

基于反馈机制的自愈算法

赵季红^{1,2}, 曲桦¹, 陈文东¹

(1. 西安交通大学 电子与信息工程学院, 陕西 西安 710049; 2. 西安邮电学院 通信与信息工程学院, 陕西 西安 710061)

摘要:提出了一种网络自愈算法,当网络中的节点发生故障或链路出现拥塞时,该算法利用 Q 学习的反馈机制、多 QoS 约束的评价函数和基于 Boltzmann-Gibbs 分布的路径选择策略,自适应地选择恢复路径,降低了选择发生故障和拥塞路径的概率,从而实现了自愈。仿真结果表明,该算法在恢复率、区分业务能力和网络资源优化等方面,表现出了良好的性能。

关键词:故障恢复;网络自愈;Q 学习;多 QoS 约束;区分业务;资源优化

中图分类号:TN915

文献标识码:A

文章编号:1000-436X(2012)01-0001-09

Self-healing algorithm based on feedback mechanism

ZHAO Ji-hong^{1,2}, QU Hua¹, CHEN Wen-dong¹

(1. School of Electronic and Information Engineering, Xi'an Jiaotong University, Xi'an 710049, China;

2. School of Telecommunications and Information Engineering, Xi'an University of Posts & Telecommunications, Xi'an 710061, China)

Abstract: A network self-healing algorithm was presented. The proposed algorithm reduced the probability of selecting the failure or congestion paths and achieved self-healing by selecting recovery path adaptively when the nodes failure or link congestion in the network, which was based on Q-learning feedback mechanism, multi-QoS constrained of the evaluation function and path selection strategy using Boltzmann-Gibbs distribution. Simulation results show that the proposed algorithm outperforms in the recovery rate, the diffserv-aware capability and network resource optimization.

Key words: failure recovery; self-healing; Q-learning; multi-QoS constraints; diffserv-aware; resource optimization

1 引言

随着通信网络宽带化进程的逐步深入,业务类型的日渐丰富,网络的生存性问题成为研究的热点。故障恢复是网络生存性研究的重要组成部分^[1,2],而自愈作为一种智能化的故障恢复技术,其研究也成为了一种趋势。

自愈通常是指网络在发生故障的情况下,不需要人工干预,能很快地、自发地恢复受影响的业务^[3]。

这就要求网络设备具备更强的计算和存储能力。自愈的基本原理是在节点检测到故障之后,立即通知源节点重新选择一条路径,即在边缘节点之间建立另外一条能够满足业务传输需求的路径,将受影响的流量切换到这条备份路径上去,从而保证业务的连续传输。

自愈的研究现状主要集中于 2 个方面:故障检测和故障恢复。故障检测旨在感知并且定位网络故障^[4]。当网络中检测到故障之后,故障恢复机制开始运作。故障恢复旨在将故障路径上的流量切换到

收稿日期:2011-03-24;修回日期:2011-11-23

基金项目:国家自然科学基金资助项目(60872058, 61071126);国家高技术研究发展计划("863"计划)基金资助项目(2009AA01Z213);新一代宽带无线移动通信网国家科技重大专项基金资助项目

Foundation Items: The National Natural Science Foundation of China (6087 58, 61071126); The National High Technology Research and Development Program of China (863 Program)(2009AA01Z213); Next Generation Broadband Wireless Mobile Telecommunications Networks National Science and Technology Major Project

另一条可承载这些流量的健康路径上，减少由于故障对业务传输所造成的影响。

目前故障检测主要有 RSVP 软状态、RSVP HELLO、LSP PING/Traceroute^[5]以及 BFD^[6]等技术。这几种技术所花费的检测时间较长，意味着在故障发生到故障被检测期间，将有大量的数据丢失。故障恢复的研究主要集中于 2 种机制：保护交换和重路由。保护交换的前提是为可能的故障路径提前找到一条备份路径。当故障发生后，将流量切换到备份路径上保证业务的传输。这种机制的主要问题是备份路径会占用大量的网络资源，造成资源的利用率偏低^[7]。重路由主要有 2 个思路，静态路由和在线路由。静态路由将网络在很长一段期间的平均流量作为网络的当前流量信息，在这个基础上计算路由。这种方法简单却精度较差。目前，静态路由主要有最宽最短路径（WSP）算法^[8]、最大可预留带宽（MRB）算法^[9]。在线路由则基于当前网络的状态信息，计算并选择最小可能出现拥塞的路径传输业务，网络状态信息更新的即时性和准确性很大程度上决定了算法的性能。目前，在线路由主要包括最小冲突路由算法（MIRA）^[10]、动态在线路由算法（DORA）^[11]、最小冲突优化算法（LIOA）^[12]等。重路由对于新路径的建立依赖于故障信息、网络路由的策略、预定义设置以及网络当前的状态信息，利用这些信息的路由算法，其收敛时间得不到保证，导致故障恢复时间不能满足业务的 QoS 传输。

根据上述分析，自愈技术的研究仍有许多问题亟待解决。第一，自愈以故障检测为前提，对故障检测结果的依赖程度过高。首先，自愈算法的触发是在故障检测之后进行的，当网络发生故障后，只有该故障被感知并且被定位之后才能触发自愈算法，因此自愈算法需要等待故障检测所需的时间。其次，如果故障未被成功检测，那么自愈算法就不能被触发，从而导致故障恢复失败。第二，自愈进行重路由计算的时间过长，其收敛的时间很难符合业务传输质量的要求。而且基于 QoS 约束重路由计算需要随时掌握当前网络状态信息，这些信息的收集往往通过泛洪的方式，会造成一定的资源浪费。

针对上述问题，本文提出了一种基于反馈机制的自愈算法。该算法通过引入 Q 学习的反馈机制，可以感知网络当前的状态信息，也可以感知到路径发生故障或者拥塞的情况，这样大大降低了自愈对于故障检测的依赖程度，不必对故障进行精确地定位，只需要

