本科生毕业论文（设计）

题目：基于强化学习的
360 度视频传输算法研究

姓　　名	官展伊
学　　号	18308048
院　　系	电子与通信工程学院
专　　业	通信工程
指导教师	蔡科超 助理教授

2022 年 3 月 15 日

基于强化学习的
360 度视频传输算法研究

Research on 360-degree video transmission algorithm with reinforcement learning

姓　　名	官展伊
学　　号	18308048
院　　系	电子与通信工程学院
专　　业	通信工程
指导教师	蔡科超 助理教授

2022 年 3 月 15 日

表一　毕业论文开题报告

论文 (设计) 题目:
基于强化学习的 360 度视频传输算法研究

(简述选题的目的、思路、方法、相关支持条件及进度安排等)

选题目的:

本次选题……

思路:

……

方法:

……

相关支持条件:

……

进度安排:

……

Student Signature:　　年　月　日

指导教师意见：

……

1. 同意开题　2. 修改后开题　3. 重新开题

指导教师签名:　　年　月　日

表三　毕业论文（设计）答辩情况登记表

答辩人	官展伊	**专业**	通信工程
论文 (设计) 题目	基于强化学习的 360 度视频传输算法研究		
答辩小组成员			

答辩记录：

记录人签名:　　年　月　日

学术诚信声明

本人郑重声明：所呈交的毕业论文（设计），是本人在导师的指导下，独立进行研究工作所取得的成果。除文中已经注明引用的内容外，本论文（设计）不包含任何其他个人或集体已经发表或撰写过的作品成果。对本论文（设计）的研究做出重要贡献的个人和集体，均已在文中以明确方式标明。本论文（设计）的知识产权归属于培养单位。本人完全意识到本声明的法律结果由本人承担。

作者签名:

日　　期：　　年　　月　　日

论文题目： 基于强化学习的 360 度视频传输算法研究
专　　业： 通信工程
姓　　名： 官展伊
学　　号： 18308048
指导教师： 蔡科超 助理教授

摘要

360 度全景视频以其身临其境的体验而备受关注。与具有相同的分辨率的传统视频流不同，它通常的带宽消耗是传统视频的 4-6 倍。不过用户并不能一次把 360 度视频全部内容尽收眼底，而是每次只能看到 360 个场景中的一部分（大约 20%）。因此，如果我们能够准确预测用户的视场（Field of View，FoV）的移动，传输 360 全景视频对应的部分，就足以满足用户需求。在实践中，为了减少预测用户 FoV 失败的概率，我们通常传输比 FoV 更大的部分。传输的部分越大，预测成功的概率就越高。然而，这将导致越低的传输成功概率。我们的目标是选择适当的交付部分，以最大限度地提高系统吞吐量。这可以表述为多臂老虎机（Multi-armed Bandit）问题，其中每个臂代表传输的速率。多臂老虎机问题一个常用的解决方法是汤普森采样（Thompson Sampling）。目前有文章提出的两层反馈的汤普森采样能够更加有效的解决 360 度全景视频的传输。考虑到现实世界中传输信道容量具有时变性。本文基于具有两层反馈的汤普森采样，提出了包括基于周期置零、滑动窗口和具有折扣系数的滑动窗口等新的汤普森采样算法，并通过仿真实验证明了本文提出的算法在时变信道中表现的更加优秀。

关键词： 强化学习；MAB 问题；汤普森采样；时变信道

Title: Research on 360-degree video transmission algorithm
Major: Communication Engineering
Name: Zhanyi Guan
Student ID: 18308048
Supervisor: Assistant Prof. Kechao CAI

