separate thread.

- BPSA: The implementation of the backup path selection algorithm was added in the BGP class.

- RC Information: The fulfillment of the RC attribute takes place in the BGP class by the node that receives the failure notification.

- New attribute: Our new attribute is added to the BGP message header in the RouterInfo class and marked as Optional non-transitive ones.

- Backup path: The backup next hop carried in the BNH attribute is saved in the routing table, so we added this entry in the Route class called BNH.

- BGP path: Current SSFNet BGP metric calculation (dop function) depends on one attribute only which is the LOCAL_PREF, we modified this function to take into consideration the AS_Path as well as BNH attribute when we have two equal local-preference paths.

#### 4.2.2.3 Why SSFNet?

We have chosen SSFNet (3) as a test environment for our solution. SSFNet has a good implementation for BGP and provides two layers (Java and DML) for building the needed topologies. We were compelled to add new modules to SSFNet since it does not support all what we need to validate our solution. We applied our solution to topologies of up to 10000 nodes, and due to a software limitation we could not simulate the real number of ASes in the Internet which is more than 30000 nodes. The choice of simulation as a mean to validate an approach for an inter-domain routing protocol like BGP might seem critical. The simulation's added values over experimentation are:

- We need to modify the BGP implementation, however, commercial routers do not give the ability to access its built in operating systems, and hence there is no way to have Internet nodes that fit our needs.
- Setting up a special (practical) environment to test our enhancements will be limited in both the number of nodes and the number of links.
- Scanning previous works that depended on experimental validations, we notice that researchers were limited to the environment they test, and they have no ability to generate the needed events to test the topology reaction. Simulation helps in testing complex topologies and generating all types of events.

Experimentation is still more convincing validation method than simulation, and we are not trying to change this. However we argue that simulation is an important step before moving to the experimental one.

### 4.2.3 Data Collection Phase

This phase includes statistical information collected from two running ISPs. Three objectives are behind this phase: first, putting hands on the divergence problem that Internet nodes (ASes) are suffering from, second, collecting information about the most frequent events in the Internet, third, building practical evaluation of possible solutions.

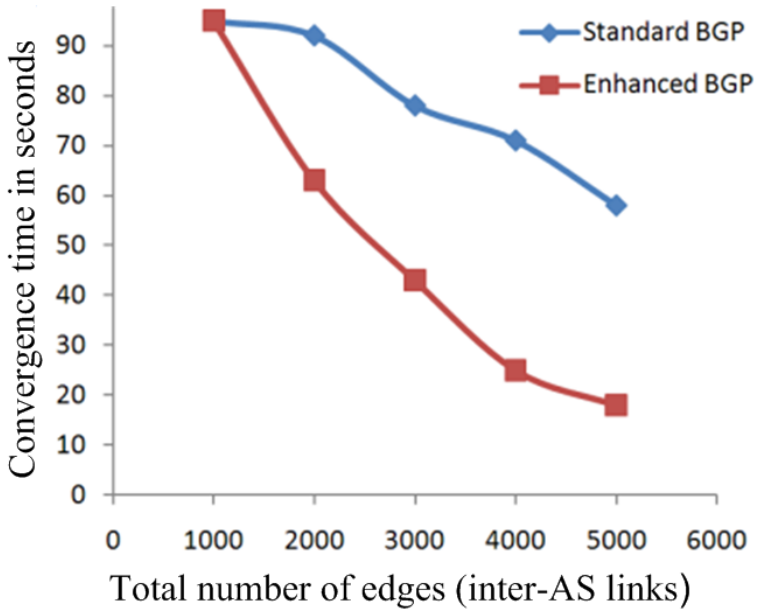


**Figure 4.1:** Convergence Time in Seconds as a function of topology's total degree

The results of this phase are just a minimized version of those announced in the CIDR Report (2). But the added value was the recommendations we have got from the ISPs' network administrators.

We found that the withdraw messages are about 30% of the total received events, while the announcements form 70%. Those events cover nearly all the prefixes in the routing table in all the days except weekends. In fact, the variation was not only from one day to another but also during the same day. The huge value of the standard deviation in results was justified by the network administrators to be due to some maintenance operations that took place periodically. The collected statistics were through several days. We have not mentioned the exact numbers because they are changing daily and the relative ratios are more interesting for us. One of the most interesting watches was the disappearance of some prefixes in the routing table. The percentage of those prefixes was nearly 1% of prefixes in the routing table. This reflects the changing nature of the Internet as well as the need to illustrate ite frequent events and the divergence caused by those events.

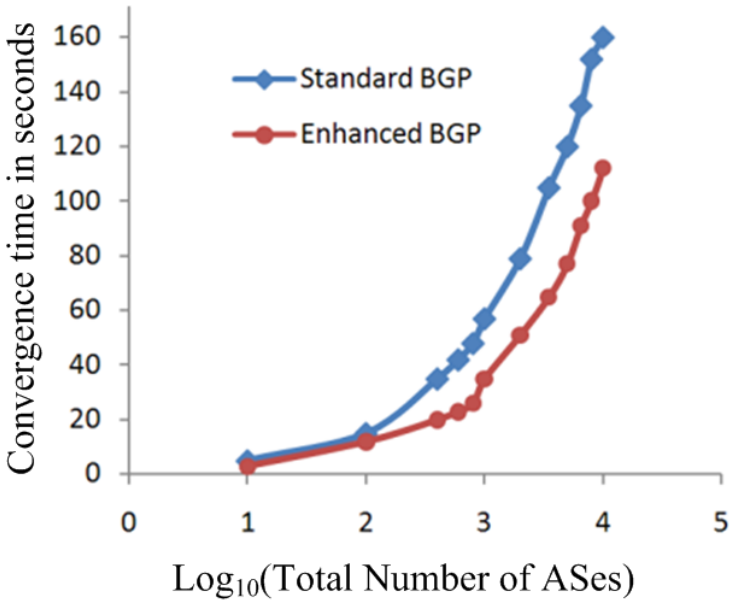


**Figure 4.2:** Convergence Time in Seconds as a function of decimal logarithmic values of the number of nodes (ASes)

### 4.2.4 Standard BGP

We have performed measurements on simulated topologies without any enhancements in BGP implementation. We have measured the convergence time according to its affecting parameters (number of nodes, number of edges, timers, etc). The convergence time we have measured is the global convergence time, that is the time interval between the instance of failure event and the instance where all the nodes converge to their stable state.

We have done the following experiments and compare them with those in phase III:

*Experiment 1:*

We run the simulation for 12 different topologies with different number of ASes. For each topology we apply different events including link/node failures and calculate the arithmetic mean value of the convergence time. The results are shown in figure 4.2.

*Experiment 2:*

We run the simulation for different topologies with different total degrees but with the

same number of nodes N. The results are shown in figure 4.1.

*Experiment 3:*

The objective of this experiment is to calculate the number of affected nodes by failure events. The calculated number reflects the failure propagation in the topology. For each topology with fix number of nodes N, we applied several events and counted the affected nodes as in figure 4.3. The node is considered to be affected if it receives a withdraw message concerning the disconnected prefix.

### 4.2.5 Enhanced BGP

We have used the same topologies of the phase II and we have done the same experiments while using the enhanced implementation of BGP.

*Experiment 1:*

For each tested topology in phase II, we applied the same events and measured the convergence time. Figure 4.2 shows the convergence time as a function of the number of nodes, where each value of the convergence time reflects the arithmetic mean value of several tests. During this experiment we made sure of the continuous flow of the traffic between the two end points.

*Experiment 2:*

We fixed the number of nodes to 1000 and changed the number of total degrees in the network, and then applied the same events of phase II. We compared our results with those of phase II as shown in figure 4.1.

*Experiment 3:*

We have used our enhanced version of BGP and apply the same events on the same tested topologies used in experiment 3 in Phase II. The results are shown in figure 4.3.

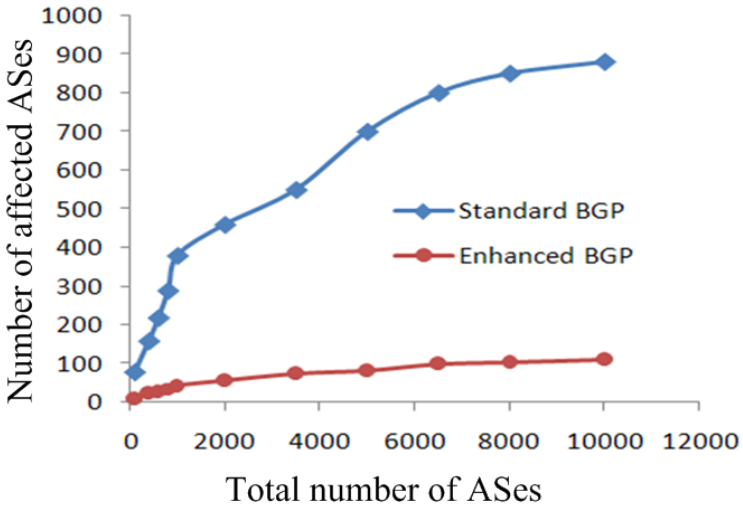


**Figure 4.3:** Number of affected ASes as a function of the total number of ASes in the topology

## 4.3 Analysis

### 4.3.1 Convergence Time

Many previous works tried to decrease the BGP convergence time, few of them talked about the continuous connectivity. In our work we believe that it is not possible to make the Internet as a fast convergence topology since this has bad effects on its stability. This directed our interest to both: finding a solution for traffic routing while the topology is converging, and reducing the convergence time as possible without affecting the stability of the Internet. However, and due to the trade-off between fast convergence and traffic loss, we have chosen to resolve the traffic loss without increasing the convergence time tremendously. We first need to remind that the convergence time we are measuring is the global convergence time, which is the time interval between the failure event and the instance where all the nodes are stable. This is an impossible measurement in the real Internet, but it is possible in the simulation environment.

