第36卷 第20期 计 算 机 工 程 2010年10月
Vol.36 No.20 Computer Engineering October 2010
·博士论文· 文章编号：1000—3428(2010)20—0016—03 文献标识码：A 中图分类号：TP393

基于SPVP协议的BGP路由收敛算法

包广斌，马栋林，张秋余，袁占亭
(兰州理工大学计算机与通信学院，兰州 730050)

摘 要：针对Internet域间路由慢收敛问题，提出基于简单路径向量协议(SPVP)的BGP路由收敛算法。分析该算法在4种全接连网络拓扑中的 T_{down} 收敛边界值得出，通过检测域间失效链路的根源节点能有效减少路由收敛时间和更新消息开销。SSFNet仿真结果表明，该算法收敛时间上限为 $O(d)$。

关键词：域间路由；边界网关协议；收敛时间；简单路径向量协议

BGP Routing Convergence Algorithm Based on SPVP Protocol

BAO Guang-bin, MA Dong-lin, ZHANG Qiu-yu, YUAN Zhan-ting
(College of Computer and Communication, Lanzhou University of Technology, Lanzhou 730050, China)

【Abstract】Aiming at the slow convergence problem of Internet inter-domain routing, this paper proposes a Border Gateway Protocol(BGP) routing convergence algorithm based on Simple Path Vector Protocol(SPVP). It detects the root-cause node in inter-domain link failure to improve the convergence speed of inter-domain routing and reduce the overhead of routing update message with the analysis of four different T_{down} convergence boundary value in full mesh network topologies. SSFNet simulation results show that convergence upper limit of this algorithm is $O(d)$.

【Key words】 inter-domain routing; Border Gateway Protocol(BGP); convergence time; Simple Path Vector Protocol(SPVP)

1 概述

边界网关协议(Border Gateway Protocol, BGP)是Internet路由体系结构的核心协议，其收敛性已成为人们关心的焦点。域间路由的收敛慢是路径向量算法没有解决的问题。文献[1]提出一个可以加快收敛速度的方法，利用BGP更新报文中的AS_PATH信息构建路由一致性断言，然后通过该断言辨别不可用路由。但是这种方法需要额外的计算资源来进行一致性检测，尤其当需要分辨路由器的策略性故障和自然故障时，不仅增加了协议复杂性，还导致该算法不可增量实现。文献[2]指出，收敛延迟的根本原因在于错误信息的扩散，并提出了错误信息泛洪算法，这样通过快速删除网络中的错误信息实现快速收敛。但该方法不能增量实现，当参与该过程的多个AS(Autonomous System)没有同时实现该算法时，收敛效果并不好。本文提出基于简单路径向量协议(Simple Path Vector Protocol, SPVP)的改进路由收敛算法，分析改进算法在不同全连接拓扑中的 T_{down} 收敛性能，并通过仿真对改进结果进行评估。

2 域间路由收敛性分析

对于距离向量协议，在有限的时间内、在2点之间找到一条最短路径是能实现的，研究表明，一般距离向量协议的收敛复杂度为 $O(n^3)$。通过对BGP路由慢收敛现象的研究，发现造成BGP路由慢收敛有5个主要原因：

(1)链路或路由器失败造成的BGP路由探索延时；

(2)BGP最小路由通告时间会推迟BGP最佳路由的通告时间；

(3)AS间路由策略会影响BGP路由收敛时间；

(4)路由抖动抑制机制会增加BGP路由收敛时间；

(5)随着网络规模和连接密度的增加，BGP路由的收敛时间和消息开销都迅速增大。

针对BGP路由慢收敛问题，提出改进的SPVP，在该协议基础上定义BGP路由收敛概念，提出改进的BGP路由收敛算法。

3 改进SPVP

3.1 SPVP定义

Internet可以被看做一个简单的直连图 $G=(V,E)$[3]，其中，$V=V_N \cup V_P$，$V_N=\{0,1,\cdots,n-1\}$ 表示运行SPVP协议的 n 个节点的集合，V_N 中的每个节点代表一个AS，且不属于网络 G 中的目的节点，节点 V_N 通过链路 E_N 连接；V_P 是网络 G 中所有目的节点的集合。不失一般性，只考虑连接节点0的单一目的网络 p。到目的网络 p 的路径是一条有序节点集合 $P=(v_k v_{k-1}\cdots v_0 p)$，$[v_i,v_{i-1}]\in E_N$。对于所有的 $0\le i\le k$，存在 $v_i\in P$、$[v_i,v_{i-1}]\in P$，且 $Length(P)=k+1$。

