

TI-MFA: Keep Calm and Reroute Segments Fast

Klaus-Tycho Foerster Mahmoud Parham Marco Chiesa Stefan Schmid
IEEE Conference on Computer Communications Workshops

王锋 SA19006101

2019.11.29

问题描述

- 分段路由的出现是为了解决基于MPLS的流量工程解决方案的运营问题
- 本文的主要问题是关于快速重路由，静态地提前定义故障转移规则，可以无须调用控制平面或等待最短路径重新收敛
- 有较强鲁棒性的快速重路由计划，可以忍受多个链路的失效

当前进展与存在问题

- 现有的方法一般是采用TI-LFA方法来实现

故障转移规则需要被提前静态地配置，这种方案没有时间去重新计算路径，也没有时间将故障相关信息向上游或者下游传输。

只能够依赖于本地的相关信息，尤其是无法获取到下游可能出现的额外故障。

没有全局信息，定义故障转移规则的算法没有规定的情况下，则可能会产生前后矛盾的路由结果

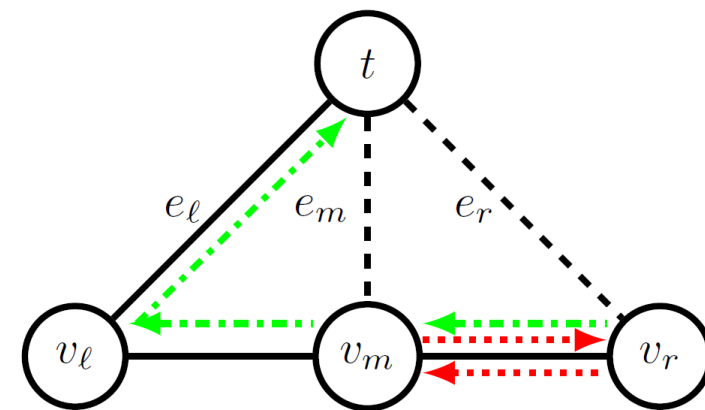


图 1 TI-LFA在多条链路失效时可能存在的问题

问题的提出与意义

- 首次探索单个故障之外分段路由的快速重路由，对于这个问题，本文对分段路由中的静态快速故障转移进行了系统的研究
- 关于分段路由中的快速故障转移能够和不能实现什么以及对其可能的权衡方式的见解
- 一个网络能够容忍多个链路同时故障，也意味着网络风险由链路组共同承担，那么它就更可能用于更大规模的网络中

思路

实现一个本地（预先计算好的）机制使得packet强制通过上述例子的第三条链路 e_l 这样可以
让packet成功到达目的地。但是这种做法的代价就是增加推送标签的数量，由此对图1进
行了扩展。对于图1中两条链路故障的情况下用图2所示的结构来替换掉 v_m 与 v_r 之间的链路。

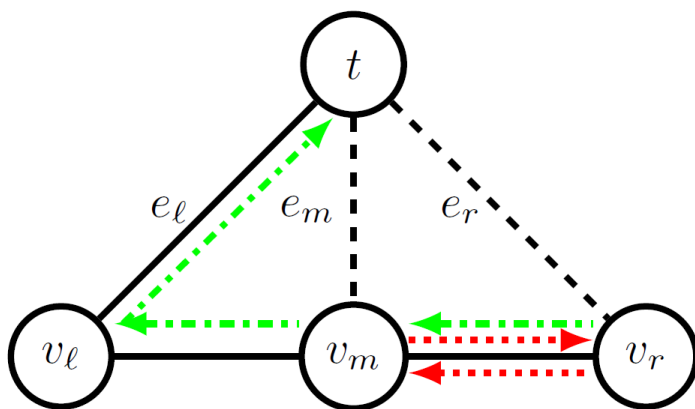


图 1 TI-LFA在多条链路失效时可能存在的问题

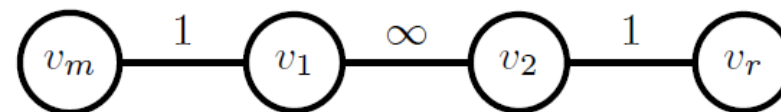


图 2 增加最少数量推送标签的结构

TI-MFA工作原理

从节点 v 的角度来描述：

1. 除了目的地 t 之外，刷新全部的标签栈
2. 根据packet报头中存储的所有的故障链路信息，在剩余部分网络 G' 中选择到目的地 t 最短的路径 P
3. 在packet的标签栈中添加分段：在路径 P 上给节点编号为 $v = v_1, v_2, \dots, v_x = t$ ，然后计算 P 上的索引最高的 v_i ，使得从 v 开始的最短路径在 G' (有故障链路) 和 G (无故障链路) 中是相同的，并且将其设置为标签栈的栈顶。如果计算的结果节点是 v ，将链路 $(v_1, v_2 = v_i)$ push到标签栈的栈顶，对于标签栈的第二项，将 v_i 作为起始节点重新开始，以此类推直到 $v_i = t$