The latest CIDR Report (2) has stated that most of Internet nodes have a 0 count of the stable paths during a period of one month. Our statistics in phase I showed also

that about 1% of prefixes in the routing tables were disappearing for periods ranging from several seconds to several minutes. We conclude from these statistics that BGP convergence process may affect all prefixes in the routing table of all the nodes in the Internet. This shows the severity of the convergence time problem and the need to protect Internet data from loss. Therefore, our work has come not only to reduce the time of disappearance of some prefixes from the routing table, but also to allow the continuous connectivity since most of Internet nodes have always another possible path.

Figure 4.2 shows a slight enhancement in the convergence time. Our experiments also ensure zero data loss, even though our trace route results have given non-policy compliant paths. The slight enhancements in the convergence time came from that the TRT timer was not considered in those experiments. Therefore, the measured time does not reflect the real convergence time since the topology is not in its intended stable state. Taking TRT into consideration will increase slightly the convergence time but will ensure the intended stable state.

Running the same experiment using standard BGP implementation gives transient disconnectivity during a larger interval. This led to accept non-policy compliant paths during the TRT interval because it is still better than the traffic loss. One might think of Internet instability due to the existence of such paths, our answer to this concern is first, those paths are not advertised to the rest of the network and they are used locally to bypass the link failure, second, those paths are used for relatively short period of time (TRT).

The difference in the values of convergence time shown in figure 4.2 is due to limiting the failure event within a failure boundary. Figure 4.1 reflects the impact of the number of links on the convergence time. It shows a bigger difference than that appeared in figure 4.2. That is because the number of links affects directly the probability of having an alternative path. Furthermore, more links means more nodes that have backup path.

### 4.3.2 Failure Propagation

Failure propagation is expressed by the number of nodes that hear about the failure. We consider that a node has information about a failure if it receives a withdraw message concerning the lost link/node. Our approach states that a node should not forward a withdraw message if it has a backup one, which draws a boundary of the failure. This gives that the number of affected nodes using our enhancements is, to some extent, independent from the total number of nodes. Figure 4.3 shows that using the standard BGP implementation causes a spread of failure that reach to about 10% of the nodes in the topology. It shows also that the graph of the standard BGP is linear in its best case, where the graph of the enhanced BGP seems to be saturated at a value less than 200 ASes. This means that, in the current Internet that has more than 30000 ASes, certain failures might disrupt 3000 AS around the world. Whereas using the enhanced implementation limits the number of affected nodes to 200. Thereby, we can guarantee, in a topology of 30000 ASes like the Internet, that the number of affected ASes is bounded by 300 as a rough estimate.

### 4.3.3 BNH/RC Attribute

BGP may not send an update message of a certain NLRI to its selected next-hop to reach the same NLRI. However, for each prefix, we propose that each node sends a backup update message, if it has backup path towards that prefix, to its primary next-hop to inform it about that backup path. This backup information is carried in a new optional non-transitive attribute called BNH.

This choice ensures the interoperability of our solution with conventional BGP, and helps in carrying the needed information. Adding new optional attribute to BGP was already validated in (56) and it was shown to resolve the problem of interoperability. This new attribute either contains the backup next hop address when carried in an update message containing a new NLRI, or it contains the root cause information when carried in an update message containing a withdrawn route.

## 4.4 Discussion

Slow convergence is a well known concern of BGP. On one hand, making this process faster may result in unwanted behaviors. On the other hand, there is no reason for losing traffic since many studies showed that in most cases the underlying infrastructure is connected during the convergence process. This motivates us to turn around the slow convergence problem and propose a solution to overcome the traffic loss problem.

Besides solving the disconnectivity problem, we have tackled two another important issues: failure propagation and intended stable state. Our results showed that the number of affected ASes has been reduced when applying our approach, which decreases the number of nodes that run their decision process and, in turn, increases topology's stability.

Since the backup path is not necessarily the second best route, we have employed an additional timer (TRT) to fetch the new best path and install it in the Loc-RIB instead of the backup one. This ensures that the Loc-RIB contains best paths, which is the intended stable state.

However, TRT slows the convergence process, because once a node move from its backup path to its best one it diffuses it. This enters the topology in a converging state. Several ideas might be employed to have smooth transition from the backup to the best path depending on the provided features from the existing routers's operation systems.

Load balancing feature is almost existed in all nowadays routers. Balancing traffic between the backup path and the new best route makes the transition between them smooth and allows the backup path to be used while the network is converging. This can be reflected on previous figures in section 4.2, where they show results in the absence of TRT. Therefore, employing load balancing technique will probably make them accurate even in the existence of TRT.

Figure 4.3 brings us back to DUAL. With such a (relatively) small number of affected nodes, DUAL may behave well and converge quickly.

# Chapter 5

# Internal BGP

Chapter 3 presented our approach to stabilize eBGP using several mechanisms. At that chapter we have considered ASes as simple nodes, which is not the real case. In fact, ASes might be complex networks or routers, moreover, instability internal events may disrupt the global stability as claimed by (6). This motivated us to illustrate internal stability and study its affecting factors.

Being under a single administration makes things less challenging than being co-operated with thousands of other nodes. Therefore, employing a stabilizing solution for iBGP is easier that eBGP. Although both eBGP and iBGP have the same protocol structure but they differ in their behaviors. For example, iBGP can establishes BGP session with an internal neighbor that is not directly connected to it, while eBGP may not (unless it uses IP tunnels). Since iBGP runs in a smaller network than eBGP, this makes its convergence faster unless there is a permanent loop. So the issue we have illustrated in eBGP, that is the packet loss during slow convergence is not a severe concern for iBGP.

However, iBGP anomalies are due to misconfiguration inside the AS as reported by several works. Three main methods are existed for the intra-AS configuration. All of them generate two distinct data and control planes. We argue that this distinction is behind most of iBGP anomalies and propose a solution that gives similar data and control planes. We validate our approach in the next chapter.

## 5.1 Related Works

Both alternatives, route reflection and confederation solve the scaling problem of full mesh iBGP (13). RRS changes the behavior of iBGP by allowing the reflectors to redistribute internally learned routes. Confederation changes the behavior of eBGP by employing it between sub-ASes of the same AS.

Hares and White (54) is among the very few works that employs AS confederation as intra-AS configuration. They used it to heal the AS split problem caused by some types of internal failures.

RRS anomalies and route deflections were reported by several studies (4; 5; 13; 20; 34; 37). The main reason behind those anomalies is that RRS (57) has not defined neither a constructing method to divide the AS into clusters, nor a selection procedure of which nodes should play the role of reflectors and which ones play the role of clients. Thus many proposals appeared to either define a constructing method or check the correctness of running configuration.

The checking works proposed methods and procedures to check the correctness of iBGP configuration. Griffin and Wilfong (20) showed that the determination of iBGP configuration correctness is NP-hard, but they proposed sufficient conditions to ensure the signaling and forwarding correctness. Buob et al (37) presented a methodology to check the deflection that may occur between an internal router and its exit points.

Some of the constructing works suggested major modifications to the RRS while others gave tuning enhancements. Musunuri and Cobb (40) inserted two new modifications to RRS: first, each RR (Route Reflector) sends multiple feasible routes to its peers instead of sending its best path. Second, each RR saves a preference list for each of its clients, each list contains all ASBRs sorted according to their distance from the client. Their solution, S-iBGP (Stable iBGP), is difficult to employ because the preference list has concerns related to performance and scalability in large topologies. Rawat and Shayman (5) gave a list of constraints, when respected, the resulted configuration will be free of forwarding loops and persistent routing oscillations. The determination

of correctness of the proposed configuration is NP-hard according to (20). Moreover their approach does not improve the number of iBGP sessions.

Vutukuru et al (36) proposed the BGPSep algorithm which is based on graph separation method. BGPSep is a pure graph theory algorithm, neither the IGP costs nor the state of the internal nodes (border or internal) are taken into consideration when building the separators. It has the same order of the number of needed iBGP sessions like FMS which is $O(n^2)$ where $n$ is the number of ASBRs.

Xiao and Nahrstedt (35) introduced a new principle of iBGP reliability based on TCP (Transport Control Protocol). They state that the exponential backoff manner of TCP retransmission and the large retransmit interval are behind the fragility of iBGP against network failures. They proposed a reliability model for iBGP sessions that suggests relations between failure probability and BGP timers as well as TCP retransmission behavior. This model helps in both determining the optimal value of BGP timers, and finding a better TCP retransmission mechanism. Further, they found that applying this model may result in making RRS more reliable than FMS.

