**《互联网协议分析与设计》经典文献阅读报告**

**学生姓名：于鹄杰**

**学 号：201506021075**

**一．论文阅读**

1. **论文基本情况**

论文标题：A Comparison of Hard-state and Soft-state Signaling Protocols

发表会议/期刊: Ji P, Ge Z, Kurose J, et al. A comparison of hard-state and soft-state signaling protocols[C]// IEEE, 2003:251-262.

其他参考文献：

[1] Berger, L, Gan, D, Swallow, G, et al. RSVP Refresh Overhead Reduction Extensions[J]. 2001.

[2] Fenner W. Internet group management protocol[J]. Rfc, 2010, 1997.

[3] Clark D. The design philosophy of the DARPA internet protocols[C]// Symposium on Communications Architectures & Protocols. ACM, 1988:106-114.

[4] DEERING,S. E. Host Extensions for IP Multicasting[J]. Rfc, 1986.

[5] Deering S, Estrin D L, Farinacci D, et al. The PIM architecture for wide-area multicast routing[J]. Acm Sigcomm Computer Communication Review, 1996, 4(2):153-162.

[6] Estrin D, Handley M, Helmy A, et al. A dynamic bootstrap mechanism for rendezvous-based multicast routing[C]// INFOCOM '99. Eighteenth Joint Conference of the IEEE Computer and Communications Societies. Proceedings. IEEE. IEEE, 2002:1090-1098 vol.3.

[7] Fenner W. Internet group management protocol[J]. Rfc, 2010, 1997.

[8] Floyd S, Jacobson V, Mccanne S, et al. A reliable multicast framework for light-weight sessions and application level framing[C]// ACM SIGCOMM Conference. 1995:342-356.

[9] M. Handley, H. Schulzrinne, E. Schooler, and J. Rosenberg. Sip: Session initiation protocol, Mar. 1999. RFC 2543.

[10] P. Ji, Z. Ge, J. Kurose, and D. Towsley. A comparison of soft-state versus hard-state signaling. Technical report, University of Massachusetts at Amherst, 2003.

[11] Kazaa ﬁle sharing network, 2002. <http://www.cazaa.com/.>

[12] P. Pan and H. Schulzrinne. Staged refresh timers for RSVP. In 2nd Global Internet Conference, Phoenix, AZ, 1997.

[13] C. Partridge and S. Pink. An Implementation of the Revised Internet Stream Protocol (ST-2). Journal of Internetworking: Research and Experience, 3(1), March 1992.

[14] Q2931. ITU-T Recommendation.

[15] S. Raman and S. McCanne. A model, analysis, and protocol framework for soft state-based communication. In Proceeding of SICOMM, Boston, MA, 1999.

[16] P. Sharma, D. Estrin, S. Floyd, and V. Jacobson. Scalable timers for soft state protocols. In Proc. of IEEE INFOCOM, Kobe, Japan, 1997.

[17] C. Topolcic. Experimental Internet Stream Protocol: Version 2 (ST-II), October 1990. Internet RFC 1190.

[18] S. Zabele, M.Dorsch, Z. Ge, P. Ji, M. Keaton, J. Kurose, J. Shapiro, and D. Towsley. Sands: Specialized active networking for distributed simulation. In DARPA Active Networks Conference and Exposition (DANCE), San Francisco, California,USA, May 2002.

[19] L. Zhang, S. Deering, D. Estrin, S. Shenker, , and D. Zappala. RSVP: A new resource reservation protocol. IEEE Network, September 1993.