ABSTRACT

360-degree panoramic video has gained much attention for its immersive experience. Unlike traditional video streaming with the same resolution, it typically consumes 4-6 times more bandwidth than traditional video. However, users cannot take in all of the 360-degree video at once, but can only see a fraction (about 20%) of the 360 scenes at a time. Therefore, if we can accurately predict the movement of the user's Field of View (FoV) and transmit the corresponding part of the 360 panoramic video, it is sufficient to meet the user's demand. In practice, to reduce the probability of failure in predicting the user's FoV, we usually transmit a larger portion than the FoV. The larger the transmitted part, the higher the probability of prediction success. However, this will result in a lower probability of transmission success. Our goal is to select the appropriate delivered fraction to maximize the system throughput. This can be formulated as a Multi-armed Bandit (MAB) problem, where each arm represents the rate of transmission. A common solution to the Multi-armed Bandit problem is Thompson Sampling. The two-level feedback Thompson Sampling proposed in the current paper can solve the transmission of 360-degree panoramic video more effectively. Considering the time-varying transmission channel capacity in the real world. In this paper, we propose a new Thompson sampling algorithm based on Thompson sampling with two levels of feedback, including period-based zeroing, sliding window and sliding window with discount factor, and demonstrate through simulation experiments that the proposed algorithm performs better in time-varying channels.

Keywords: Reinforcement learning, MAB problem, Thompson Sampling, time-varying channel

目录

第 1 章 绪论

1.1 选题背景与意义

360° 视频流内容允许用户在观看视频时在多个方向上改变她/他的观看方向，从而获得比观看具有固定观看方向的传统视频内容更身临其境的体验。此类视频内容可以使用不同的设备观看，从智能手机和台式电脑到特殊的头戴式显示器(HMD)，如 Oculus Rift、三星 Gear VR、HTC Vive 等。使用 HMD（Human Mounted Display）观看此类内容时，可以通过头部移动来改变观察方向。在智能手机和平板电脑上，可以通过触摸交互或通过内置传感器移动设备来改变观看方向。在台式电脑上，鼠标或键盘可用于与全向视频交互。360° 视频流能够提供全景视图，让用户拥有身临其境的体验，现在在主要的视频共享网站和社交媒体渠道上越来越流行。近些年来，随着“元宇宙”等概念越来越热门以及头戴式设备的深入研究使得价格不断下降，“虚拟现实”也越来越热门。“虚拟现实”中也有相当一部分比例的内容离不开 360 度全景视频的支持。如此，可以想象 360 度全景视频将在未来大放光彩。然而 360° 视频流的传输具有挑战相当大的挑战性。首先，由于全景的性质，在相同的感知质量下，360° 视频比传统视频大得多，一般为 4 倍到 6 倍大小关系[1]。因此与普通视频相比，360° 视频的传输消耗的带宽要高得多，这在无线和移动网络中尤其突出。其次，流式 360° 视频为移动终端设备引入了更高的计算和能量开销，这些设备的 CPU、GPU、存储和电池容量通常是有限。传统的传输方法是整体传输 360° 视频，这不仅仅带来了高额的带宽开销、还导致了计算和能量消耗，更可能因为用户的设备无法支撑起这些开销而引起较差的用户体验。另一方面，正如1-1所示，用户在在观看 360° 视频时，用户观看的是整个球形图像的有限部分，这通常由用户的视场（FoV）（也被称为视口）决定。如果我们能够准确预测用户的运动，传输满足用户要求的视频相关部分，就能在显著降低网络带宽消耗和其他无谓的消耗，同时也保证了用户体验质量。为此，基于切片的方法被提出来用于 360° 视频流传输，它将每个全景帧分为较小尺寸的非重叠矩形区域，称为切片。一般来说，基于切片的流媒体利用了传输效率和用户体验之间的权衡。一方面，只有一个切片子集被传输，这可以大大减少带宽消耗；另一方面，这个子集不一定覆盖实际的用户视野，因此用户体验质量受到切片选择的极大影响。