## 5.2 Motivations and Objectives

Intra-AS divergence has direct impact on global instability. Current iBGP configurations give no optimistic results concerning the stability inside the autonomous systems. This motivates us to look for an alternative intra-AS configuration that ensures stability and robustness.

In AS confederation a special version of eBGP (eiBGP) inside huge ASes may create concerns related to slow convergence inside the AS. This depends on how the internal topology is divided. Choosing clusters in RRs, and sub-ASes in confederation are two critical problems in both proposals. Throughout the different solutions proposed to enhance iBGP configuration and according to many studies, we agreed on that the reported iBGP anomalies and divergence problems come from misconfiguration inside the AS, especially when dividing the AS.

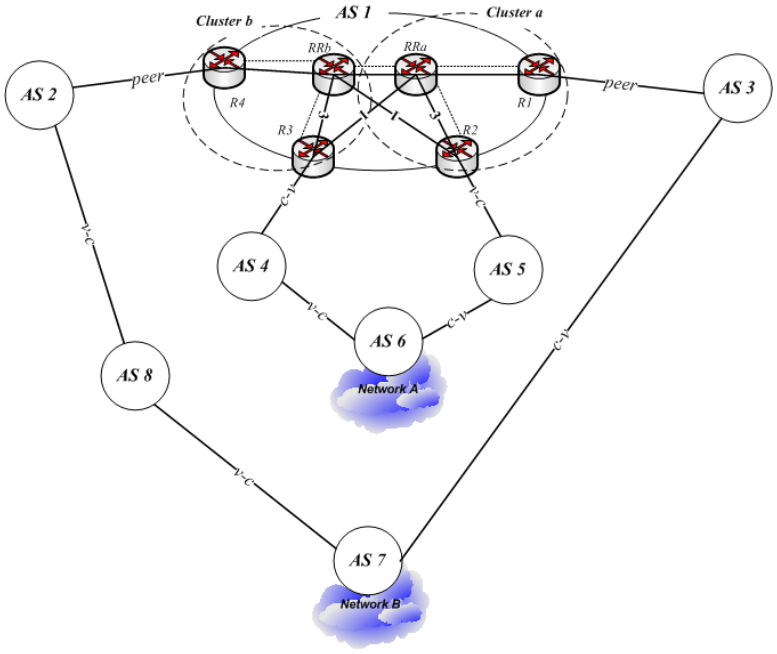


**Figure 5.1:** Inter-AS oscillation caused by internal oscillation

Figure 5.1 shows how an internal divergence causes inter-AS instability. AS7 has two paths to reach network A, AS3 is its preferred next hop (shortest AS path). The route reflector *RRa* selects R2 as its preferred exit router to reach network A. The resulted path between networks B and A is: AS7-AS3-AS1(R1-RRa-R2)-AS5-AS6. However, RRa receives un update from its peer route reflector *RRb* indicating that its preferred exit router to reach network A is R3. R3 has lower IGP metric than R2, so RRa moves to R3 and withdraws the ancient path. This withdraw message reaches to AS3 and in turn to AS7, this latter moves to AS8. Two possible events will reach to AS7 to get back to AS3: First, RRb receives an update from RRa concerning network A, so it moves to R2 as its preferred exit router. Thus, it withdraws the ancient path from R4 which in turn will send it to AS2 and so on till it reaches AS7. Second, once RRb withdraws its ancient path, RRa will be among the receivers of this withdraw so it will get back to R2 and withdraws the path via R3 and AS4. In both cases AS7 will get back to AS3. Since, the oscillation between RRa and RRb is permanent, AS7 will

oscillate between AS3 and AS8 permanently.

Our main objective is to give a correct and employable configuration without the need of changing routers or their operating systems. The proposed solution is not an enhancement to neither RRS nor confederation, it is a third alternative of full mesh iBGP. This approach takes the internal graph as input and gives the Skeleton graph as an output. We mean by the internal or IGP graph the representation of routers and links inside the AS, where each router represents a node and each physical link (or multiple links) between two routers represents an edge in that graph. Skeleton does not divide the AS neither into clusters nor into sub-ASes. It might be considered closer to RRs than to confederation since its nodes are able of reflecting routes.

To meet these objectives we show that Skeleton satisfies the sufficient correctness conditions provided by (20). Safety and robustness characteristics are ensured using the PBR algebra. The scalability of our approach is guaranteed due to that each Skeleton iBGP session is established over single physical link, as detailed in chapter 6.

## 5.3 BGP Skeleton

The basic idea of Skeleton (51) can be summarized in three points:

- The calculation of the best path between each internal node and each egress one.
- Skeleton iBGP sessions are established between IGP neighbors only. In other words, iBGP sessions are established over one hop physical links only.
- All Skeleton nodes are capable of forwarding (reflecting) internally learned routes.

We present in figure 5.2 a simple topology, dash lines represent iBGP sessions and continuous lines represent physical links, with IGP costs next to them. In the first case where RRS is employed, node $b$ is a route reflector and $a, c, d$ are its clients. P,Q are two paths for the same destination. $b$ selects the path advertised by $a$ that has a smaller IGP cost than $d$. $c$ receives routing information only from its reflector $b$, so it will never hear about the path advertised by $d$ which is its preferred egress point. In

the second case, where the Skeleton is employed, $b, c$ have the necessary iBGP sessions to get all the routes from $a, d$. Moreover, they are able to reflect routes to each others, hence, $a, d$ can hear the routes from each others. Thus all the nodes are able to quit the AS using the optimal path.

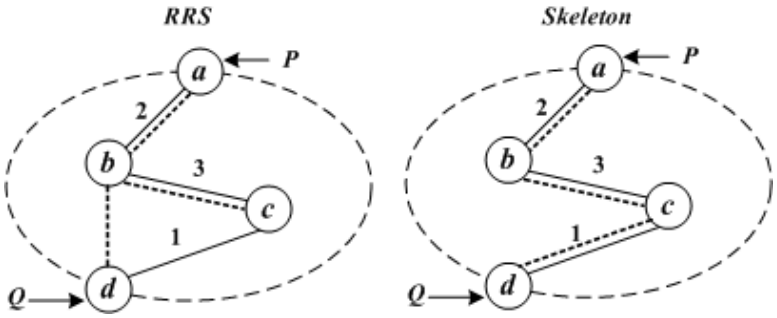


**Figure 5.2:** RRS vs Skeleton

The objective of our approach is to give an alternative iBGP configuration that ensures scalability, stability, and robustness. It is based on building a subgraph (Skeleton) depending on the required iBGP sessions between the nodes. An iBGP session is required between each node and the optimal next hop nodes towards all the ASBRs. This method ensures that each node receives all the advertised prefixes from the optimal next hop to each ASBR, and hence all the nodes in the AS are able to determine their best exit point.

FMS (Full mesh solution) is an optimal solution, but it does not scale for large networks. RRS is known to have problems like route oscillation and forwarding loops, most of these problems are related to that the iBGP session is over a multiple IGP links. AS confederation employs eBGP inside the AS which may impose slow convergence inside large ISPs (Internet Service Providers). Our approach differs from these solutions in the following points:

- It does not divide the AS into clusters or Sub-ASes.
- All the internal nodes inside the AS are Skeleton nodes. And all of them are capable of reflecting routes.

- Nodes are not classified as reflector and clients nodes.
- Besides the IGP routing protocol, iBGP is the only routing protocol that runs inside the AS to diffuse external prefixes.
- Skeleton edges set is a subset of the set of total edges.

We denote C-iBGP (Complete iBGP) the configuration in which each node establishes iBGP sessions with all its IGP neighbors. Physical and C-iBGP graphs are identical. Skeleton is a subgraph of the C-iBGP graph, its nodes are all the AS nodes, and its edges are a subset of C-iBGP. Each edge is an iBGP session carried over an IGP link between two Skeleton nodes. The main drawback of C-iBGP is that it amplifies the number of routing updates, which causes churns of updates that have negative impact on the performance of the network. C-iBGP diffuses the received Internet prefixes to all its directly connected neighbors, and then each neighbor does the same. Regarding the huge number of Internet prefixes, the act of re-diffusing routes inside the AS by each node to all its neighbors has severe effect on the performance of the internal routers. In addition, this increases the link utilization by the control traffic. To fix this performance issue, we propose the Skeleton algorithm to build a partial complete iBGP.

### 5.3.1 Skeleton Algorithm

**INPUT**: Set of Nodes ($V$)$\wedge$ Set of ASBRs ($\Gamma$) $\wedge$ IGP graph $G(V, E)$

**OUTPUT**: Skeleton Subgraph $G_S(V, E_S)$

$E_S \longleftarrow \phi$

**for** each $n \in V$ **do**

  **for** each $\alpha \in \Gamma$ **do**

    $s \longleftarrow \beta_n(\alpha)$

    /* $\beta_n(\alpha)$: *gives the best IGP next hop to reach the ASBR* $\alpha$ from $n$ * /

    $IBGP(n, s)$

    /* *IBGP: is a function that establishes an iBGP session between* $n$ *and* $s$ * /

$E_S \longleftarrow E_S \cup \{n \overset{iBGP}{\longleftrightarrow} s\}$

/* *the new edge is added to the Skeleton subgraph* * /

**end for**

**end for**

Our algorithm takes the physical graph as an input and apply the following steps on each internal node $n \in V$:

- Calculates, for all the internal nodes, the best IGP path towards each ASBR $\alpha \in \Gamma$.
- Extract the next hop of the calculated path.
- Establish an iBGP session with the extracted next hop.