SPVP是一个单路径路由协议，对于节点 v，从邻居节点 u 收到的最新路由存放在 $rib\_in(v\leftarrow u)$。在路由宣告初始化结束后，只有最佳路由发生变化才会有新的路由更新。节点 v 根据路由策略选择它的最佳路由，记作 $rib(v)$。当存在2条

基金项目：国家科技支撑计划基金资助项目(2006BAF01A21)；国家自然科学基金资助项目(50877034)；甘肃省自然科学基金资助项目(1010RJZA050)；兰州理工大学优秀青年教师培养计划基金资助项目(Q200913)

作者简介：包广斌(1975－)，男，副教授、博士，主研方向：网络协议分析；马栋林，副教授；张秋余，研究员；袁占亭，教授

收稿日期：2010-04-22 **E-mail：**baogb@lut.cn

相同路径长度的路由时，选择邻居 ID 较小的路由。在实际网络中，V_N 的节点和 E_N 的链路都有可能失效或者恢复，假设链路[u v]的 2 个节点 u 和 v 都能在有限时间内检测到链路的失效或恢复。在节点 v 检测到链路[u v]失效后，$rib\_in\ (v \leftarrow u)$ 就变为 ε。在节点 v 检测到链路[u v]恢复后，节点 v 把它的最佳路由 $rib(v)$发送给节点 u。不管链路状态发生变化，还是收到了路由更新消息，节点 v 都要重新计算最佳路由 $rib(v)$。如果节点 v 的最佳路由发生了变化，则它将把新的最佳路由 $rib(v)$发送给它的邻居路由器。如果节点 v 原有的路由失效，则将给邻居发送 $rib(v)=\varepsilon$ 的路由撤销信息。

3.2 收敛定义

本文研究主要针对单链路故障事件，在模型中把 BGP 路由事件分为了 4 类[3]：

(1)T_{down} 表示链路[0 p]失效；

(2)T_{up} 表示节点 0 检测到链路[0 p]从失效状态恢复可用；

(3)T_{long} 指除了[0 p]以外其他链路失效引起触发；

(4)T_{short} 指除了[0 p]外其他链路恢复引起触发。

为了便于算法描述，给出以下定义。

定义 1(收敛状态) 节点 v 处于收敛状态指当且仅当新事件发生，否则 $rib(v)$不发生变化。

在实际网络中，路由收敛性事件都可以对应到上面 4 种路由事件之一。多点路由故障可以看作是多个独立的单点路由故障来处理。

定义 2(网络收敛延时) 用 T 来表示，指从路由触发事件开始到网络中所有节点收敛所用的时间。

文献[4]指出 T_{up} 和 T_{short} 事件的收敛时间在 $M\times d$ 的范围内，其中，M 是 MRAI 定时器值(默认值 30 s)；d 是网络直径。证明 T_{down} 事件的收敛时间在 $M\times n$ 范围内，n 是网络的节点数。此外，在每条特定方向的链路上，每 M 秒至多发送一个路由宣告，SPVP 的消息时间开销小于 $(|E_N|\cdot\frac{M\cdot n}{M})=|E_N|\cdot n$，其中，$|E_N|$ 表示 AS 间的直连链路数。本文算法主要改善 T_{down} 的收敛时间。

4 基于 SPVP 协议的 BGP 路由收敛算法

下文给出基于 SPVP 协议的 BGP 路由收敛算法，并讨论该算法的 T_{down} 收敛边界值。

4.1 算法描述

由于 SPVP 模型没有周期性路由宣告机制，因此只有链路状态发生变化才会触发路由更新消息。当一条链路状态发生改变，和该链路连接的 2 个节点会检测到这种变化[5]。对于一个给定的目的网络，连接的 2 个节点中最先只有一个节点触发路由改动，这个节点称作根源节点。根源节点把自己的 ID 添加到根源信息中向外传播，后续所有的 SPVP 更新都由此产生。这样网络中的任意节点都会收到唯一根源节点的路由更新消息。本文算法在路由更新消息中携带了根源信息，这样只有直连邻居和受影响节点会收到路由通告。