在实践中，我们通常提供一个大于 FoV 的部分来容忍运动预测误差。为了让用户能够成功地查看他/她想要的内容，该部分应该成功传输并覆盖用户的视野。

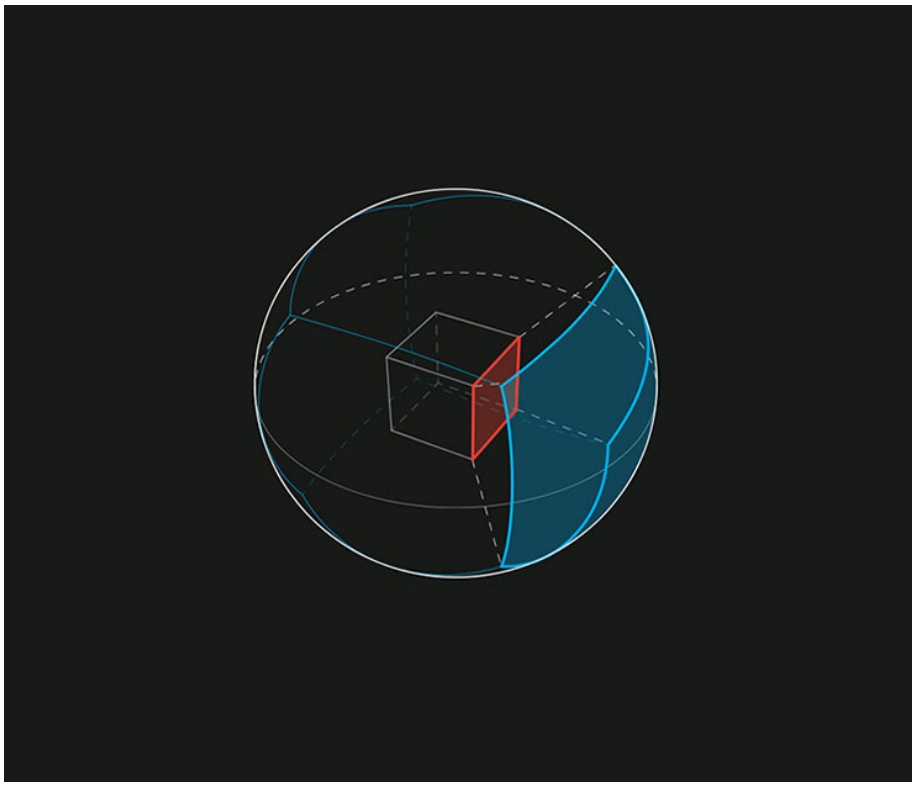

图 1-1 360 度视频原理

直观地说，传送切片部分越大，对运动预测误差的容忍度就越高，无线传输成功的几率就越低传输。传送切片部分越小，对运动预测误差的容忍度就越高，无线传输成功的几率就越低传输因此。因此我们的问题是如何在每次选择适当部分进行传输，以最大化系统吞吐量。

1.2 国内外研究现状和相关工作

近年来，已有不少针对 360 度视频流的传输算法研究工作。Xing Liu 等[2] 提出的研究使用跨学科方法优化 360° 视频流。他们从系统的角度出发，使用多种方法来优化 360 度视频流的传输。Harsh Gupta 等人[3] 考虑了一个具有两级反馈的多臂老虎机问题，将其应用于全景视频流。论文开发了 KL-UCB 算法的一个重要变体用来解决多臂老虎机问题。J. Chen 等人[4] 考虑了全景视频流的自适应速率选择问题，并将其表述为具有两级反馈信息的多臂老虎机问题，我们提出了一种改进的 Thompson 采样算法，该算法有效地利用了两级反馈信息，并且证明了它的性能比单级反馈信息的性能要好得多。Tang M 等人[5] 提出了一种基于时空域比特率自适应的 360 度视频流模型。