This algorithm gives the internal nodes two advantages:

- Each node gets advertisements for each prefix $\rho$ from all the advertising ASBRs $\gamma_\rho$, which allows it to choose the best egress node.
- It gets the advertisements from its preferred next hop, which allows it to optimally reach its optimal exit point.

### 5.3.2 Skeleton Session Types

We define three types of relationships between Skeleton nodes:

- **Successor**: In each path $P_{n_0,\alpha_j} = n_0 n_1 \cdots n_{m-1} n_m$ $(n_m = \alpha_j)$, we consider that $n_{i+1}$ is a successor to $n_i$ and we write: $n_{i+1} = successor_{\alpha_j}(n_i)$ which means that $n_{i+1}$ is the preferred next hop for $n_i$ to reach $\alpha_j$. The successor reflects all its routes to its predecessor.
- **Predecessor**: We consider that $n_i$ is a predecessor to $n_{i+1}$ and we write: $n_i = predecessor_{\alpha_j}(n_{i+1})$. The predecessor reflects its routes and routes of its predecessor to its successor.

- **Peer**: For two or more intersected paths in $l > 1$ common nodes. If there exists $(n, m) \in P_{i,\alpha_i} \cap P_{j,\alpha_j}$ so that: $n = successor_{\alpha_i}(m) \wedge m = successor_{\alpha_j}(n)$. In this case we denote: $n = peer(m)$ which is equivalent to $m = peer(n)$. The two peers exchange all their routes.

Figure 5.3 shows a simple topology of 8 nodes, 4 ASBRs {a,b,c,d} and 4 internal nodes {*i,j,k,l*}. We illustrate only two nodes $k$ and $l$. The best four paths for node $k$ to reach the egress nodes a,b,c,d are {a}, {b}, {*j*,c}, {*l*,d} respectively. Thus $k$'s best next hops are a,b,*j*, $l$, this leads that iBGP sessions should be established between $k$ and its best next hops. We can write $k = predecessor_a(a) = predecessor_b(b) = predecessor_c(j) = predecessor_d(l)$. The four paths for node $l$ to reach a,b,c,d are {a}, {*k*, b}, {*k*, *j*, c}, {d} respectively. Thus $l$'s best next hops are a, *k*, d, so iBGP sessions should be established between $l$ and its best next hops, and we can write: $l = predecessor_a(a) = predecessor_{b,c}(k) = predecessor_d(d)$. This is equal to that: $a = successor_a(l), k = successor_{b,c}(l), d = successor_d(l)$.

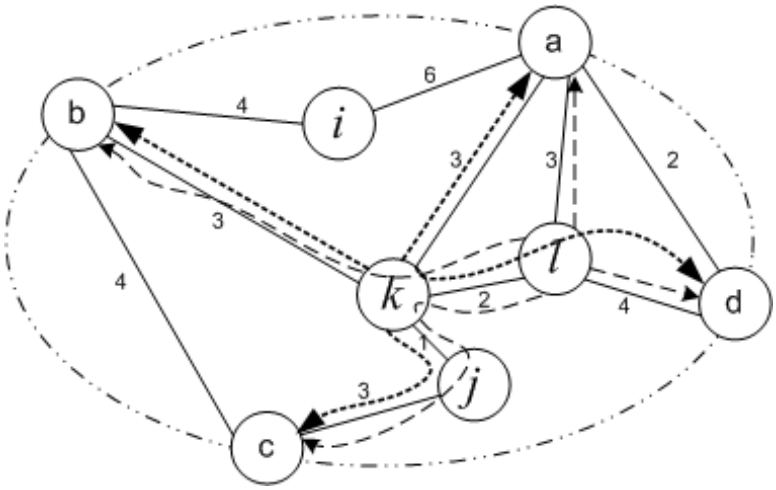


**Figure 5.3:** Example: Skeleton Graph

Node $k$ is the successor of $l$ to reach b and c, and $l$ is $k$'s successor to reach d. Thus $k = peer(l)$. In this particular example, the number of iBGP sessions reaches its maximum value of 10 sessions, which is the number of IGP (physical) links. While in

FMS it is equal to: $\frac{e(e-1)}{2} + ne = 22$ where $e$ is the number of ASBRs and $n$ is the number of internal nodes.

### 5.3.3 Comparisons

#### 5.3.3.1 Skeleton vs Confederation

Skeleton may look similar to AS confederation if we choose that each sub-AS is a single router. However they differ from each other in two main points:

1. The intra AS protocol is iBGP in Skeleton configuration, but it is a special type of eBGP (eiBGP) in confederation.

2. Sub-ASes are connected (BGP peering) to all their neighbors, which makes it like C-BGP but not like Skeleton.

#### 5.3.3.2 Skeleton vs RRS

In a topology that implements RRS, if we make each cluster of a single router, then will this configuration be like Skeleton?. According to the definition of route reflection configuration (57), each cluster must contain at least one route reflector. This makes all internal routers are route reflectors, which leads to that they should be fully meshed according to (57). This is not the case in Skeleton, because internal nodes may not be fully meshed.

### 5.3.4 Implementation

RRS is supported by all routers' operating systems. This makes it mandatory to find mapping relationship between Skeleton and RRS in such a way that makes the implementation of our approach feasible for the network operators.

In each path $P_{n_0,\alpha_i} = n_0 n_1 \cdots n_{m-1} n_m$ between $n_0$ and an ASBR $n_m = \alpha_i$, we configure $\alpha_i$ as a route reflector with $n_{m-1}$ is one of its clients. Each node in each Skeleton path plays a role of client with its successor and a reflector with its predecessor. Thus it receives the clients' routes, then it reflects them with its own ones to its successor.

The latter does the same till we reach an ASBR. In the opposite direction, each ASBR exports all its routes to all its predecessors, then each one of them reflects these routes to all its predecessors till we reach the end nodes. This guarantees that each ASBR diffuses all its eBGP routes to the other nodes of the AS, and all the ASBRs know about all the prefixes inside the AS.

$u = successor_\alpha(v) \Leftrightarrow v = predecessor_\alpha(u)$ is equal to that $v$ is a client to the route reflector $u$ in the RRS notion. Thus the successor relationship resembles the client-reflector one in RRS, the predecessor relationship resembles reflector-client one in RRS, and the peering is equivalent to reflector-reflector relationship in RRS. Using this method ensures the employment nature of our model using the same operating systems in the existed routers.

Skeleton is implemented as a module in OpenNMS (1). This module discovers the topology and build the input file in a *cch* format (43)[1]. Skeleton algorithm is then applied on the *cch* file and gives a list of pairs of nodes. Each pair represents two nodes between which there should be an iBGP session. The last step is the configuration of routers using the mapping with RRS.

[1]In our case, we did not use opennms to generate the cch files, because we either got them from the Rocketfuel project, or generating them by applying the Rocketfuel method to discover real topologies.

# Chapter 6

# Validation

Skeleton validation is divided into three parts:

- Scalability: This validation phase is done by applying our approach on some large real ASes and compare the number of iBGP sessions with both FMS and one of RRS proposals.
- Safety and Robustness: We use our proposed PBR algebra to prove that Skeleton ensures these two characteristics.
- Correct configuration: This is an additional validation step that put the previous two validation methods together. In this validation part, we show that Skeleton configuration satisfies the sufficient correctness conditions proposed in (17; 20).

## 6.1 Scalability Evaluation

We evaluate the performance of our algorithm on five large real-world topologies. Three topologies were taken from the Rocketfuel project (43), and the other two are discovered using the rocketfuel method. The ISP backbone topologies are summarized in Table 6.1 [1].

[1] The presented numbers are approximate values, this does not affect the results since the objective is to apply our approach on a real-like topologies.

| Name | ASN | Internal Routers | ASBRs $a$ | Links | $Nb_{max}$ | $Nb_{avg}$ |
|---|---|---|---|---|---|---|
| Tiscali | 3257 | 276 | 161 | 400 | 13 | 4 |
| Ebone | 1755 | 163 | 87 | 300 | 16 | 8 |
| TiNet | 9121 | 325 | 125 | 525 | 15 | 3 |
| Opentransit | 5511 | 355 | 134 | 600 | 12 | 4 |
| Abovenet | 6461 | 367 | 138 | 1000 | 25 | 7 |

**Table 6.1:** ISP Backbones

Rocketfuel (43) is an ISP (Internet Service Provider) topology mapping engine. It uses trace route technique to get accurate ISP topology. It is an alternative mechanism to the brute force one, which requires a large number of traces to produce an accurate topology. Traceroutes were sourced from 800 vantage points hosted by nearly 300 traceroute web servers. It started by mapping 10 ISPs in Europe, Australia and the United States. In that process, authors constructed a database of over 50 thousand IP addresses representing 45 thousand routers in 537 POPs connected by 80 thousand links. Rocketfuel uses an innovative alias resolution technique to find which IP addresses represent interfaces on the same router.