模糊的感知故障或拥塞，自愈就可以启动。另一方面，该算法可以通过业务的传输收集路径的信息（时延、带宽等），通过对这些信息的学习，可以获得学习“经验”并作为重路由选择的依据，避免了繁杂的重路由计算过程，降低了重路由所需的时间。同时，该算法引入了多 QoS 参数约束评价函数的路由计算以及 Boltzmann-Gibbs 分布的路径选择策略，使算法具备了区分业务能力以及网络资源优化能力。

本文首先介绍了算法设计中的几个基本概念，然后详细介绍了算法应用于网络中的网络模型、算法的数学模型以及算法实现的具体流程和步骤。最后，用 MATLAB 对该算法进行了仿真，并对算法的性能进行了讨论。

2 算法设计中的基本概念

2.1 反馈机制

本文的研究旨在通过引入机器学习理论改善目前传统自愈机制缺乏对于网络环境的学习能力的状况，提高自愈机制的智能性。机器学习理论中的学习策略很多，比如类比学习、归纳学习、支持向量机和强化学习等，其中强化学习源于心理学中的“试错法”，通过不断与环境交互来改善策略最终找到适合环境的最优策略，这种与环境交互的思想可以应用到网络中实现对网络环境的学习。

本文中提出的算法通过引入强化学习理论中的 Q 学习^[13]反馈机制来感知网络当前状态的变化，一定程度上缓解了传统自愈机制对于故障检测技术依赖程度过高的问题。同时，利用反馈机制可以降低在线路由计算中由于泛洪流量对网络资源的浪费。反馈机制原理如图 1 所示。

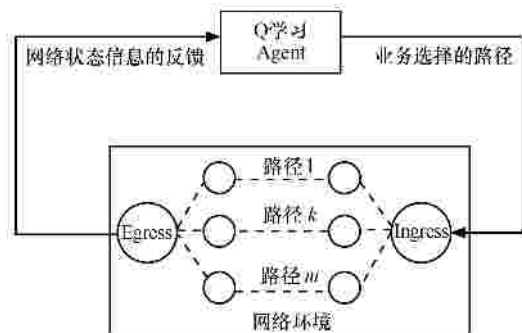


图 1 基于 Q 学习的反馈机制

图 1 中，当每个业务传输结束的时候，目的节点 Egress 会将该业务传输路径的状态信息反馈给源节点 Ingress。Ingress 通过这个反馈回来的状态信息感知该

路径的承载能力。反馈回来的状态信息本质上是业务传输完成后学习到的“经验”。这样就可以避免每次重路由时重新计算新的路径，而是直接利用先前的“经验”自适应地选择合适的备份路径，从路由重计算到路由重查表，可以大大节省不必要的计算负担。

考虑到可以人为定义反映网络状态信息的评价函数，那么可以把其定义为一个数值。这样反馈流量占用的带宽和传输产生的时延可以忽略不记。因此，这种反馈机制较传统的泛洪广播网络状态信息的方式更加简单，而且能够有效减少网络中不必要的泛洪流量。

综上所述，通过反馈机制，Ingress 节点依靠感知当前网络状态并且积累学习“经验”来选择路径，在很大程度上降低自愈机制对故障检测的依赖程度，同时提高了网络资源利用率。

2.2 业务分类

网络中的有些业务对实时性的要求较高，必须满足其实时传输，才能保证业务的服务质量(QoS)；而有些业务对网络的吞吐量和分组丢失率的要求较高，只有足够大的吞吐量和足够低的分组丢失率才能保证业务的服务质量；还有一些业务对网络的要求不高，采用传统“尽力而为”服务即可，这类业务应该尽量少地占用网络资源。

为了合理地利用有限的网络资源，本文提出的算法的路由策略会依据不同业务类型选择适合其传输需求的最优路径。在本文中，将业务分成 3 种类型：EF 业务、AF 业务和 BE 业务。EF 业务为加速转发业务，AF 业务为确保转发业务，BE 业务为尽力而为转发业务。3 种业务对于 QoS 参数的要求参见表 1。

表 1 不同业务类型对 QoS 参数的要求

| 业务类型 | 吞吐量 | 时延 | 分组丢失率 | 节点利用率 |
|------|-----|----|-------|-------|
| EF | 较高 | 高 | 较高 | — |
| AF | 高 | 较高 | 高 | — |
| BE | — | — | — | 尽量小 |

2.3 多 QoS 约束评价函数

一些自愈技术在寻找备份路径进行故障恢复的时候只考虑单一因素的约束，这样带来的问题是选择的路径不一定能够充分地满足业务传输的 QoS 需求，因为保证业务 QoS 传输需要考虑的参数很多，只满足其中一个不一定能达到传输的要求。因此，本文提出的算法引入多 QoS 参数的约束，这样反馈的评价函数所包含的网络状态信息会更加全面。

本文中，选取 3 种常用的 QoS 参数作为约束集合：吞吐量 (TH)、端到端时延 (delay) 和分组丢失率 (P)。这些参数都体现的是路径端到端的性能，综合考虑这些约束因素选择出来的路径就会融入这些约束信息，使路径的选择更加科学和可靠，更能满足业务的 QoS 传输需求。很明显，EF、AF 业务的优先级和重要性要远远高于 BE 业务，因此希望 BE 业务尽量少地占用网络中的节点和链路资源。使用 U 表示路径对网络节点和链路资源的利用率， $U=N_{path} / N_{all}$ ，其中， N_{path} 表示一条路径使用的节点数， N_{all} 表示网络中节点个数总和。那么，约束集合 $C=\{TH, delay, P, U\}$ 。