由于同一个路由根源节点触发的路由更新以不同的速度在多条路径上传播，因此每个节点应能够辨识最新链路状态变化的更新信息。本文算法通过给每个节点 v 保留一个序号 $t(v)$，节点 v 到网络前缀 p 的路由每变化 1 次，序号 $t(v)$增加 1。在本文算法中，路由定义为 r={$r.as\_path$, $r.ts$}、$r.as\_path$ 是 SPVP 的 AS_PATH，$r.ts=\{ts(u)|u\in r.aspath\}$ 是和 $r.as\_path$ 中的节点一一对应的序号列表，$r.ts$、$r.as\_path$ 的任何变化都会引起 $t(v)$的增加。在本文算法中，路由更新定义为 $update$={$update.r$, $update.new$}，$update.r$ 是路由，$update.new$={c, $ts(c)$}指根源节点的 ID 和序号。

为了检测无效暂态路由，每个节点 v 保存一个序号列表。这个序号表保留了节点 v 收到的到达节点 x 的最高序号，用 $seqnum(v, x)$表示。节点 v 在收到式(1)或式(2)的任何一个更新消息后，更新 $seqnum(v, x)$。

$$\begin{cases} x\in update.r.aspath \\ update.r.ts(x)>seqnum(v,x) \end{cases} \tag{1}$$

$$\begin{cases} x=update.new.c \\ update.new.ts(c)>seqnum(v,x) \end{cases} \tag{2}$$

对于 rib_in、rib、$update$ 中的任何路由，有 $t(x)\geq r.ts(x)$；对于网络中任何 new 传播，有 $t(x)\geq new.ts(x)$；对于任意节点 v，有 $t(x)\geq seqnum(v,x)$。

若 $seqnum(v, x)$发生变化，节点 v 将验证它的 rib_in 列表中所有路由。如果 $x\in rib\_in(v\leftarrow u).aspath$ 且 $rib\_in(v\leftarrow u).ts(x)<seqnum(v,x)$，则说明路由 $rib\_in(v\leftarrow u)$ 过期，路由在收敛过程中将撤销该路由。因此，节点 v 可以删除该路由(用 ε 代替)。这样节点 v 可以快速地删除失效路由，缩短收敛时间。

定理 1 在时间点 t，如果 $x\in rib\_in(v\leftarrow u).aspath$ 且 $rib\_in(v\leftarrow u).ts(x)<seqnum(v,x)$，那么节点 u 在网络收敛后必须发送当前路由 $rib\_in(v\leftarrow u)$ 的撤销消息。

证明 在时间点 t，$P=rib\_in(v\leftarrow u).aspath=(x_i,x_{i-1},\cdots,0)$，其中，$x_i$=$u$。假设网络收敛后，节点 v 从节点 u 学习到路由 $rib\_in'(v\leftarrow u)$。

如果 $rib\_in'(v\leftarrow u).aspath\neq P$，那么 $rib\_in(v\leftarrow u)\neq rib\_in'(v\leftarrow u)$。即在时间点 t，路由 $rib\_in(v\leftarrow u)$ 被其他 AS_PATH 替代。

如果 $rib\_in'(v\leftarrow u).aspath=P$，则 $rib\_in(v\leftarrow u).ts\neq rib\_in'(v\leftarrow u).ts$，且在时间点 t 后，路由 $rib\_in(v\leftarrow u)$ 撤销。由于 $rib\_in(v\leftarrow u).ts<seqnum(v,x)$，因此节点 x 改变了它的路由，且 $t(x)\geq seqnum(v,x)$。把 $t(x)$在网络收敛后的值赋给 $T(x)$，则 $T(x)\geq seqnum(v,x)>rib\_in(v\leftarrow u).ts(x)$。本文算法要求节点 x 发送更新 $T(x)>rib\_in(v\leftarrow u).ts(x)$ 给 x_1。节点 x_1 收到节点 x 的更新信息后，必须变更它的序号 $ts(x_1)$，并给 x_2 发送更新信息。类似的，对于所有的 $1\leq j\leq i-1$，x_{j-1} 必须给 x_j 发送路由更新。同理，节点 u=x_i 会接到 x_{i-1} 的路由更新，增加自己的序号，并发送路由更新给节点 v，可得 $rib\_in(v\leftarrow u).ts\neq rib\_in'(v\leftarrow u).ts$。证明完毕。