We have used the produced topologies to apply our skeleton approach on them. Rocketfuel's topologies have been used for scalability validation of skeleton configuration. Skeleton configuration was applied and then we calculated the number of iBGP sessions and compare the resulted number with both full mesh and one of the route reflector solutions (41).

In the collected test beds, we were interested in additional measurements that help in demonstrating our approach. In table 6.1, we introduce the average $Nb_{avg}$ and maximum $Nb_{max}$ number of nodes' neighbors. This helps in estimating the maximum limit of the number of iBGP sessions.

We applied our approach on the ISPs listed in table 6.1, and study the scalability

of our approach by measuring the number of iBGP sessions. Through our observations we found that FMS is not employed in any tier-1 ISP. Thus we are not interested in comparing our results to FMS that has deterministic results $\approx O(a^2)$.

The idea of our approach is to establish an iBGP session between every node and each one of its best next hop neighbor towards each ASBR node. A neighbor can be a preferred next hop node for multiple ASBRs, and there is only one neighbor that can be the best next hop to a certain ASBR, unless the IGP routing protocols is configured to load the balance between multiple paths. This gives that in the worst case each node establishes iBGP sessions with all its directly connected neighbors.

Recall table 6.1, we can notice that the average number of neighbors in an AS is relatively small in comparison with the number of the ASBRs. Hence the number of iBGP sessions in an AS that uses Skeleton approach is limited by the value $Nb_{avg} \times a \approx O(a)$. However, In the topologies listed in Table I, we notice that the number of physical links inside the AS is smaller than $Nb_{avg} \times a$. Since Skeleton iBGP sessions are established over physical links, the number of iBGP sessions may not exceed the number of physical links.

Figure 6.1 shows the number of iBGP sessions upon applying the Skeleton algorithm on the topologies listed in Table I. We compare our results with the number of physical links which we found that it is the minimum limit.

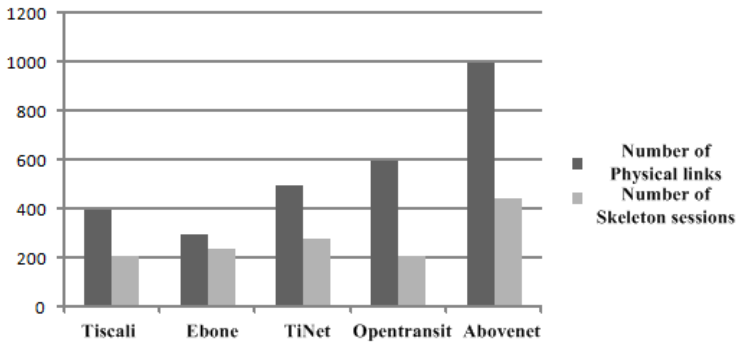


**Figure 6.1:** Physical links vs Skeleton iBGP sessions

Although our results might seem close to its maximum limit (the number of physical links), the maximum limit is still much more smaller than other RRS related proposals. Figure 6.2 compares our results with those from (36) when the two approaches are applied on three backbones taken from (43).

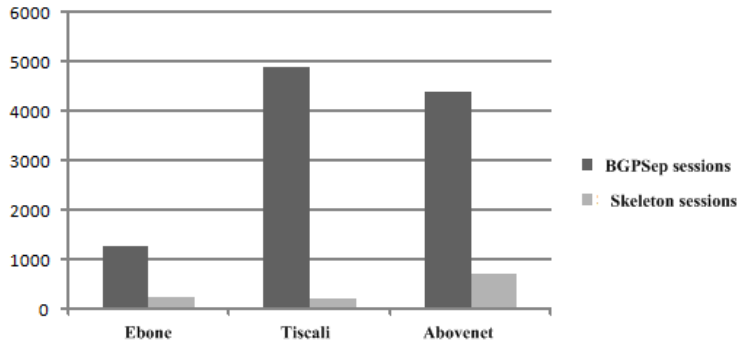


**Figure 6.2:** BGPSep vs Skeleton

One of the main differences between our approach and RRS dependent ones, is that the latter needs to establish full mesh iBGP sessions in the reflectors layer, while Skeleton does not divide the nodes into reflectors and clients, and requires no full mesh iBGP sessions between any two subsets of nodes.

## 6.2 Proof of Correctness

To ensure the correctness and safety of our configuration, we show in this section that iBGP configuration, produced by our approach, satisfies the sufficient correctness conditions given by (20) and (30).

### 6.2.1 Signaling Correctness

Griffin and Wilfong (20) gave sufficient conditions under which a generalized configuration is signaling correct. We first present the sufficient conditions, and then translate them to Skeleton notions, and finally we prove that the Skeleton approach satisfies

them.

#### 6.2.1.1 Sufficient Conditions

- For an $arc(u, v) \in$ **up**, $u$ exports only its routes and those of its clients to the reflector $v$.
- For an $arc(u, v) \in$ **down**, $u$ exports all routes to $v$.
- For an $arc(u, v) \in$ **over**, $u$ exports its routes and those of its clients to $v$.

Where:

- $arc(u, v) \in$ **up** represents the signaling of route update messages from a client $u$ to its route reflector $v$.
- $arc(u, v) \in$ **down** represents the signaling of route update messages from a route reflector $u$ to one of its clients $v$.
- $arc(u, v) \in$ **over** represents the signaling of route update messages between two route reflectors $u$ and $v$.

#### 6.2.1.2 Translation to Skeleton

In a path between $n_0$ and an ASBR $\alpha$, $P_{n_0,\alpha} = n_0 n_1 \cdots n_{m-1} n_m$ ($n_m = \alpha$), the sufficient conditions can be written as follows:

- $n_i$ exports only its routes and the routes of $n_{i-1}$ to $n_{i+1}$
- $n_i$ exports all routes to $n_{i-1}$.
- $u$ exports all its routes and those of its predecessors to $v = peer(u)$.

#### 6.2.1.3 Proof

The first two points can easily be concluded from the successor-predecessor relationship between the nodes $n_{i-1}, n_i, n_{i+1}$. The third point is true because it is equal to saying

that $v = successor(u) \wedge v = predecessor(u)$, thus each node sends all its routes and those of its predecessor to its peer.

### 6.2.2 Forwarding Correctness

Griffin and Wilfong (20) proved that if a signaling correct generalized configuration satisfies the following conditions, then we have a correct generalized configuration. In this section we prove that the generalized configuration in Skeleton model satisfies these sufficient conditions.

#### 6.2.2.1 Sufficient Conditions

- If $(v, u) \in$ **up** and $(w, u) \notin$ **up** then it must be that $u$ prefers paths from $v$ over those from $w$.
- For any nodes $u$ and $v$, $sp(u, v) = P$ for some valid signaling path $P$.

The first condition states that each node $u$ ranks the path that passes by its client before those that they do not. And the second condition means that the signaling paths between any two nodes should be the shortest path between them.

#### 6.2.2.2 Translation to Skeleton

We first need to clarify that in Skeleton we do not distinguish between signaling and forwarding paths because it does not have iBGP sessions over routed paths.

The first condition states that each node $u$ ranks the path that passes by its predecessor before those that they do not. The second condition says that one of the signaling paths between any two nodes should be the shortest path between them.

#### 6.2.2.3 proof

- Condition 1: The way Skeleton is built ensures that an internal node gets all external prefixes from its successors. Preferring successor paths over predecessor ones might lead to deflections. But this preference does not exist in Skeleton,

because the space of choices (of external routes) of an internal node is limited to the paths advertised by its successors.

- Condition 2: Skeleton is built in such a way that ensures that each node establishes an iBGP session with each of its preferred next hops to ASBRs. Thus the signaling paths are always over the shortest link between any pair of nodes.

### 6.2.3 Safety

This solution is based on Skeleton, where iBGP sessions are established between direct neighbors only. Sobrinho (30) has built an algebra to model the route reflection configuration of iBGP. He concluded a sufficient condition under which iBGP converges to a stable state: $dist(reflector(k), k) \leq dist(j, k)$ for all clients $k$ and all reflectors $j$, where $reflector(k)$ is the route reflector of $k$ of the same cluster.

We have seen in subsection 5.3.4 that Skeleton configuration can be implemented using the route reflection notion. Each internal node can be considered as a client to all its adjacent iBGP neighbors. And since those adjacent neighbors are chosen as the best next hops to reach the ASBRs, so the above condition is satisfied, thereby the Skeleton configuration makes iBGP converges to a stable state.

## 6.3 Robustness

Intra-AS topologies (network of routers) are less complex than the Internet (network of ASes), this makes its convergence faster. The importance of intra-AS robustness appears mostly in transit ASes, where the internal topology must ensure a safe and robust connectivity between the two end ASBRs (in-router and exit router).

The best way to ensure robustness is to give all internal nodes the ability to switch to an available path once the primary one is down. In BGP speaking, creating more available paths costs more iBGP sessions in networks that use single-hop iBGP sessions only.