评价函数为约束集合 C 的函数，即 $f(C)$ 。评价函数的大小就体现了网络中路径的端到端的承载能力的好坏。其值越大，说明该路径当前的承载能力越强。

2.4 资源优化

静态路由算法通常会使业务集中于一条基于某种准则的“最优”路径。当网络中负载较大时，会产生路径拥塞，这就需要全局服务器对流量进行控制来解决这一问题。在本文中，源节点 Ingress 可以得到评价函数提供的信息，因此可以感知到网络当前各条路径的负载情况，从而避免选择负载较大的路径进行业务的传输。但网络中的负载情况是随着时间变化而变化的，这就需要算法对网络状态信息的动态感知。本文中，源节点具有动态感知网络状态信息的能力，从而能够依据路径选择策略自适应地选择路径，使网络负载均衡，实现网络资源的优化。

3 基于反馈机制的自愈算法

基于反馈机制的自愈算法以强化学习中的 Q 学习算法作为理论基础，应用于网络中以增强自愈机制的学习能力，提高自愈机制对于网络环境的感知力，实现网络的智能化控制。

3.1 数学模型

本文中反馈机制的实现是基于强化学习理论中的 Q 学习算法。Q 学习就是要在系统动力学特性未知的情况下估计最优策略的 Q 值，详见文献[13]。在线 Q 学习方法的实现是按递归公式(3)进行的：在每个时间步 t ，观察当前状态 S_t ，选择和执行动作 A_t ，再观察后续状态 S_{t+1} 并接受立即回报 R_t ，然后根据式(3)来调整 Q 值。

$$Q_{t+1}(S_t, A_t) = (1 - a)Q_t(S_t, A_t) + a [R_t + g \cdot \max_A Q_{t+1}(S_{t+1}, A_{t+1}) - Q_t(S_t, A_t)] \quad (1)$$

式(1)中, a 为学习率, 它控制着学习的速度, a 越大则收敛越快。但是过大的 a 可能导致不收敛。Watkins 证明了 a 在满足一定条件时, 如果任一个二元组 (S_t, A_t) 能用方程(1)进行无穷多次迭代, 则当 t 趋于无穷时, $Q_t(S_t, A_t)$ 以概率 1 收敛到关于最优策略的 $Q^*(S_t, A_t)$ ^[13]。

根据多 QoS 参数约束的需要, 定义立即回报 *Reward* 为多 QoS 约束的评价函数以满足业务的 QoS 需求。从上文可知, *Reward* 是约束集 C 的函数。令 $R=\{R_1, R_2, \dots, R_m\}$ 表示每个行动对应的 *Reward* 的集合。定义

$$Reward = w_1 TH + w_2 / delay + w_3 / P + w_4 / U \quad (2)$$

式(2)中 w_1 、 w_2 、 w_3 和 w_4 为权系数。调整权系数的大小可以体现出不同类型业务 (EF 业务、AF 业务和 BE 业务) 的需求。

Q 学习算法中, Agent 面临的一个问题是如果选择下一个动作, 通常需要考虑 2 方面的因素。其一是 Agent 必须对状态动作空间做充分的探索从而能够找出最优或近最优的策略。另一方面是利用通过学习已获得的经验进行动作选择, 从而使学习的成本降低。可以采用连续可微且近似贪婪的 Boltzmann-Gibbs 分布^[14]作为动作选取策略, 动作 A_k 被选取的概率为

$$prob(A_t = A_k | S_t) = \frac{\exp(Q(S_t, A_k) / T_t)}{\sum_A \exp(Q(S_t, A) / T_t)} \quad (3)$$

式(3)中, $T_t (>0)$ 为温度参数, 控制行为选取的随机程度。为了提高学习的速度, 利用模拟退火

技术在学习过程中按式(4)进行动态调整温度值, 即在学习的初期选择较大的温度, 以保证动作选取的随机性较大, 增加搜索能力, 然后在学习的过程中逐渐降低温度, 保证以前的学习效果不被破坏^[14]。

$$\begin{cases} T_{t+1} = T_{\min} + b(T_t - T_{\min}) \\ T_0 = T_{\max} \end{cases} \quad (4)$$

式(4)中, b 为退火因子, 并且 $0 < b < 1$ 。

3.2 网络模型

本文提出的基于反馈机制的自愈算法可以应用在任何 Mesh 网络中。本文将以 MPLS 网络作为应用场景对算法的实现进行详细地研究。

将 MPLS 网络的拓扑抽象为 $G(V, E)$, V 代表网络中的节点 (路由器), E 代表网络中的链路。令 $S=\{S_1, S_2, \dots, S_n\}$ 表示 MPLS 网络环境中的边缘节点的集合 (边缘路由器 LER 或者自治域范围内的边缘标签交换路由器 LSR), $S_i \in V$ 。假设 S_1 为业务传输源节点, 那么选择 S_1 作为 Q 学习 Agent, 潜在的目的节点集合 $S'=\{S_2, \dots, S_n\}$ 即作为 Q 学习中的环境状态集, 显然 S' 为有限集。源节点与目的节点之间所有可能的传输路径集合为 $A=\{A_1, A_2, \dots, A_m\}$, A 即为 Q 学习 Agent 可以采取的行动集合, 即是 MPLS 网络的标签交换路径 (LSP)。Agent 每采取一次行动, 即每选择一条路径, 都会产生一个与这次行动相对应的回报 *Reward* (式 (2)) 反馈给源节点, 然后根据式(1)更新此次业务传输之后对应路径的 $Q_t(S_t, A_j)$ 值, S_i 表示业务传输的目的节点; A_j 表示业务到达 S_i 的传输路径。网络模型如图 2 所示。