1. **论文摘要**

One of the key infrastructure components in all telecommunication networks, ranging from the telephone network, to VC-oriented data networks, to the Internet, is its signaling system. Two broad approaches towards signaling can be identiﬁed: so-called hard-state and soft-state approaches. Despite the fundamental importance of signaling, our understanding of these approaches - their pros and cons and the circumstances in which they might best be employed - is mostly anecdotal (and occasionally religious). In this paper, we compare and contrast a variety of signaling approaches ranging from a “pure” soft state, to soft-state approaches augmented with explicit state removal and/or reliable signaling, to a “pure” hard state approach. We develop an analytic model that allows us to quantify state inconsistency in single- and multiple-hop signaling scenarios, and the “cost” (both in terms of signaling overhead, and application-speciﬁc costs resulting from state inconsistency) associated with a given signaling approach and its parameters (e.g., state refresh and removal timers). Among the class of soft-state approaches, we ﬁnd that a soft-state approach coupled with explicit removal substantially improves the degree of state consistency while introducing little additional signaling message overhead. The addition of reliable explicit setup/update/removal allows the soft-state approach to achieve comparable (and sometimes better) consistency than that of the hard-state approach.

1. **研究动机(Motivation/Introduction)**

最根本的问题是理解已经发展成为包含在各种硬状态和软状态信令协议中的机制如何能够在给定情况下得到最好的使用，以及为什么 。

1. **相关工作(Related Work)**

**4.1还原两大类信令方法：硬态和软态方法（hardstate and soft-state approaches）**

软态方法的实质是状态持有者使用stateinstaller进行尽力而为的周期性状态安装/刷新，并使用state-removal-by-timeout。

而“硬状态”信号采取相反的方式进入软状态 - 除非通过从状态安装程序收到状态拆卸消息明确删除，否则安装状态将保持安装状态，因此可靠的（而不是尽力而为的）信令协议通常与硬状态协议相关联，硬态方法的本质是可靠和明确的安装和删除状态信息。

**4.2考虑五种不同的抽象信令协议的操作：**

**Puresoft-state**（纯软状态（SS））：在这种方式中，信令发送方向信令接收方发送包含状态安装或更新信息的触发消息，并启动状态刷新定时器（值为T）。当状态刷新定时器到期时，信令发送器发出包含最新信令状态信息的刷新消息，并重置刷新定时器。触发器和刷新消息以尽力而为（不可靠）的方式发送。

**Soft-state with Explicit Removal**（显式移除的软状态（SS + ER））：SS + ER与SS方法类似，但增加了明确的状态移除消息。当信令发送方移除状态时，发送方向承载明确状态移除信息的信令接收方发送尽力而为（不可靠）的信令消息。状态刷新和触发消息，以及状态超时定时器都像SS一样使用。

**Soft-State with Reliable Trigger**（具有可靠触发的软状态（SS + RT））：SS + RT与SS类似，有两个重要的附加功能。首先，触发消息在SS + RT中可靠传输。其次，SS + RT还采用通知机制，在该通知机制中，信令目的地通知信令发送方关于状态超时定时器到期的状态移除。

**Soft-State withReliable Trigger/Removal**（具有可靠触发/移除的软状态（SS + RTR））：除SS + RTR方法使用可靠消息来处理状态设置/更新以及状态移除外，SS + RTR与SS + RT方法类似。

**Hard-State (HS) approach**（硬状态（HS）方法）：在HS方法中，可靠的显式消息用于在信令接收器处建立，更新和移除状态。 刷新消息和软状态超时移除机制都不被采用。 任何硬状态协议的关键问题是去除信号接收器处的孤立状态。 由于硬状态协议不提供基于超时的状态移除，它必须依靠外部信号来检测它是否处于孤立状态。