4.2 T_{down} 收敛边界值

下文讨论节点 v 的收敛边界值。表 1 是边界值讨论中使用的符号定义。

表 1 符号定义

符号表示	说明
h	平均节点延时，即传输经过一个 AS 所用的时间，包括处理延时和传输延时
$l(u,v)$	节点 u 到 v 的延时下限
l	$l=\min_{link[u\ v]\in E_N}\{l(u,v)\}$
$\mu(u,v)$	节点 u 到 v 的延时上限
μ	$\mu=\max_{link[u\ v]\in E_N}\{l(u,v)\}$
$d(u,v)$	节点 u 到 v 的最短 AS 路径
d	网络直径，$d=\max_{u,v\in V}\{d(u,v)\}$

定理 2 若 $conv(v)$表示节点 v 在 T_{down} 事件后的收敛时间，则 $l \cdot d \le conv(v) \le \mu \cdot d$。

证明：设节点 0 是到达目的网络 p 的唯一直连节点，不失一般性，设在事件 T 前有 $t(0)=0$。当事件 T 开始时，节点 0 检测到链路[0 p]失效，节点 0 立刻收敛，$conv(0)=0$。根据收敛时间，把 V_N 中的节点 $v_i(1 \le i \le n-1)$，标记为 $conv(v_i) \ge conv(v_{i-1})$。节点 v_1 收到节点 v_0 的消息后立刻收敛，由于该消息携带 *new* 序号，因此 $ts(0)$增加为 1，节点 v_1 的当前的所有路径失效，并在 T_{down} 事件收敛后重新获得。因此，对于节点 v_1，存在 $l(v_0,v_1)+conv(v_0) \le conv(v_1) \le \mu(v_0,v_1)+conv(v_0)$。

用归纳法证明：$\forall v_i \in V_N$，$\min_{0\le j<i}\{l(v_j,v_i)+conv(v_j)\} \le conv(v_i) \le \min_{0\le j<i}\{\mu(v_j,v_i)+conv(v_j)\}$。假设 v_i 的命题为真，由于每个更新信息包含 *new* 序号使得 $ts(0)=1$，节点 v_{i+1} 收到的任何信息将使 v_{i+1} 当前所有路由失效，并且在后期 T_{down} 事件收敛时重新获得。这样，v_{i+1} 在收到已经收敛的节点 $(v_0,v_1,\cdots,v_i)$ 的首个更新消息后，v_{i+1} 也收敛。v_{i+1} 收到首个更新消息的时间小于 $\min_{0\le j<i+1}\{\mu(v_j,v_{i+1})+conv(v_j)\}$，大于 $\min_{0\le j<i+1}\{l(v_j,v_{i+1})+conv(v_j)\}$。命题为真。

定理 3 本文算法的 T_{down} 收敛的消息开销边界值是 $|E_N|$。

证明：本文算法中每个节点在收到首个更新消息后能立刻收敛，并且向每个邻居最多发送一个撤销消息。由于直连链路数量为 $|E_N|$，因此消息开销边界值为 $|E_N|$。

5 仿真分析

使用仿真软件 SSFNet 对本文算法和文献[1-2]算法进行对比分析。仿真实验在 SSFNet 组件基础上添加了这 3 种算法，利用第 3 方软件包[6]实现 BGP 路由更新消息的统计。

5.1 仿真设置

仿真实验采用 4 种全连接网络拓扑，分别为 4 节点、8 节点、16 节点和 32 节点，图 1 所示为 4 节点拓扑。仿真参数设置如下：MARI 定时器值设为 BGP 默认值 30 s 外加随机抖动时间，链路传播延时为 2 ms，路由信息的处理延时在 0.1 s~0.5 s 内随机选取。按照 SSFNet 默认值，设置数据包大小为 24 Byte、TTL 值为 128、链路带宽为 10 Mb/s，且没有拥塞丢包。