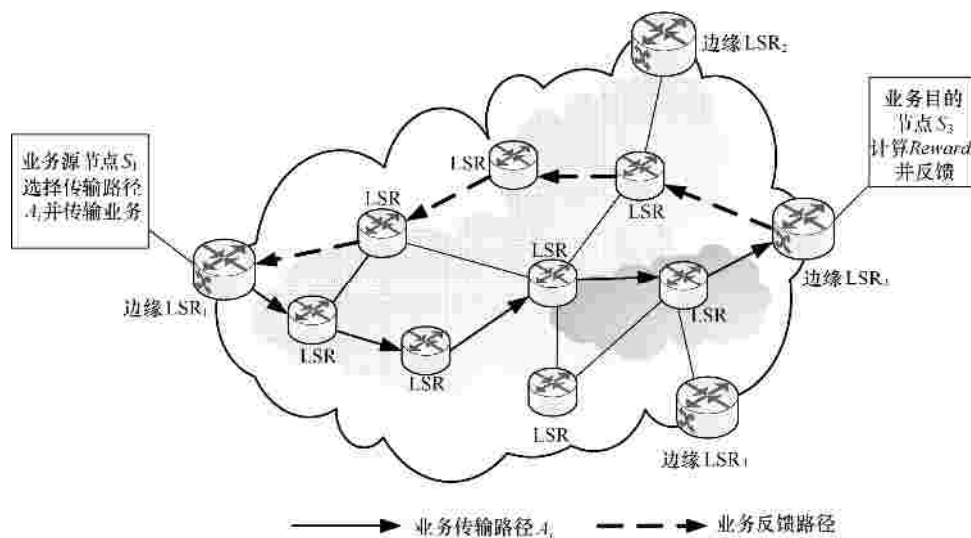


图 2 Q 学习在 MPLS 网络中的应用模型

$Q_i(S_i, A_j)$ 值大小决定了选择路径 A_j 到目标节点 S_i 的“倾向”。将所有 Q 值建立起来的矩阵 $Q_i(S_i, A_j) (i \in [1, n], j \in [1, m])$ 定义为 Q 表格，表示业务的路由信息。 S_1 节点中储存的 Q 表格如表 2 所示。

| 目标节点 | 传输路径 | | | | |
|-------|-----------------|-----|-----------------|-----|-----------------|
| | A_1 | ... | A_k | ... | A_m |
| S_2 | $Q_i(S_2, A_1)$ | ... | $Q_i(S_2, A_k)$ | ... | $Q_i(S_2, A_m)$ |
| ... | ... | ... | ... | ... | ... |
| S_n | $Q_i(S_n, A_1)$ | ... | $Q_i(S_n, A_k)$ | ... | $Q_i(S_n, A_m)$ |

当业务传输结束并反馈 $Reward$ 更新 Q 表格信息之后，就可以以式(3)作为路径选取策略来选择新业务的传输路径，更新后的 Q 表格融入了反馈信息，因此具备网络状态的感知能力，同样也具备了故障的感知能力。

3.3 算法流程

算法的实现流程如图 3 所示。

具体实现步骤如下。

step1 当业务从源节点接入后，判断业务的类型。

step2 源节点根据业务的类型查询相应的路径优先级表格 (EF 业务 Q 表格、AF 业务 Q 表格和 BE 业务 Q 表格)。

step3 依据基于 Boltzmann-Gibbs 分布的路径选择策略 (式(3)) 得到业务的传输路径 A_{i_0} 。

step4 在业务传输过程中经过链路时，记录链路的吞吐量和时延信息。

step5 在业务到达目的节点后，将所得到的路径信息进行整合，计算出路径吞吐量、路径时延以及业务数据的分组丢失率，根据式(2)计算得到此次业务传输的 $Reward$ ，并反馈给源节点。