理论证明

定理一：假设 G 是一个网络其中有 k 个链路发生故障，剩余的部分网络 G' 保持连通。TI-MFA成功路由到目的地。

证明：定理一的证明将通过嵌套的与故障次数有关的归纳讨论进行。

现在假设packet已经遇到了 $x-1$ 次故障，并且在其路径上遇到了第 x 次故障。进行以下两种情况的讨论：

1. $x = k$: G' (有故障链路) 和 (无故障链路) 中的最短路径路由再次相同。
2. $x > k$: 除非接下来遇到某个链路第 $x+1$ 次故障 (前 x 次故障不会再被遇到)，否则目的地是可达的。

然后，再次调用归纳构造：前提是已经遇到的故障不会再被遇到，即最终要么遇到了所有的故障，要么就成功到达目的地。但是，一旦遇到所有的故障，这意味着通过路径 P 目的地是可达的。

实验与实验分析

- 实验在Rocketfuel拓扑结构上运行，使用其提供的链路权重和延迟变体。
- 实验列举了所有TI-LFA和TI-LMA将遇到两个故障链路的故障情形。
- 实验过程丢弃了所有断开连接的实例。
- 实验过程用程序模拟了TI-LFA和TI-MFA的逐跳行为，并且记录：
 1. packet是否成功到达目的地
 2. 使用的最大标签栈的大小
 3. 所走路径的长度。
- 分别在是否刷新标签栈的情形下运行TI-LFA和TI-MFA，因此一共有四种算法。

实验与实验分析

- 成功率/可达性分析。两种TI-MFA变体实例在所有实例中都到达了目的地，而TI-LFA(无刷新)有大约 $\frac{1}{5}$ 的实验陷入了无限循环。实验结果如图3所示。(四种算法共进行超过500万次实验)。