Among the main differences between eBGP and iBGP, is that eBGP is a signaling

and forwarding protocol. However, iBGP has only a signaling functionality and leaves the forwarding one to the IGP protocol. This special characteristic of iBGP is behind its probable divergence. All previous iBGP configurations (full mesh, route reflection, and confederation) have separate signaling and forwarding intra-AS graphs. One of the most distinction between Skeleton and other configuration methods is that it enforces identical signaling and forwarding graphs. This has pros and cons as we will see in the next subsection.

A failure of one physical link between two internal routers in Skeleton graph ends one iBGP session between them. One of them $r_2$ is the best next hop of the other $r_1$ to reach a certain ASBR $\alpha$, such a failure stops updates from $\alpha$ to $r_1$. If $r_1$ has no other iBGP session with any of its neighbors, then it will be unable to reach any external prefix. But if $r_1$ has another iBGP session with another neighbor $r_3$ that is related to different branch, then $r_1$ is able to overcome that failure.

As we have seen in the previous subsection, Skeleton produces iBGP topology similar to the eBGP one. This leads to that we can apply similar backup solution inside the AS, which gives continuous connectivity inside the AS as long as the physical topology is connected. However, an internal node has not iBGP sessions with all its neighbors. Therefore, to make the backup solution feasible inside the autonomous system, we introduce the principle of complete iBGP (52). It states that each internal router establishes iBGP sessions with all its neighbors, thereby, the number of iBGP sessions is equal to the number of physical links inside the AS.

The backup solution that we need to employ inside the autonomous system states that, besides Skeleton's iBGP sessions, each router establishes an iBGP session with a router from different branch for each ASBR-based subgraph. This choice meets with the disjoint property of the backup path for eBGP, that is the backup path should have as minimum common nodes with the primary one as possible.

In the stable state, these additional iBGP sessions do not break the *star-ness* property of ASBR-based model, because this property describes the forwarding side of this model not the signaling one. Once a failure occurs, the backup signaling path becomes

the backup iBGP session that ensures the signaling flow. The main objective of our proposed configuration is to make all internal nodes aware of all external updates, which is ensured by backup signaling paths. However, and since the backup sessions are not necessarily established with the second best next-hop, forwarding traffic may have different path according to the IGP decision, which makes it responsible of the internal routing convergence. In this type of configuration, intra-AS robustness inherits the same order of IGP robustness. Francois *et al.* (46) states that sub-second link-state IGP convergence can be easily met on an ISP network without any compromise on stability. Therefore, making BGP of the same robustness order of an IGP routing protocol is a good achievement.

Three concerns may be appeared because of the additional iBGP sessions inside the autonomous systems: Router's resource consumption, signaling overhead, and routing anomalies. We discuss them in the rest of this section.

### 6.3.1 Router CPU Utilization

Once an update message is received by a BGP speaker, it performs the following tasks (24; 49). First, the appropriate RIB-In needs to be updated. Ingress filtering, as defined in the routers configuration, has to be applied to the route announcement. If it is not filtered out, the route undergoes the BGP route selection rules and it is compared against other routes. If it is selected, then it is added to the BGP routing table and the appropriate forwarding table entry is updated. Egress filtering then needs to be applied for every BGP peer (except the one that sent the original announcement). New BGP announcements need to be generated and then added to the appropriate RIB-out queues. Statistics show that BGP process may increase the CPU (Central Processing Unit) usage to 100% (49). These statistics were collected from eBGP speaker receiving an update from eBGP neighbor, which reflects the high cost for each new eBGP session.

However, adding a new iBGP session has not the same cost since iBGP neighbors do not send as much updates as an eBGP one, because intra-AS topology is less changing than the external one. Further, most iBGP sessions are related to Skeleton, this means

that most of internal routers have sessions with their best IGP next hops and may not transit update messages between ASBRs. Moreover, Skeleton nodes send updates to their one-hop neighbors only, which means that BGP process has no need to look for its BGP next hop in the routing table since its is directly connected.

Despite the above discussion, Skeleton with its backup sessions produces much more less iBGP sessions than other proposed internal configuration, as in figure 6.2 that compares Skeleton's iBGP sessions with R-BGP (41) that performed notable improvement in the number of iBGP sessions.

### 6.3.2 Signaling Overhead

Signaling overhead may cause high link utilization, BGP employs several timers to control its signaling flow. MRAI timer enforces an amount of time between update messages to reduce signaling overhead (CPU and link). Update messages in Skeleton topology are destined to direct neighbors only, this means that each inter-router link carries only the update messages originated by its end routers. However, in all other configuration that use multi-hop iBGP sessions, same link carries all signaling traffic between any pair routers whose IGP path traverses this link. Therefore, each additional (backup) iBGP session will not cause more than, at most, two update messages each MRAI interval.

### 6.3.3 Routing Anomalies

Skeleton approach ensures that every internal router gets updates from all ASBRs, this makes iBGP's signaling traffic passes over the preferred IGP path. Additional iBGP sessions (backup) have no effect on the traffic flow, they just replicate routing information for some routers. The following discussion investigates whether the backup iBGP sessions in Skeleton graph increase the internal routing anomalies:

- Non optimal exit router: An internal router may select a non optimal ASBR as its BGP next-hop if it does not receive routing updates from its optimal one. However, one of the most valuable objectives of Skeleton is to ensure that each

internal router receives updates from its preferred exit point. Additional iBGP sessions act in the opposite direction, because it adds BGP neighbor, and in turn, increases the visibility of internal routers in such a way that it receives updates from optimal and non-optimal ASBR. Standard BGP preference leads to the selection of optimal ASBR.

- Route oscillation: It occurs when one or more router is not able to settle on a stable configuration. Oscillation is a result of non-complete knowledge of internal topology. Once each internal router has all the needed routing information to optimally exit its AS, then there will be no oscillations. As in the previous point, additional backup iBGP sessions increase the internal routing visibility, so it has no bad effect on the routing decision.

- Routing loops: routing loops may not occur in a DAG topology. Skeleton's backup iBGP sessions creates additional sessions between different branches which changes the topology type. However, this change is at the signaling level only, but at the forwarding level the ASBR-based model conserves the DAG topology no matter how many additional backup sessions are added.

### 6.3.4 Multi-hop vs single-hop iBGP session

One of the most distinct properties of Skeleton is that it establishes its iBGP sessions between one-hop neighbors only. This may, on one hand, simplifies iBGP topology, and may, on the other hand, arises scalability concern. Table 6.2 compares between the two methods using different criteria.

## 6.4 Discussion

We discuss in this section two points: first, is Skeleton a complete iBGP graph?. Second, what did we mean by a backup iBGP session?.

Complete iBGP configuration states that an iBGP session should be established over each physical link. Figure 6.1 compares between Skeleton iBGP sessions and physical links, that is identical to complete iBGP graph. Adding backup iBGP sessions to Skeleton graph may make it complete graph. However, we have seen in section 6.3 that router overhead may be increased because of this additional sessions. This motivates us to make the backup sessions different from the primary ones. We suggest to establish backup iBGP sessions between routers' logical interfaces. This gives us the ability to activate and deactivate them when necessary.

Therefore, our proposed configuration might be built in a way different to that in section 5.3.1. That is to establish iBGP sessions over all physical links, but with logical interfaces terminations. Then, and according to the ASBR-based model, we keep all sessions over the branches and deactivate the rest (put them in passive mode).

There is still an open challenge that is how to make the activation of passive sessions done automatically after the occurrence of failures. This issue is related to routers' implementation and is part of our future work.

| | **Single-Hop** | **Multi-Hop** |
|---|---|---|
| Failure | A failure in a link stops signaling exchanges between the two end routers. | A failure in a link that relates to an IGP best path between two iBGP peers does not stop exchanging update messages as long as there is another possible IGP path. |
| Recovery | The recovery is immediate due to the backup iBGP sessions that ensures backup signaling connectivity with all ASBRs. | The recovery is of the order of IGP convergence. |
| Link Overhead | Each link carries the signaling of its two end routers only. | Each link carries all signaling traffic between all routers whose inter IGP path traverse this link. |
| Router CPU load | Update messages are processed at each router. However, in single-hop based configuration, update messages senders are limited to physical neighbors only. | Update messages are sent to remote routers and processed only by the end routers, all routers between the sender and receiver forward it as any data traffic. However, multi-hop iBGP sessions requires much more sessions than those that they do not, especially for ASBRs. |
| BGP process | All internal routers MUST run BGP process, being a BGP speaker means that the router should process each BGP update message and selects the best path, and then disseminates it to other BGP neighbors | A subset of internal routers should be BGP speakers. However, all of them should maintain a full BGP routing table, unless they employ default routes. |
| Routing table | All internal routers have a complete Internet routing table, however, route summary should be employed to reduce the size of internal routers' routing table. Small routers may have default routes to other high capacity internal routers. | Default routes may substitute that huge routing table, however, if an internal node is connected to several ASBRs, where each subset of prefixes has different exit router, then the complete routing table will exist. |

**Table 6.2:** Single-hop vs Multi-hop iBGP sessions

# Chapter 7

# Conclusion and Future Work

## 7.1 Conclusion

This work tackled principally the Internet stability. Illustrating this problematic is divided into two parts according to the hierarchy of the Internet: AS level and router level. Internet topology, from AS level point of view, is a set of nodes, each represents a single AS. Stability of this topology is related to the stability of the routing protocol that is responsible of exchanging routing information between Internet nodes. From router level point of view, each AS is a complex topology by its own, it may contain thousands of routers. Stability at this level is related to the routing protocol inside each AS too. Internal instability affects directly the global Internet stability. This motivates us to divide our work into two parts: inter-domain and intra-domain routing.