step6 利用 Q 学习算法 (式(1)) 得到更新

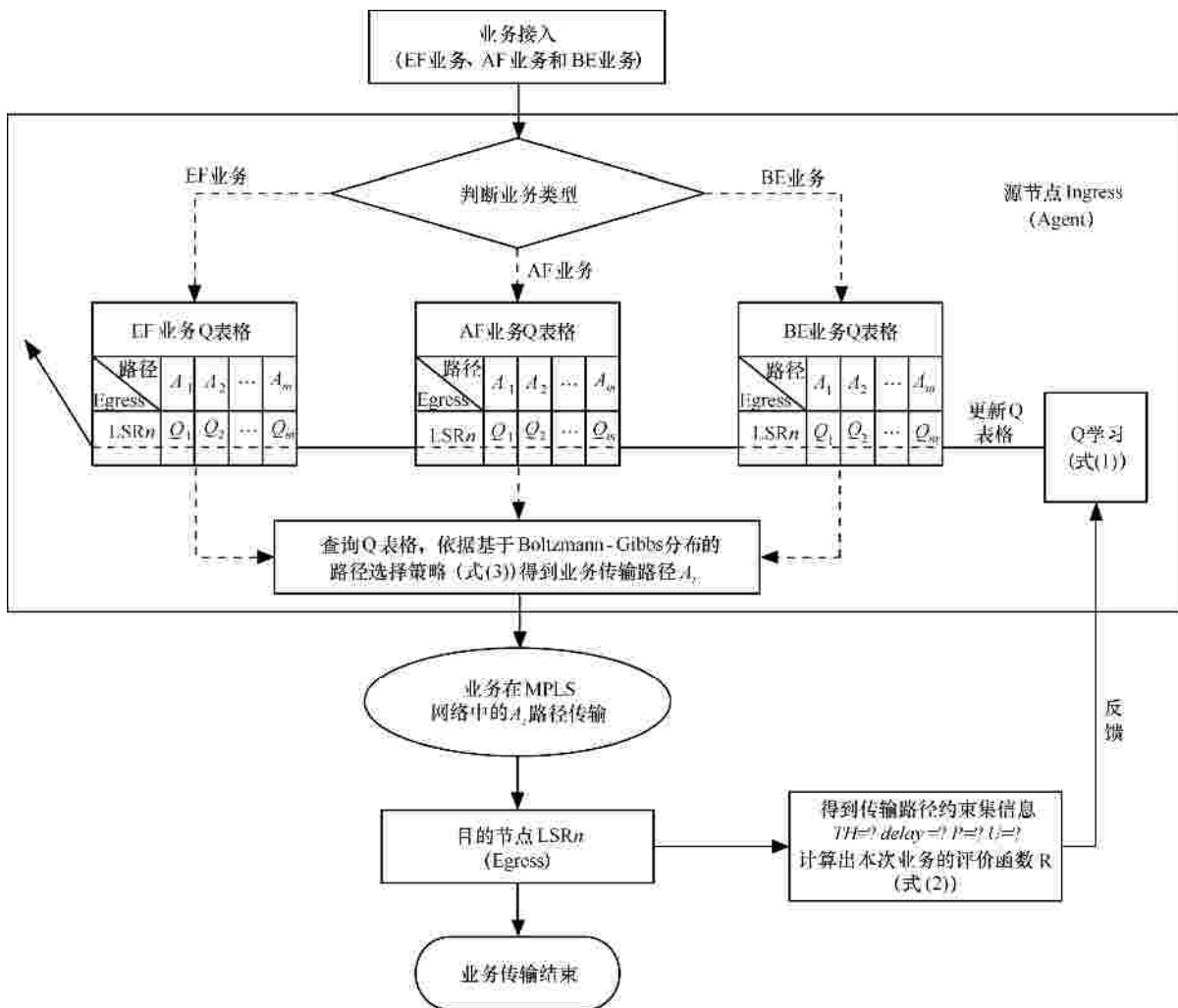


图 3 算法的实现流程

后的 Q 值,从而对路径的优先级信息进行更新。新业务到来时便查询更新后的相应业务的 Q 表格,重复如上的步骤找到传输路径。

在本文中,业务的传输需要考虑 2 种情况,一种是业务传输成功的情况,一种是业务传输失败的情况。在第 1 种情况下,对算法收敛速度的要求不能过高,因为过快地收敛可能导致较大的扰动,学习的精确性不能得到很好地保证。在第 2 种情况下,业务传输失败很可能是由于链路或者节点出现故障造成的,对于这种情况,希望算法能对其进行较快地学习,因此希望收敛速度快。这 2 种情况可以通过使用不同的学习率 a 实现。算法可用下面的伪代码表示。

算法 1 基于反馈机制的自愈算法

对于任意源节点 S_i S ;

初始化 $Q_i(S_j, A_k)$, 其中, S_j 为目的节点, A_k 为选择的路径, 在所有 A_k 中, 取最少跳数路径的 $Q_i(S_j, A_k)$ 为 1, 其余值为 0;

根据式(3)的随机分布选择路径, 假设选择路径 A_m ;

根据式(2), 计算相应的 *Reward* 函数;

如果业务传输成功, 依据下式更新 Q 表格:

$$Q_i(S_j, A_m) = (1 - a_1)Q_i(S_j, A_m) + a_1[R_t + g_1 \max_A Q_i(S_j, A) - Q_i(S_j, A_m)]$$

如果业务传输失败, 依据下式更新 Q 表格:

$$Q_i(S_j, A_m) = (1 - a_2)Q_i(S_j, A_m) + a_2[R_t + g_2 \max_A Q_i(S_j, A) - Q_i(S_j, A_m)]$$

4 仿真与性能分析

本节将对算法进行仿真并依据仿真结果分析算法各方面的性能。需要特别说明的是本文提出的算法改变了传统的自愈机制的以故障检测为前提的故障恢复模式, 而是利用 Q 学习反馈机制的故障感知能力先于定位故障提前响应故障, 因此缺少同类的自愈机制进行比较。在仿真中, 将采用无反馈的静态自愈机制说明基于反馈的自愈机制对于网络环境的学习能力的改善。

4.1 网络参数设置

仿真中的网络拓扑如图 4 所示。其中, LSR_1 为源节点, 同时也是 Q 学习 Agent。 LSR_{15} 为目的节点, *Reward* 将从这个节点反馈到 LSR_1 以更新 Q 表格。

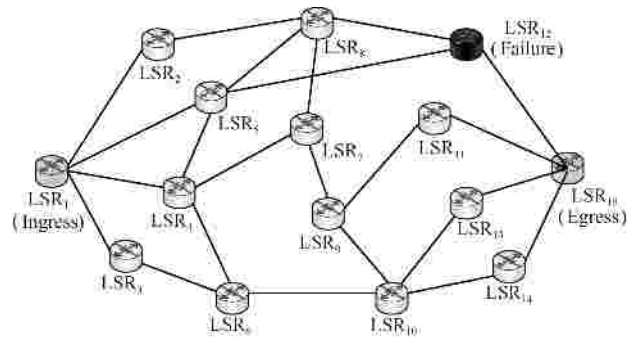


图 4 网络拓扑 (LSR_{12} 为故障场景中的故障节点)

LSR_1 和 LSR_{15} 之间有 22 条可用传输路径, 路径信息参见表 3。其中, 对路径的约束参数包括路径吞吐量 (Mbit/s) 路径时延 (ms) 和路径节点数。在仿真中, 路径的分组丢失率与路径剩余带宽之间呈指数关系。