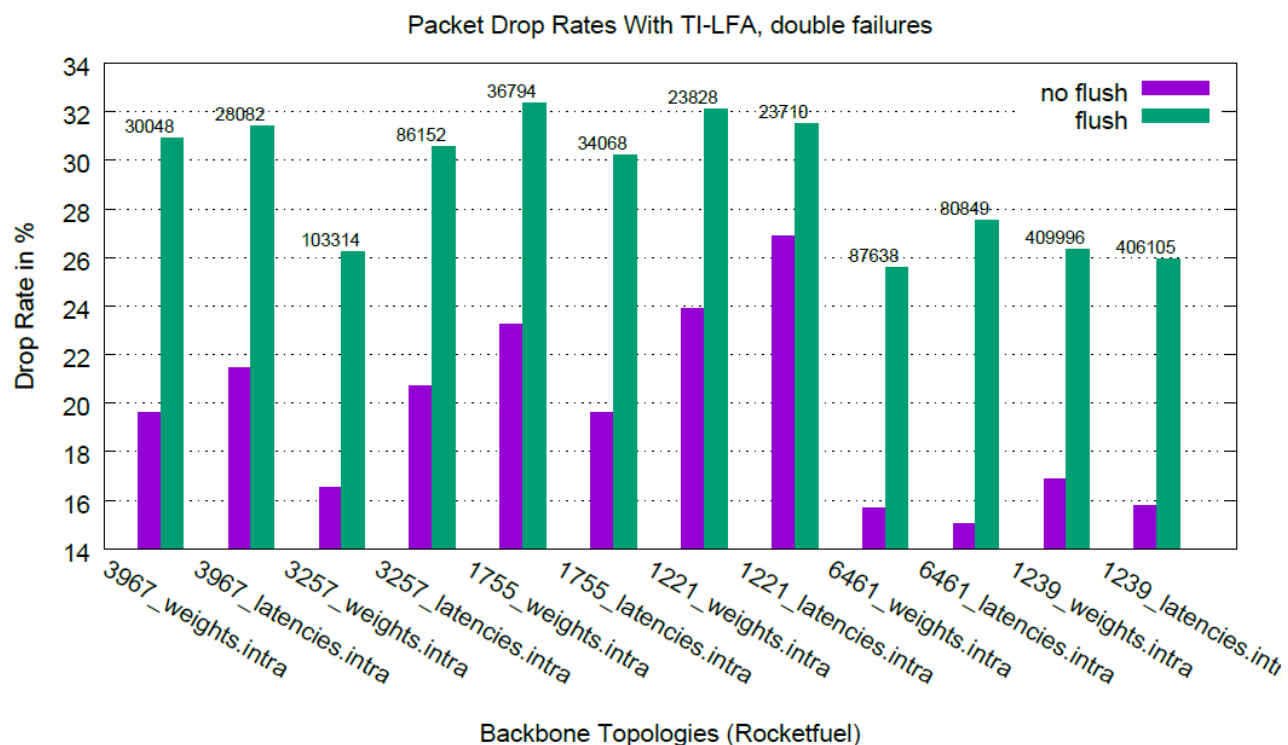


图3 TI-LFA失败率

实验与实验分析

- 最大标签栈大小分析。实验结果可以看出TI-MFA在标签栈大小方面的工作是有效的，并且可以预测在使用刷新的情形下， $2k+1$ 就足够处理 k 个链路故障的情形。

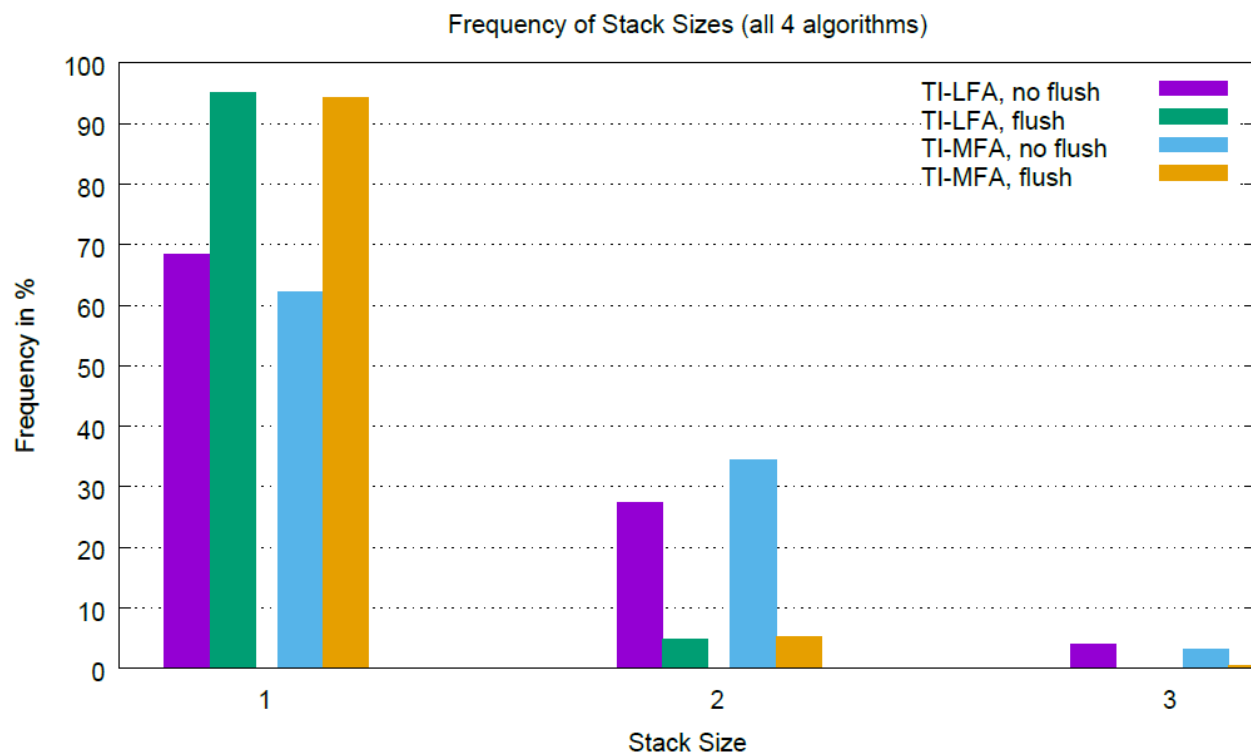


图 4 最大标签栈大小出现的频率

实验与实验分析

- 路径长度分析。图5为所有成功的实例的路径长度，延迟和权重成本函数。可以看出，带刷新的TI-MFA在所有拓扑中表现最好，而其他三个算法变体排名取决于特定的拓扑。