1.3 本文的研究内容与主要工作

我们把每个全景视频划分为一系列切片，其中每个切片具有相同持续时间的固定数量的块。在每个时隙中，可以选择块的子集进行传输。我们可以把选择不同的视频块对应为选择不同的速率，选择的切片越大，那么选择的传输速率也越大。如此，速率越大，传输成功可能性越低，但是覆盖到用户的 FoV 的可能性越大。如何合理选择传输速度，使得 360 度视频传输系统的吞吐量最大呢？我们考虑使用强化学习的方法来解决这个问题。本文将 360 度全景视频速率选择问题建模为一个两级反馈的多臂老虎机问题（MAB）。MAB 问题描述了一个玩家选择多个老虎机（Arm）的决策（Action）并从所选老虎机获得收益（Reward）的过程。在 360 度视频速率选择问题中，Arm 对应不同的速率。Action 是选择哪一个速率传输 360 度全景视频块。Reward 对应传输系统的一段时间内累积吞吐量。两级反馈分别是是否覆盖到用户的 FoV，是否传输成功。通过多臂老虎机的解决算法选出最合适的速度，使得累积吞吐量和最佳吞吐量之间的差距变小，即最小化遗憾（Regret）。本文进一步考虑了周期性变化的信道上传输概率发生变化的情况，提出了周期性重置 Thompson Sampling 算法、基于折扣窗口的 Thompson Sampling 算法、基于滑动窗口的 Thompson Sampling 算法和基于折扣滑动窗口的 Thompson Sampling 算法。并且通过模拟数据验证这五种算法的性能。

1.4 本文的论文结构与章节安排

本文共分为六章。第一章是绪论，描述了选题的背景和研究的意义，同时简单说明了国内外在 360° 视频传输的研究进展，然后说明了本文的研究内容。第二章详细描述了国内外在 360° 视频传输方面的相关研究，同时指出他们的研究在时变信道下的缺陷。第三章对 360° 视频传输建模为多臂老虎机问题并且提出基于两层反馈的汤普森采样算法进行改进来解决这个问题。第四章基于两层反馈的汤普森采样算法设计了若干算法。第五章使用模拟数据对这些算法实际效果进行测试并证明算法的有效性。第六章对全文进行总结。

第 2 章 相关工作

简述已有的 360 度视频传输算法和已有的基于 MAB 的 360 度视频算法，同时提出基于 MAB 的 360 度视频算法的缺陷。

2.1 已有的 360 度视频传输算法

当前已有的 360 度视频传输算法可以分为以下三种。

第一种是使用系统工程的方法优化 360 度视频传输。Xing Liu 等[2] 提出的研究使用跨学科方法优化 360° 视频流。包括：（1）创造性地将视频编码社区开发的现有技术应用于新环境。（2）使用大数据和众包来增加流算法的智能；（3）利用跨层知识来促进多个网络上的内容分发。（4）在优化广播方和观众方的 360° 实时视频流方面提供了新的见解。（5）将创新集成到 Sperke 中，这是一个具有各种系统级优化的整体系统。总之，这是一种全方位优化 360 度视频传输的方法。

第二种是对历史数据进行大规模训练后试图对 FoV 进行预测。训练的数据有当前用户的头部运动轨迹、当前播放的视频的其他用户的头部运动轨迹和视频内容。相关的研究有：Zhang 等人在[1] 中提出了一种深度强化学习（DRL）算法，根据预测的 FoV 和带宽学习选择最优的速率进行传输。Xie 等人在[6] 中提出了一种算法，通过比较用户和不同类别的其他用户的历史 FoV 来识别用户的类别，从而最大化用户的视频质量。这种类型的算法在实际表现上效果不错，但是最大的问题是需要大量数据进行训练。

第三种是对传输系统建模为多臂老虎机问题（MAB），然后采用针对 MAB 问题的方法进行解决，并且针对 360 度视频传输的特点对传统解决方法进行优化，比如采用两层反馈而不是单反馈。更详细的论述见下一小节。

2.2 已有的基于 MAB 的 360 度视频传输算法及缺陷

MAB 的解决方法有很多，比如 A-B test、epsilon-greedy 算法、UCB 算法、Thompson Sampling 算法等。目前已有的研究中有对 UCB 算法进行改进的解决方法。Harsh Gupta 等人在[7] 中提出两层反馈的 MAB 模型，开发了 KL-UCB 算法的一个重要变体，该算法有效地利用了两级反馈。论文进一步证明了它渐近匹配基本下界，这意味着它的渐近最优性。并且论文的实验结果表明，与经典的单反馈 KL-UCB 算