表 3 仿真参数设置

| 序号 | LSR ₁ 与 LSR ₁₅ 间的恢复连接 | 吞吐量 / (Mbit·s ⁻¹) | 时延 / ms | 静态路径选择 | | |
|----|---|-------------------------------|---------|--------|----|----|
| | | | | EF | AF | BE |
| 1 | 1-4-5-12-15 | 6 | 6.5 | v | v | v |
| 2 | 1-4-7-9-11-15 | 4 | 8.2 | v | | v |
| 3 | 1-4-7-9-10-13-15 | 4 | 10.8 | | v | v |
| 4 | 1-4-7-9-10-14-15 | 4 | 9.7 | v | | v |
| 5 | 1-4-7-8-12-15 | 6 | 12 | v | | v |
| 6 | 1-2-8-7-9-11-15 | 4 | 22.8 | | | v |
| 7 | 1-2-8-7-9-10-13-15 | 4 | 21.2 | | v | v |
| 8 | 1-2-8-7-9-10-14-15 | 4 | 33.4 | | | v |
| 9 | 1-2-8-12-15 | 6 | 27.2 | | | v |
| 10 | 1-5-4-7-9-11-15 | 4 | 28.4 | | v | v |
| 11 | 1-5-4-7-9-10-13-15 | 4 | 17.6 | | v | v |
| 12 | 1-5-4-7-9-10-14-15 | 4 | 15.3 | v | | v |
| 13 | 1-5-4-7-8-12-15 | 6 | 24 | | | v |
| 14 | 1-5-12-8-7-9-11-15 | 4 | 39.8 | | | v |
| 15 | 1-5-12-8-7-9-10-13-15 | 4 | 29.7 | | | v |
| 16 | 1-5-12-8-7-9-10-14-15 | 4 | 34.6 | | | v |
| 17 | 1-5-12-15 | 6 | 16 | | v | v |
| 18 | 1-3-6-10-9-7-4-5-12-15 | 4 | 28.7 | | | v |
| 19 | 1-3-6-10-9-7-8-12-15 | 4 | 38.9 | | | v |
| 20 | 1-3-6-10-9-11-15 | 5 | 7.6 | v | v | v |
| 21 | 1-3-6-10-13-15 | 5 | 6.8 | v | v | v |
| 22 | 1-3-6-10-14-15 | 5 | 7.2 | v | v | v |

表 3 中, 静态路径选择指的是静态路由算法中 3 种业务可以选择的路径。EF 业务选择的路径为 1、2、4、5、12、20、21 和 22; AF 业务选择的路径为 1、3、7、10、11、17、20、21 和 22; BE 业务可以选择所有路径。静态算法没有引入反馈机制, 作为与本文提出算法的对比算法。

在仿真中，设置 LSR₁₂ 为可能发生故障的节点。从表 3 中可以看出，如果 LSR₁₂ 发生故障，受影响的路径有 1、5、9、13、18 和 19。仿真中还加入了链路拥塞的情况，为了便于观察，设置路径 22 为可能发生拥塞的路径。因此，仿真中包括 3 种场景，分别为无故障场景、LSR₁₂ 故障场景和路径 22 拥塞场景。

在仿真中，需要在网络负载大小变化条件下讨论算法性能，因此，对网络负载定义如下 10 个等级，如表 4 所示。等级越高，表明网络负载越大。

表 4 负载级别定义

| 级别 | 业务所需带宽平均值/(Mbit·s ⁻¹) | 级别 | 业务所需带宽平均值/(Mbit·s ⁻¹) |
|----|-----------------------------------|----|-----------------------------------|
| 1 | 0.100 0 | 6 | 0.133 3 |
| 2 | 0.105 3 | 7 | 0.142 8 |
| 3 | 0.111 1 | 8 | 0.153 8 |
| 4 | 0.117 6 | 9 | 0.166 7 |
| 5 | 0.125 0 | 10 | 0.181 8 |

4.2 性能分析

4.2.1 故障感知

本文中提出的算法通过业务传输之后评价函数的反馈机制感知网络当前的状态信息。因此，该算法具备故障感知的能力。令

$$j = \frac{\text{业务重传总数}}{\text{业务总数}}$$

j 越小说明算法感知故障的能力越强；反之，算法感知故障的能力就越弱。仿真结果如图 5 所示。图 5(b)反映了本文提出的算法对故障感知的性能，图 5(a)反映了没有加入反馈机制的静态路由算法对故障感知的性能。从仿真结果可以看出，基于反馈机制的算法在故障感知性能上表现出了一定的优势。

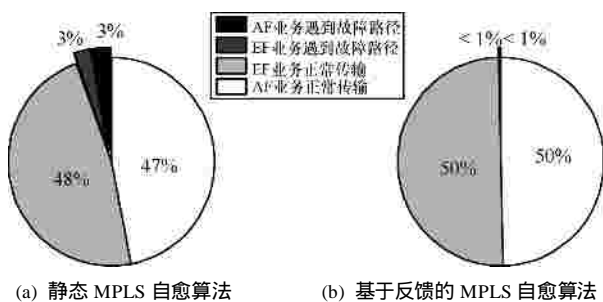


图 5 故障感知性能

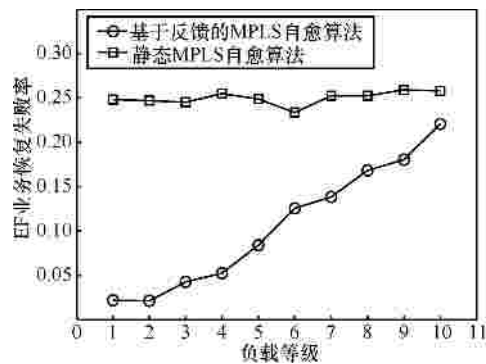
4.2.2 故障恢复

在本文算法的自愈机制中，如果 EF 业务或者 AF 业务传输失败时，即选择了故障路径时，源节

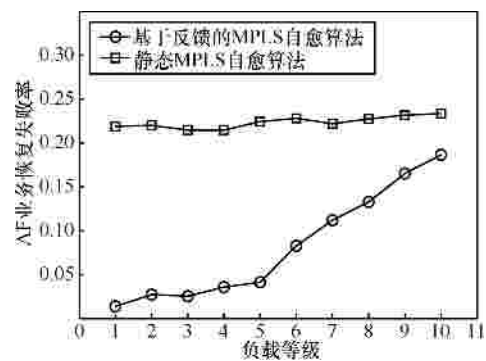
点会重新选择路径传输业务。如果重新选择的路径不能满足业务传输的 QoS 或者仍然选择了故障路径，则认为业务恢复失败，即没有找到合适的备份路径保证业务的传输；如果重新选择的路径可以满足业务的 QoS，并且不是故障路径，则认为业务恢复成功，即找到了一条合适的备份路径保证业务的传输。因此，业务恢复失败率可以表示为