Efficient proposals should be applied to well modeled problematic, that is to have the ability to give a correctness proof. This incites us to employ a formal model of the routing process based on abstract algebra, and then use that model to validate the correctness characteristics. The following subsections scan our contributions in this thesis.

### 7.1.1 External BGP

Slow convergence is a well known concern of BGP. On one hand, making this process faster may result in unwanted behaviors. On the other hand, there is no reason for losing traffic since many studies showed that in most cases the underlying infrastructure is connected during the convergence process. This motivates us to turn around the slow convergence problem and propose a solution to overcome the traffic loss problem.

Besides our main objective, that is the continuous connectivity, we were interested in the following characteristics:

- Freeness of loops during and after the convergence process.
- limiting failure propagation.
- Improve the convergence time.

To meet the above objectives we have proposed four mechanisms:

#### 7.1.1.1 Propagating RC information

The basic idea of this mechanism is to create additional information to prevent the other nodes from selecting backup paths that pass through the failed (root cause) node. This might be considered as a loop avoidance mechanism. In some cases, all backup paths may contain that information. Therefore, we propose to add this information at the router level, because the failed node might be a single ASBR while the neighboring ASBR of the same AS is still running. So the RC information is to be included only in the withdraw messages that are exported to other ASes after making sure that no other ASBR can reach the lost destination.

#### 7.1.1.2 Backup solution

This mechanisms is divided into two phases:
- Backup path calculation: This is done during the stable state, each node builds a list of backup paths and rank them according to their disjointness from the primary path

of each prefix.
- Backup path selection: Once a node receives a withdraw message, it filters the backup paths according to the received RC information, and then starts using the best backup path.

#### 7.1.1.3 Selective withdrawal

This mechanism is to limit the propagation of failure. It states that if a node receives withdraw message and has no backup one, then it disseminates it to their neighbors. If it has, then it forwards it only to its backup next hop if this latter was its predecessor in the old primary path.

#### 7.1.1.4 Transient route timer

Since the backup path is not necessarily the second best route, we have employed an additional timer (TRT) to fetch the new best path and install it in the Loc-RIB instead of the backup one. This ensures that the Loc-RIB contains best paths, which is the intended stable state.

Simulation results have shown that disconnectivity is (relatively) eliminated during the convergence process. Another interesting simulation result was the failure boundary, which is the number of ASes that are affected by topology events. PBR algebra gave us an extra validation tool to prove the safety of our proposed backup solution. Moreover, it helps us, through the relationship policy, to apply safety conditions on the backup path in such a way that makes the topology free of loops during and after the convergence process.

### 7.1.2 Internal BGP

Internal AS topology is an inherent part of the global Internet, and internal events are one of the affecting factors of the Internet stability. This makes the stability of iBGP essential for global BGP stability.

In intra-domain routing, and according to the internal configuration, control and data traffic may not have similar topologies. We figured out that this distinction is behind most of intra-domain routing divergence phenomenon. This motivates us to propose an alternative to the current known intra-AS configurations that maps between data and control traffic paths.

We propose the *Skeleton* approach as an alternative to current configurations like: full mesh, route reflection, or confederation. Skeleton is both scalable and anomaly free. It ensures simplicity, correctness, and compatibility with existing operating systems. Skeleton is an easy to employ solution, it can be deployed using the existing devices that support route reflection.

The proposed topological model gives for each ASBR a DAG rooted at the ASBR itself. Skeleton establishes iBGP sessions along each branch in that tree. This ensures that each internal router receives routing information through its best path to quit the AS, and enforces that control and data traffic follow the same paths as well. To increase skeleton's redundancy we have employed a similar backup strategy to the one proposed for eBGP. It states that an internal router establishes additional (passive) iBGP session with its second best next hop related to different branch. We showed that this additional session ensures robustness against failures.

The correctness characteristic is held by proving that our approach satisfies sufficient conditions. The scalability is guaranteed because the number of iBGP sessions in skeleton is limited by the number of internal AS physical links. The backward compatibility is guaranteed using the mapping between reflector/client and successor/predecessor models. Skeleton overcomes the asymmetricity concern between iBGP peers through establishing iBGP sessions between IGP neighbors only. Moreover, MED oscillations are treated by providing every node with all the routing updates from all the egress nodes, which guarantees a stable decision to be taken by each node. Finally we prove that Skeleton configuration is safe using the routing algebra.

### 7.1.3 Validation Methods

Validation is an inherent part of this dissertation. The first challenge was to choose the best validation tool that gives experimental or theoretical proofs of the claimed properties of our proposal. We proposed several approaches and tried to make all of them satisfy the following properties:

- Safety: freeness of loops.
- Robustness against failures.
- Scalability: limited consumption of network resources.
- Backward compatibility: the interoperability with conventional BGP.
- Correctness: reach the topology to the intended stable state.

We could not find a validation tool with which we can prove all the above properties. That is why we have employed several methods to validate our approaches separately. For our eBGP proposal, we rely on SSFNet as a simulation framework to prove that our solution achieves the continuous connectivity, and to measure the convergence time as well as failure propagation. We then employed the routing algebra to validate the safety property of our approach.

In our iBGP proposal, we have used the rocketfuel approach to retrieve internal topologies of some large ASes. We applied our approach on those topologies and compare our results with one of the best route reflection configurations and shows that we produce less iBGP sessions. This ensures the scalability of Skeleton. We validate the safety, robustness, and correctness properties using sufficient correctness conditions taken from (17; 20; 30).

## 7.2 Future Work

Our main objective in this work was to fulfill the stability gap from which the Internet topology suffers. However, and like any other work, ours is not perfect and has its own

limitations and bugs. Therefore, there are still many addons to make our proposed solutions better and more usable. This section lists some of the possible future directions in interdomain routing.

### 7.2.1 Additional parameters

AS relationships play an important role in Internet stability (16). It reflects the commercial agreements between ASes, and determines the rules under which a transit AS may accept forwarding traffic. Taking inter-AS relationships into consideration is done locally between each AS and its neighbors. However, neighboring ASes may not agree on the same policy. For example, an AS may select a path to a certain destination via a peering AS, but its customers may prefer not to send traffic over a peering link if they have another option that has no peering links.

Therefore, we think of that adding a new optional attribute that carries the inter-AS relationship type may improve the BGP decision process. However, we should first investigate whether this additional information can be employed using the existing attributes like the *communities* one!

Another possible parameters that we think it deserves to be considered when BGP select its paths are the external and internal metric. The internal metric is the path cost for each AS between the incoming ASBR and the outgoing one. The external metric represents the link cost between ASes.

### 7.2.2 Security

Several works like (26; 27) have stated that BGP is vulnerable to damaging attacks, and no countermeasures prevent bogus routes from being injected and diffused by BGP. This makes the security policy is indispensable when working on BGP.

Previous works concentrated on prefix hijacking and other anomalies in eBGP. However, in network security, more than 70% of security threats come from the inside of the networks. This leads to that although intra-AS BGP speakers are controlled by single authority, they are subject to the same security threats that the external

speakers suffer from.

We argue that single hop iBGP sessions is one step towards securing iBGP sessions. The composite policy proposed in PBR algebra allows the addition of new policy members and, thereby, takes them into consideration during the decision process. This helps in selecting paths that avoid untrusted ASes in eBGP process.

### 7.2.3 Quality of Service

Talking about quality of service in the context of interdomain routing is still objected by the Internet community. They argue that routing protocol should not be aware of applications' issues because this may create additional overload on a protocol that carries hundreds of thousands of prefixes.

One possible step is to make BGP aware of QoS control protocols like RSVP (Resource Reservation Protocol). In this case BGP will be aware of a digest of QoS information. A step towards this objective was done by Kumaki *et al.* (31). We are optimistic that we can model the QoS policy in our proposed PBR algebra, so we can prove that more reliable paths can be selected when introducing this additional policy.

### 7.2.4 Abstract Routing

Modeling the routing process may have additional advantages other than those related to formal validation. One possible benefit is to use routing models for academic and design objectives. Taylor and Griffin (59) have proposed a configuration language to model routing operation in an abstract way. Their work is a good step to use routing algebra for non-research objectives. A good future plan is to build a simple abstract routing model that can be used to teach students routing baselines as a step before detailing the routing protocols.