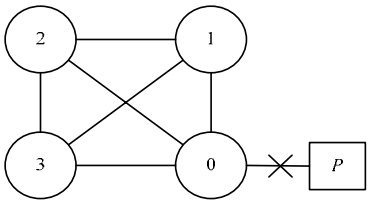

图 1　4 节点全连接拓扑

5.2 T_{down} 结果分析

对于全连接拓扑，选择节点 0 作为源 AS 来宣告目标网络前缀，通过关闭节点 0 来仿真 T_{down} 收敛事件。运行 50 次不同随机种子值得出仿真结果。

图 2 为 50 次运行、置信区间为 95%时的收敛时间对比分析。图 3 为 50 次运行、置信区间为 95%的更新数据包数量对比分析。其中，x 轴和 y 轴均为 lb 坐标。

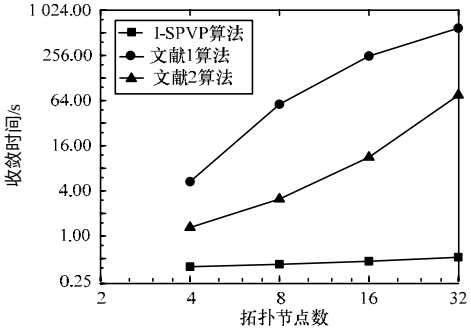

图 2　全连接拓扑中 T_{down} 收敛时间

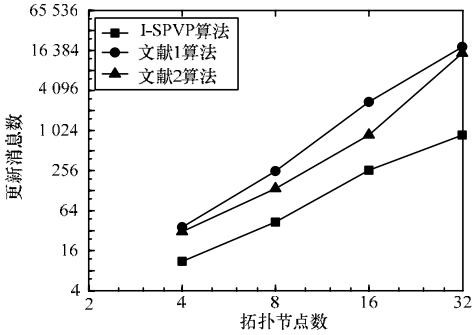

图 3　全连接拓扑中 T_{down} 更新消息

通过对本文算法和文献[1-2]算法分析，得到 3 种算法的 T_{down} 收敛延时和更新消息开销的上限值，如表 2 所示。

表 2　收敛时间和消息开销上限值

算法	T_{down} 收敛延时	T_{down} 消息开销
文献[1]算法	$M \cdot n$	$\|E_N\| \cdot n$
文献[2]算法	$h \cdot n$	$2\|E_N\| \cdot n \cdot h/M$
本文算法	$h \cdot d$	$\|E_N\|$

6 结束语

本文通过对 BGP 路由收敛问题的研究，提出基于 SPVP 的路由收敛改进算法。理论上证明了改进算法能有效降低 BGP 路由的收敛时间和更新消息开销，仿真实验对比分析表明，实验结果和理论分析基本吻合。

参考文献

[1] Labovitz C, Wattenhofer R, Venkatachary S, et al. The Impact of Internet Policy and Topology on Delayed Routing Convergence[C]// Proc. of IEEE INFOCOM'01. NewYork, USA: IEEE Press, 2001.

[2] Afek B Y, Schwarz S. Improved BGP Convergence via Ghost Flushing[J]. IEEE Journal on Selected Areas in Communications, 2004, 22(10): 1933-1948.

[3] Griffin T, Shepherd F B, Wilfong G. The Stable Path Problem and Interdomain Routing[J]. IEEE/ACM Transactions on Networks, 2002, 10(2): 232-243.

[4] Azuma M, Massey D. BGP-RCN: Improving BGP Convergence[J]. Computer Networks, 2005, 48(2):175-194.

[5] 陈文平, 张兴明, 张建辉, 等. 基于距离矢量的多下一跳路由信息协议[J]. 计算机工程, 2010, 36(2): 94-96.

[6] Liljenstam M. The SSFNet DML Reference[EB/OL]. (2009-03-08). http://www.ssfnet.org/InternetDocs/ssfnetDMLReference.html#graph.

编辑　陆燕菲