**4.3考虑影响信令协议性能的五种因素：**

**Application-speciﬁc inconsistency cost**（应用程序特定的不一致成本）：这些是与信令发送方和接收方处于不一致状态相关的成本。

**Refresh timeout value**（刷新超时值）：刷新定时器的值越小，状态持有者的一致性状态越早被安装，因此状态不一致导致的应用程序特定成本越小。

**Soft-state timeout value**（软状态超时值）：由于此计时器旨在移除未刷新的状态，理想情况下，此值应尽可能小，以便在信号发送方离开时立即移除孤立状态。

**Signaling message loss**（信令消息丢失）：随着消息丢失的概率变得更高，我们预计信令发送方和接收方状态不一致的时间比例也会增加，因为消息要可靠地传递或者尽力而为的刷新消息需要更长的时间交付。

**Number of hops**（跳数）：随着跳数增加，所有节点处于不一致状态的时间比例也将增加。

**4.4在信号发送器/接收器的生命周期中可能发生的事件：**

**Signaling state setup**（信令状态设置）：当信令会话首先安装（初始化）其本地状态时，它将包含状态的信令消息发送给接收方。延迟一段时间后，信令消息到达远程接收器，使发送器和接收器都达到一致状态。

**Signaling state update**（信令状态更新）：发件人也可以更新其本地状态。 和状态设置一样，发送者然后在接收者处安装新的状态值。当发送者更新其本地状态时，发送者和接收者的状态将不一致，直到更新成功传播到接收者。

**Signaling state removal**（信号状态删除）：在生命周期结束时，发件人将删除其状态。此时，接收者的状态也应该被删除。 一旦发送者删除了它的状态，接收者的状态就会“陈旧”（不一致），直到它被删除。许多依赖于协议的机制（包括状态超时和显式移除消息）可以用于移除接收器状态。

**False signaling state removal**（虚假信号状态消除）：即使发件人仍处于维护状态，目标可能会错误地删除状态。这可能是各种协议相关事件的结果。例如，在软状态方法中，即使发送方仍处于维持状态，状态超时计时器也可以在接收方和移除状态过期。

1. **解决方案(Solution)**

**5.1实验方法：**

比较和对比了从“纯”软状态方法到增强了显式远端状态消除和可靠信号传递的软状态方法到“纯”硬状态做法。

定义了一系列与此频谱相关的通用协议，并开发了一个统一的参数化分析模型，使我们能够量化与给定信令协议相关的关键性能指标 - 状态安装程序的状态和 状态持有者不一致。

量化了与给定信令方法及其参数值（例如，状态刷新和移除超时间隔）相关联的“成本”（既涉及信令开销，也涉及由状态不一致导致的特定于应用的成本）。

**5.2建模分析：**

鉴于信令发送者和接收者生命周期中的这些事件，开发了一个马尔可夫模型，以捕捉这种行为。马尔科夫模型的状态定义如下。 每个状态由一对值（xs，xd）组成，其中xs和xd分别表示信令发送者和接收者的状态：

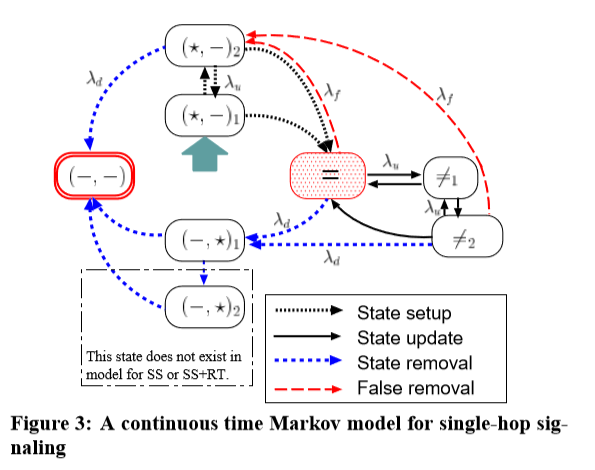
**马尔可夫状态（\*，-）：**捕获生命周期的初始阶段，当信号状态已经安装在发送端而不是接收端时。这是一个不一致的状态，因为发送者和接收者的状态值不匹配。

**马尔可夫状态（- ，\*）：**对应于发送者已经删除了状态但没有接收者的情况。这些状态也不一致。

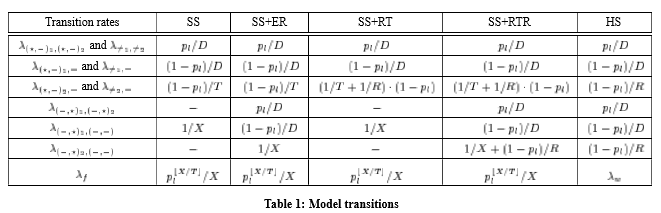
**马尔可夫状态=或！=：**当发送者和接收者具有一致的信令状态时。

**马尔可夫状态（- ，-）：**当信号状态从发送者和接收者都被移除时，系统进入吸收状态。

马尔可夫状态之间的转换在图3中用不同的线型表示，它们表示导致状态转换的不同事件（状态设置，状态更新，状态移除和错误移除）。



我们讨论了五种不同的信令方式。可以使用图3所示的模型对这些方法中的每一种进行建模，对于每种方法都有不同的转换率（有些情况下禁用转换）。 接下来是描述这些不同信令方法的模型转换。这些转换显示在模型图或表1中。