$$m = \frac{\text{业务重传失败数}}{\text{业务重传总数}}$$

仿真结果如图 6 所示。



(a) EF 业务故障恢复失败率与负载等级的关系



(b) AF 业务故障恢复失败率与负载等级的关系

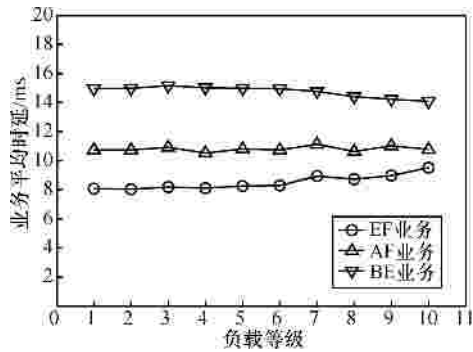
图 6 故障恢复性能

4.2.3 区分业务

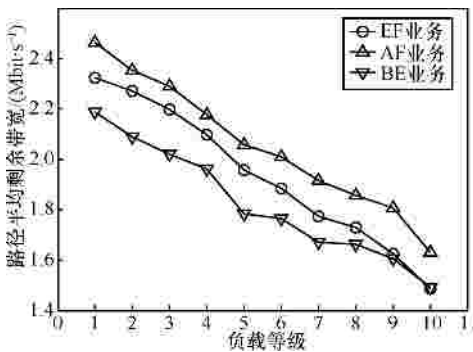
图 7 从业务平均时延、路径平均剩余带宽和业务平均分组丢失率 3 个方面说明了算法对于 EF、AF 和 BE 业务的区分能力。

从图 7(a)中可以明显看出，EF 业务的平均时延最小，AF 业务次之，BE 业务最大。这说明，算法对于 EF 业务，更倾向于选择时延小的路径保证业务的实时传输，而 AF 业务和 BE 业务的时延保证则相对较差。从图 7(b)中可以明显看出，AF 业务选择路径的平均剩余带宽最大，EF 业务次之，

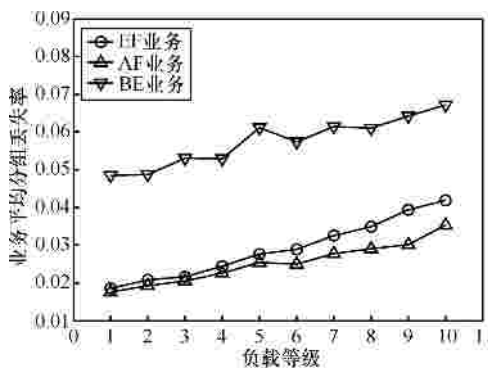
BE 业务最小。算法对于 AF 业务，更倾向于选择吞吐量大的路径保证业务的传输，而对 EF 业务和 BE 业务，则这方面的保证相对较差。从图 7(c)中可以明显看出，AF 业务的平均分组丢失率最小，EF 业务次之，BE 业务最大。这说明，算法对于 AF 业务和 EF 业务，更倾向于选择分组丢失率小的路径保证业务的传输，而对 BE 业务的分组丢失率保证相对较差。



(a) 业务选择路径的平均时延



(b) 业务选择路径的平均剩余带宽



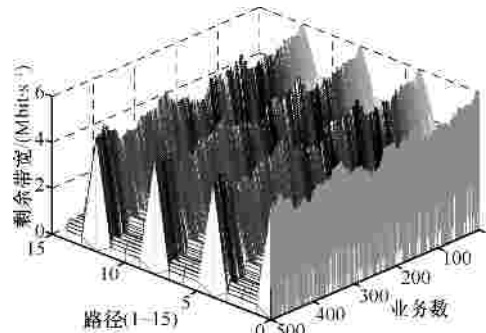
(c) 业务选择路径的平均分组丢失率

图 7 区分业务性能

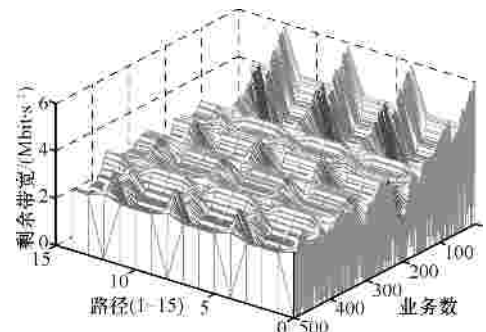
图 7 的结果说明算法对 EF、AF 和 BE 3 种业务进行了区分服务，在有限的网络资源条件下，满足了 EF 业务和 AF 业务的传输需求。

4.2.4 网络资源优化

本文提出的算法通过反馈机制可以感知路径的剩余带宽，在源节点选择路径时，会根据剩余带宽情况动态进行路径的分配。因此，网络中流量会比较均衡。而静态路由算法由于没有引入反馈机制，缺少对路径剩余带宽的感知能力，因此在选路时，不会考虑到网络的负载情况，只根据静态路由表进行路由。因此，各路径的负载情况不一定会均衡，网络资源利用率偏低。将图 8(a)和图 8(b)进行比较，可以明显看出，基于反馈机制的自愈算法可以使网络中各路径的负载均衡化，具有网络资源优化的能力，在图中显示的效果是路径剩余带宽平面比较平滑，而静态路由算法中的路径剩余带宽平面就显得凹凸不平。