# References


[1] Open source management tool. *http://www.opennms.org*, -. 89

[2] Cidr report. *www.cidr-report.org*. 15, 24, 41, 48, 49, 51, 71, 74

[3] Scalable simulator framework. *http://www.ssfnet.org*. 19, 68, 70

[4] A. Rasala F. Shepherd A. Basu, C. Luke Onge and G. Wilfong. Route oscillation in ibgp with route reflection. *ACM SIGCOMM*, 2002. 80

[5] M. Shayman A. Rawat. Preventing persistent oscillations and loops in ibgp configuration with route reflection. *The International Journal of Computer and Telecommunications Networking*, 2006. 39, 80

[6] T. Griffin A. Shaikh, R. Teixeira and J. Rexford. Dynamics of hot potato routing in ip networks. *ACM SIGMETRICS/Performance*, 2004. 17, 49, 51, 79

[7] J. J. Garcia-Luna-Aceves Albrightson, B. and J. Boyle. Eigrp a fast routing protocol based on distance vectors. *In Proceedings of Networld/Interop*, 1994. 62

[8] A. Bose C. Labovitz, A. Ahuja and F. Jahnian. Delayed internet routing convergence. *SIGCOMM*, 2000. 51

[9] D. Walton D. McPherson, V. Gill and A. Retana. Border gateway protocol (bgp) persistent route oscillation condition. *RFC 3345*, 2002. 36, 39

[10] D. Massey D. Pei, M. Azuma and L. Zhang. Bgp-rcn: Improving bgp convergence through root cause notification. *Computer Networks Journal*, 2005. 51, 52, 53, 57

## REFERENCES


[11] S. Sangli D. Tappan and Y. Rekhter. Bgp extended communities attribute. *RFC 4360*, 2006. 26

[12] Jeff Doyle and Jennifer DeHaven Carroll. *Routing TCP/IP*. Cisco Press, 2005. 62

[13] R. Dube. A comparison of scaling techniques for bgp. *ACM SIGCOMM*, 1999. 80

[14] J. W. L. Gao F. Wang, Z. M. Mao and R. Bush. A measurement study on the impact of routing events on end-to-end internet path performance. *SIGCOMM*, 2006. 51

[15] M. Patrignani G. Battista and M. Pizzonia. Computing the types of the relationships between autonomous systems. *IEEE INFOCOM*, 2003. 24, 25

[16] L. Gao. On inferring autonomous system relationships in the internet. *IEEE/ACM Transactions on Networking*, 2001. 24, 25, 110

[17] L. Gao and J. Rexford. Stable internet routing without global coordination. *ACM SIGMETRICS*, 2000. 10, 91, 109

[18] J.J. Garcia-Lunes-Aceves. Loop-free routing using diffusing computations. *IEEE/ACM Transactions on Networking*, 1993. 62

[19] T. Griffin and B. Premore. An experimental analysis of bgp convergence time. *ICNP*, 2001. 51, 55

[20] T. Griffin and G. Wilfong. On the correctness of ibgp configuration. *ACM SIGCOMM*, 2002. 10, 40, 80, 81, 83, 91, 94, 96, 109

[21] T. G. Griffin and G. Wilfong. An analysis of bgp convergence properties. *ACM SIGCOMM*, 1999. 48, 51

[22] Timothy G. Griffin and J.L.Sobrinho. Metarouting. *ACM SIGCOMM*, 2005. 45

[23] S. Jamin S. Shenker H. Chang, R. Govindan and W. Willinger. Towards capturing representative as-level internet topologies. *Computer Networks Journal*, 2004. 41

[24] B. Halabi. *Internet Routing Architecture*. Cisco Press, 1997. 15, 32, 99

[25] G. Huston. Interconnection peering and financial settlements. *INET*, 1999. 25

[26] S. Forrest J. Karlin and J. Rexford. Pretty good bgp: Improving bgp by cautiously adopting routes. *IEEE ICNP*, 2006. 110

[27] S. Ranjan J. Qiu, L. Gao and A. Nucci. Detecting bogus bgp route information: Going beyond prefix hijacking. *SecureComm*, 2007. 110

[28] T.M. Jaafar, G.F. Riley, D. Reddy, and D. Blair. Simulation-based routing protocol performance analysis - a case study. *Simulation Conference, 2006. WSC 06. Proceedings of the Winter*. 62

[29] D.M.Nicol J.H.Cowie and A.T Ogielski. Modeling the global internet. *Computing in Science and Engineering*, 1999. 18

[30] J.L.Sobrinho. An algebraic theory of dynamic network routing. *IEEE/ACM Transactions on Networking*, 2005. 10, 18, 19, 20, 42, 45, 46, 66, 94, 97, 109

[31] R. Zhang K. Kumaki and Y. Kamite. Requirements for supporting customer rsvp and rsvp-te over a bgp/mpls ip-vpn. *IETF RFC draft*, 2008. 111

[32] T. Griffin L. Gao and J. Rexford. Inherently safe backup routing with bgp. *IEEE INFOCOM*, 2001. 51

[33] J. Rexford L. Subramanian, S. Agarwal and R. H. Katz. Characterizing the internet hierarchy from multiple vantage points. *IEEE INFOCOM*, 2002. 24, 25

[34] and K. Nahrstedt L. Xiao, J. Wang. Optimizing ibgp route reflection network. *IEEE INFOCOM*, 2003. 80

[35] L.Xiao and K.Nahrstedt. Reliability models and evaluation of internal bgp networks. *IEEE INFOCOM*, 2004. 81

[36] S. Kopparty M . Vutukuru, P. Valiant and H. Balakrishnan. How to construct a correct and scalable ibgp configuration. *IEEE INFOCOM*, 2006. 81, 94

[37] S. Uhlig M. Buob, M. Meulle. Checking for optimal egress points in ibgp routing. *IEEE DRCN*, 2007. 80

[38] P. Faloutsos M. Faloutsos and C. Faloutsos. On power law relationships of the internet topology. *ACM SIGCOMM*, 1999. 18

[39] Xavier Masip-Bruin M. Yannuzzi and Olivier Bonaventure. Open issues in inter-domain routing: A survey. *IEEE Network*, 2005. 51, 52

[40] R. Musunuri and J. Cobb. Complete solution to stable ibgp. *IEEE ICC*, 2004. 80

[41] D. Katabi B. Maags N. Kushman, S. Kandula. R-bgp staying connected in a connected world. *Usenix NSDI*, 2007. 51, 52, 53, 92, 100

[42] S. Kandula N. Kushman and D. Katabi. Can you hear me now?! it must be bgp. *ACM Journal of Computer and Communication*, 2007. 48, 51, 52

[43] David Wetherall Neil Spring, Ratul Mahajan and Thomas Anderson. Measuring isp topologies with rocketfuel. *IEEE/ACM Trans. on Networking*, 2004. 89, 91, 92, 94

[44] P. Francois O. Bonaventure, C. Filsfils. Achieving sub-50 milliseconds recovery upon bgp peering link failures. *ACM CoNEXT*, 2005. 51

[45] D. Obradovic. Real-time model and convergence time of bgp. *IEEE INFOCOM*, 2002. 51

[46] J. Evans O. Bonaventure P. Francois, C. Filsfils. Achieving subsecond igp convergence in large ip networks. *ACM SIGCOMM*, 2005. 99

[47] D. McPherson P. Traina and J. Scudder. Autonomous system confederations for bgp. *RFC 5065*, 2007. 37

[48] P. Traina R. Chandra and T. Li. Bgp communities attribute. *RFC 1997*, 1996. 26

[49] S. Bhattacharyya S. Agarwal1, C. Chuah and C. Diot. Impact of bgp dynamics on router cpu utilization. *Springer*, 2004. 99

[50] B. Sarakbi and S. Maag. Bgp convergence time, a step towards continuous connectivity. *ITA*, 2009. 57

[51] B. Sarakbi and S. Maag. Bgp skeleton, an alternative to ibgp route reflection. *IEEE INFOCOM*, 2010. 83

[52] B. Sarakbi and S. Maag. Partial complete ibgp. *IEEE ICC*, 2010. 98

[53] S. Shakkottai and R. Srikant. Economics of network pricing with multiple isps. *IEEE INFOCOM*, 2005. 25

[54] S.Hares and R.White. Bgp dynamic as reconfiguration. *IEEE Military Communications Conference*, 2007. 80

[55] H.Jeong S.Yook and A.Barabási. Modeling the internet's large-scale topology. *National Academy of Science*, 2002. 18

[56] D. Katz T. Bates, R. Chandra and Y. Rekhter. Multiprotocol extensions for bgp-4. *RFC 4760*, 2007. 76

[57] R. Chandra T. Bates and E. Chen. Route reflection - an alternative to full mesh ibgp. *RFC 4456*, 2006. 38, 80, 88

[58] F. Shepherd T. Griffin and G. Wilfong. The stable path problem and interdomain routing. *IEEE/ACM Transactions on Networking*, 2002. 36

[59] P. Taylor and T. Griffin. A model of configuration languages for routing protocols. *PRESTO workshop (at SIGCOMM)*, 2009. 111

[60] Boston university representative internet topology generator. *http://www.cs.bu.edu/brite.* 19, 67, 68

[61] O. Maenne M. Roughna W. Mhlbauer, A. Feldmann and S. Uhlig. Building an as-topology model that captures route diversity. *ACM SIGCOMM*, 2006. 41

[62] Z. Morely W. Sun and K. Shin. Differentiated bgp update processing for improved routing convergence. *ICNP*, 2006. 52

[63] MB Weiss and SJ Shin. Internet interconnection economic model and its analysis: Peering and settlement. *Netnomics - Springer*, 2004. 24, 25

[64] T. Li Y. Rekhter and S. Hares. Border gateway protocol 4. *RFC 4271*, 2006. 17, 27, 29, 36