**软状态（SS）模型**：模型的初始状态（\*， - ）1对应于在发送者处创建新的信号状态。正如前面所讨论的，这会导致触发器消息被发送到接收器上安装状态。 频道延迟后，可能会发生两个事件之一。 首先，触发消息可以成功到达目的地。 该事件以概率（1-pl）发生，并且通过从状态（\*， - ）1到状态的速率（1-pl）/ D的转换来建模。第二种可能性是触发信息丢失。该事件以概率pl发生，并 且由从（\*， - ）1到（\*， - ）2与速率pl / D的转换表示。最终刷新消息将到达目的地。

**具有显式删除（SS + ER）模型的软状态**：回想一下，在SS + ER中，信令消息携带显式状态移除信息（除了状态超时机制）以移除信令状态。我们通过修改SS模型中的状态移除过程来模拟这种显式移除，如下所示。当马尔可夫链进入作为发送者状态移除的结果的状态（ - ，\*）1时，发送明确的状态移除消息。以概率（1-pl）和通道延迟后，此消息到达目的地并触发相应状态的移除。我们通过让马尔可夫链从速率（1-pl）/ D从（ - ，\*）1转变为吸收状态（ - ， - ）来对此进行建模。明确去除信息的丢失导致马尔可夫链从（ - ， \*）1过渡到（ - ，\*）2。 从那里，系统以速率1 / X转换到吸收状态（ - ， - ），捕获由状态超时计时器到期引起的状态移除。

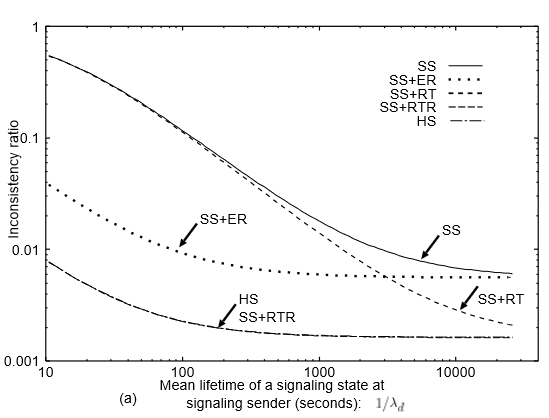
**具有可靠触发消息（SS + RT）模型的软状态**：用于SS + RT 的马尔可夫模型不同于用于SS的模型，因为当携带状态设置/ 更新信息的触发消息丢失时，成功的刷新消息或触发消息的成功重传可以使马尔可夫链状态或状态（\*， - ）2以速率（1 / T + 1 / R）·（1-pl） 表示状态=。

**具有可靠的触发/移除消息（SS + RTR）模型的软状态**：用于SS + RTR的马尔可夫模型不同于用于SS + RT的模型，因为当明确的移除消息丢失时，系统等待状态超时计时器到期或成功重传删除消息进入状态（ - ， - ）。因此从状态（ - ，\*）2到状态（ - ， - ）的转换率为1 / X +（1-pl）/R。