(a) 静态 MPLS 自愈算法

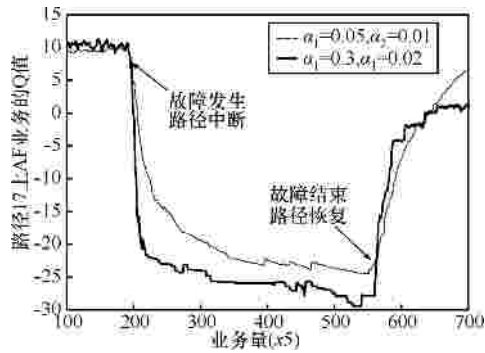


(b) 基于反馈的 MPLS 自愈算法

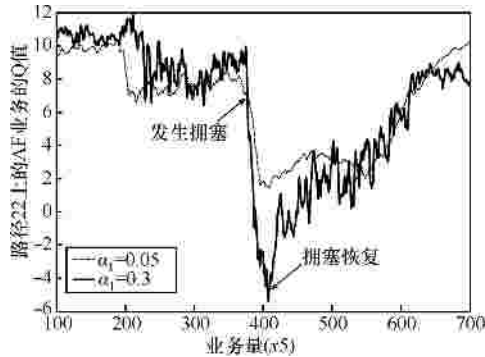
图 8 资源优化性能

4.2.5 学习率 a_1 、 a_2 对算法性能的影响

在本文中， a_1 为业务传输成功时的学习率， a_2 为业务传输失败时的学习率。学习率越大，则算法收敛速度越快，但抖动也越大，而且如果学习率过大，还有可能导致算法不收敛。分别选取 AF 业务在故障路径 17 和拥塞路径 22 上的 Q 值变化曲线说明学习率对算法性能的影响，如图 9 所示。



(a) AF 业务在故障路径上的 Q 值变化曲线



(b) AF 业务在拥塞路径上的 Q 值变化曲线

图 9 学习率 a_1 、 a_2 对算法性能的影响

在仿真进行到 1/4 时，故障发生，路径 17 中断，所以此时影响算法收敛速度的学习率为 a_2 ；当仿真进行到 3/4 时，故障结束，路径 17 恢复，所以影响算法收敛速度的学习率变为 a_1 。从图 9(a)中可以看出，在故障发生时，当学习率较大时，收敛速度较快，但有一定的抖动。从图 9(b)中可以看出，在拥塞发生的时候，较大的 a_1 对应曲线的收敛速度较快，但抖动较大。

5 结束语

本文提出了一种基于反馈机制的自愈算法。利用 Q 学习的反馈机制，降低了传统自愈机制对故障检测的依赖程度，在一定程度上提高了算法对于故障感知的灵敏度，从而提高了故障恢复率。算法利用多 QoS 约束的评价函数，对于不同类型业务查询不同的 Q 表格从而使其具备区分业务的能力，同时，通过对网络状态信息的反馈以及基于 Boltzmann-Gibbs 分布的路径选择策略也在一定程度提高了网络资源的利用率，达到了网络资源优化的目的。

参考文献：

[1] IETF RFC 3469. Framework for Multi-Protocol Label Switching (MPLS)-based Recovery[S]. 2003.

- [2] JORGE L, GOMES T. Survey of recovery schemes in MPLS networks[A]. 2006 International Conference on Dependability of Computer Systems[C]. Szklarska Poreba, Poland, 2006.
- [3] YIJUN X, MASON L G. Restoration strategies and spare capacity requirements in self-healing ATM networks[J]. IEEE/ACM Transactions on Networking, 1999, 7(1): 98-110.
- [4] CAVENDISH D, OHTA H, RAKOTORANTO H. Operation, administration, and maintenance in MPLS Networks[A]. IEEE Communications Magazine[C]. 2004. 91-99.
- [5] IETF RFC 4379. Detecting Multi-protocol Label Switched (MPLS) Data Plane Failures[S]. 2006.
- [6] IETF DRAFT. Bidirectional Forwarding Detection[S]. 2007.
- [7] IETF RFC 3469. Framework for Multi-Protocol Label Switching (MPLS)-based Recovery[S]. 2003.
- [8] GRERIN R, WILLIAMS D, ORDA A. QoS routing mechanisms and OSPF extensions[A]. 1997 Global Telecommunications Conference[C]. Phoenix, 1997.
- [9] KE Y, COPELAND G A. Scalability of routing advertisement for QoS routing in an IP network with guaranteed QoS[A]. 2000 Global Telecommunications Conference[C]. San Francisco, 2000.
- [10] KODIALAM M, LAKSHMAN T. Minimum interference routing with applications to MPLS traffic engineering[A]. Nineteenth Annual Joint Conference of the IEEE Computer and Communications Societies[C]. Tel Aviv, 2000.
- [11] SZETO W, BOUTABA R, IRAQI Y. Dynamic online routing algorithm for MPLS traffic engineering[A]. The 9th International Conference on Advanced Communication Technology[C]. Gangwon-Do, Korea, 2007.
- [12] BAGULA B, BOTHA M, KRZESINSKI A. Online traffic engineering: the least interference optimization algorithm[A]. 2004 IEEE International Conference on Communications[C]. Paris, France, 2004.
- [13] KAELBLING L P, LITTMAN M L, MOORE A W. Reinforcement learning: a survey[J]. Journal of Artificial Intelligence Research, 1996, 4(1): 237-285.
- [14] SUTTON R S, BARTO A G. Reinforcement Learning: An Introduction[M]. The MIT Press, Cambridge, MA, 1998.

作者简介：



赵季红 (1963-)，女，北京人，博士，西安交通大学教授、博士生导师，主要研究方向为宽带通信网、新一代网络的管理与控制。

曲桦 (1961-)，男，陕西杨凌人，博士，西安交通大学教授、博士生导师，主要研究方向为现代通信网和计算机网络体系结构。

陈文东 (1987-)，男，山西阳泉人，西安交通大学硕士生，主要研究方向为宽带通信网。