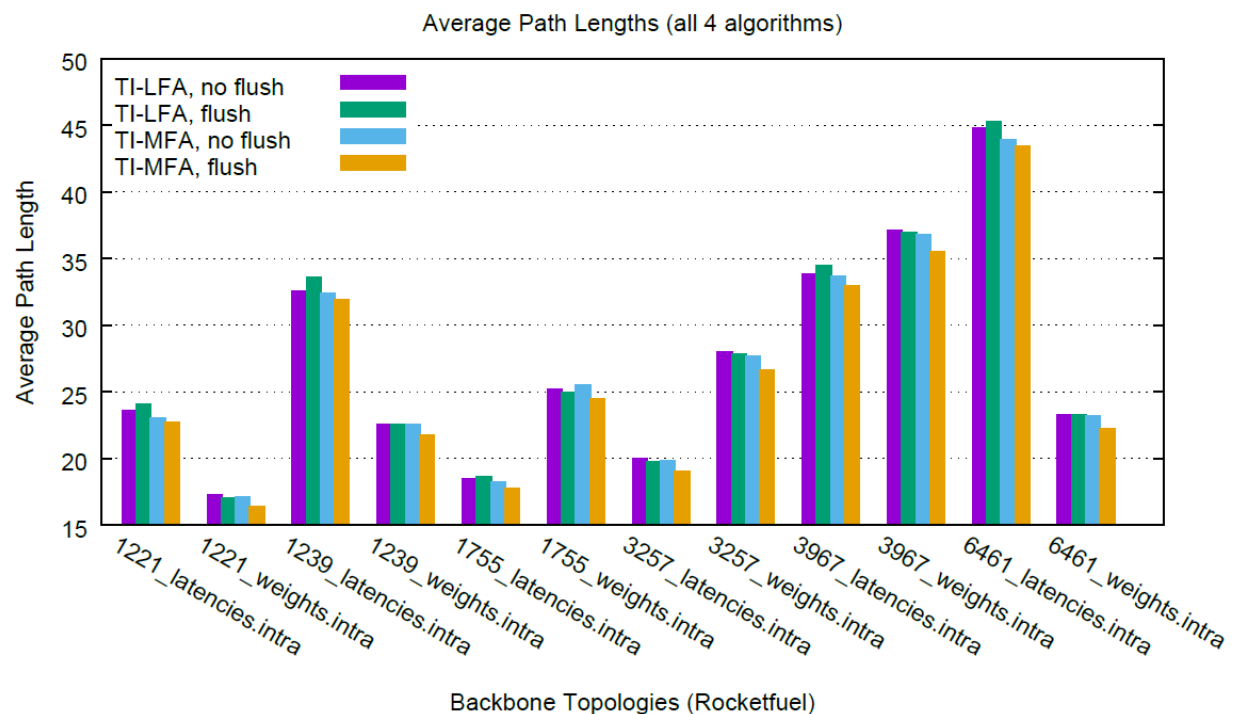


图 5 成功实验中的平均路径长度

贡献

1. 文中展示了现有的基于TI-LFA的分段路由快速故障转移解决方案在面对两个或更多故障时无法工作，此外 k 个故障的任何TI-LFA扩展都必须有 $2k+1$ 个栈的分段来提供支持。
2. 本文证明了，至少在原则上，基于文献中已知的非分段路由网络的不连通树的现有的快速重路由技术也可以在分段路由中进行仿真。但是，这种仿真的开销非常大。
3. 本文在故障转移的效率和鲁棒性之间找到了一个基本的折衷。特别是对于任何没有扩展的最少可以容忍两个故障的分段路由故障转移方案，即使在只出现了一个故障的情况下，也有可能被迫使用代价昂贵的路由。

结论

本文研究了多个链路故障且不调用控制平面的分段路由网络算法，以及算法的局限性。

本文提出的算法，即使在多个链路故障的情形下也能保证可达性。文中称其为TI-MFA，是TI-LFA的扩展，并且具有可证明的正确性保证。并且在使用Rocketfuel拓扑中进行了评估，与TI-LFA的对比展示了TI-MFA相较于现有方法的优点。此外，文中还研究了在故障情形下刷新标签栈的效果。，即移除中间目标来优化接下来的故障转移路径。并且在模拟中，TI-LFA的故障率在这个设置中几乎翻了一番。

未来工作

- 算法的最优性，最坏情况下的弹性，以及在所需分段数量上的开销等相关方面还需要进一步研究。
- 研究给定配置的弹性和其他属性。这方面，最近在MPLS网络上的一个结果也可以应用到分段网络中进行多项式时间假设分析：它可以有效地测试某个网络配置是否可达，是否在多次故障下符合策略。但是，这种测试的精确复杂性也是未来研究的主题。

THANK YOU!