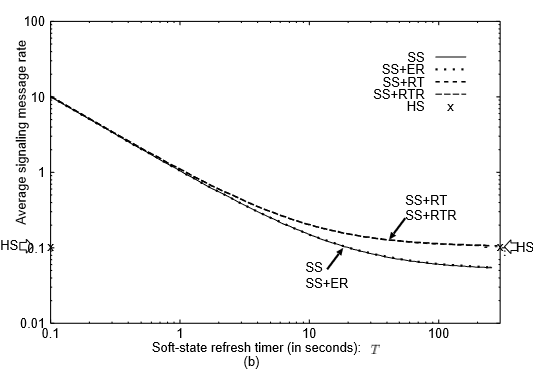
**硬状态（HS）模型：**HS模型类似于SS + RTR模型，不包括与刷新消息和状态超时定时器相关的转换速率。另外，正如所讨论的，HS方法必须依靠外部信号来从发送方故障中恢复。考虑到这种外部信号的相关成本是困难的，因为它取决于执行硬状态方法的故障检测的底层方案。

1. **实验结果(Evaluation)**

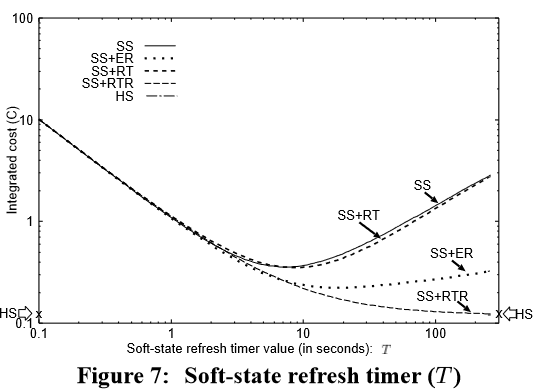
结果1：即使对于适度的丢失率（例如5％），可靠传输也明显提高了软状态协议的性能。



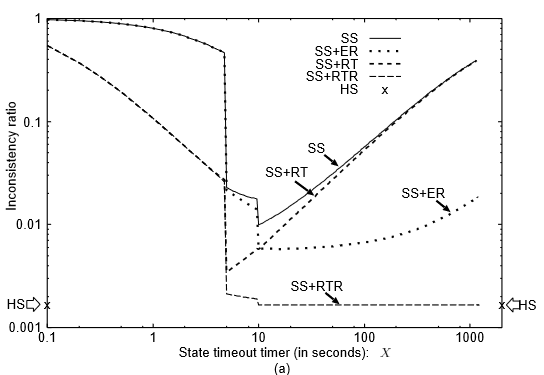
结果2：为了从损失中恢复过来，可靠传输的方法在环境中受到更长的传输延迟。



结果3：对于SS + RTR，优选更长的定时器值，并且当定时器足够大（大约100秒的量级）时，它提供了与硬状态方法相当的性能。



结果4：当状态超时定时器比刷新定时器短时，所有基于软状态的方法执行得不好，因为刷新消息到达太迟而不能保持信令接收器处的信令状态。



1. **结论(Conclusion)**

（1）在软状态方法中，软状态方法与显式移除相结合显著提高了状态一致性，同时引入了很少的附加信令消息开销。

（2）增加可靠的显式设置/更新/删除进一步允许软状态方法实现与硬状态方法相比（并且有时更好）的一致性。

1. **自己的思考**

**8.1 阅读本文的收获和感想**

之前我只是听说过hardstate and soft-state approaches这个名词，并不了解它们的运行机制是什么，在学习这篇论文的过程中，虽然这篇论文对我来说非常难懂，但是通过查阅大量的资料并投入了大量的时间来研究之后，我基本理解了已经发展成为包含在各种硬状态和软状态信令协议中的机制如何能够在给定情况下得到最好的使用，以及为什么，也算是达到了理解这篇文章的基本目标。

**8.2 本文提出方案存在的缺陷/可能的改进方法/实际的解决方案**

本篇论文的研究主要是在单跳系统下研究各种硬状态和软状态信令协议中的机制的运行原理和规律，在多跳系统下的研究涉猎较为简单，所以可能存在的缺陷就是单跳系统下发现的原理和规律并不适用于多跳系统，需要我们针对多跳系统的特点来制定对应的模型。

文中提出了一种针对于多跳的实验方案模型，但是是借用了单跳实验的一些条件，如果可能的话，可以为多跳系统的马尔可夫模型多制定几个马尔可夫状态，再根据硬态和软态的不同状态制定对应的模型。