法相比，该算法具有更好的性能。

在实际效果测试中，Thompson Sampling 算法的改进版本表现出来的性能比 UCB 算法更好。UCB 算法的改进版本由 J. Chen 等人[8] 提出，该算法有效地利用了两级反馈信息，并且证明了它的性能比单级反馈信息的性能要好得多。但是该算法未能考虑到信道容量的复杂性，测试只在单一不变的信道容量下进行。本论文进一步提出了在周期性变化的信道容量下，可以采用置零或者带有折扣系数的滑动窗口的方法来优化双反馈 Thompson Sampling 算法的性能。

2.3 本章小结

360 度视频传输算法大致分为三种，第一种是使用系统方法进行优化。第二种方法使用大量数据进行训练，从而提高预测 FoV 的准确率。第三种基于 MAB 模型。在已有的基于 MAB 的 360 度视频传输算法研究中，表现优异的双反馈 Thompson Sampling 的信道条件过于单一，未能模拟出真实信道。因此本论文提出针对周期性变化的信道的双反馈 Thompson Sampling 的改进，包括置零或者带有折扣系数的滑动窗口的方法等方法。

第 3 章 360 度视频传输建模

本论文把 360 度视频传输问题建模为一个多臂老虎机问题（MAB），并且使用了两层反馈来提高模型的效果。我们考虑单个用户通过无线信道从接入点下载全景视频。我们假设系统以时隙方式运行。在每个时隙，用户只能看到整个全景场景的一部分（通常为 20%），即视野（FoV）。如果用户的头部运动可以准确地预测，这样就足以提供 20% 的全景图像，这大大降低了无线带宽消耗。不幸的是，很难准确预测用户的运动。因此，我们通常提供比 FoV 更大的部分来克服不准确的用户运动预测，但是增加传输的部分会使传输成功的可能性降低。因此我们需要正确的选择恰当的传输部分来最大化传输的吞吐量。

3.1 视频传输的两层反馈

在视频传输过程中，让 $R(t)$ 表示将在时隙 t 中通过无线信道传输的全景图像部分的大小，因此我们称 $R(t)$ 为 t 时隙的选择速度。我们假定 $R(t)$ 只能从集合 $\mathcal{R} = \{r_1, r_2, \ldots, r_N\}$ 中选择，其中 $0 < r_1 < r_2 < \cdots < r_N$。$r_1$ 和 r_N 分别表示 FoV 的大小和整个全景图片的大小。

第一个反馈是传输的全景视频块能否覆盖到用户的视野。我们使用 $X_n(t) = 1$ 标记 $R(t) = r_n$ 足够的大以至于在时隙 t 被传输的块能够覆盖到用户的视野。反之 $X_n(t) = 0$ 则不能覆盖。让 $\alpha_n \triangleq \Pr\{X_n(t) = 1\}$ 做为预测成功率，显而易见 $0 < \alpha_1 < \alpha_2 < \cdots < \alpha_N$（因为传输的块越大，预测成功率越高）。在这里，由于用户的设备自动记录用户的当前位置（偏航、俯仰、滚动）并发送回 AP（Access Point，接入点），因此 AP 在时隙 t 时，不管传输失败还是成功，都知道 X_t 的结果。

第二个反馈是传输是否成功。我们使用 C_t 来捕捉用户在时隙 t 时的信道衰落，假定其随时间独立且同相同的分布。我们并不能知道每一个时隙开始时的信道速率。如果所选择的传输速率并没有大于信道速率，即 $R_t \leq C_t$，那无线传输在时隙 t 将会成功，否则失败。我们使用 $Y_n(t) = 1$ 标记在时隙 t 通过选择速率 r_t 传输成功，$Y_n(t) = 1$ 则相反。让 $\beta_n \triangleq \Pr\{Y_n(t) = 1\}$ 表示传输成功概率。注意选择的速率越大，传输成功概率越低。因此，我们有 $\beta_1 > \beta_2 > \cdots > \beta_N$。

3.2 两层 MAB 模型

在本文中，我们将交付部分选择问题描述为一个随机多臂老虎机（MAB）问题。MAB 问题描述了一个玩家选择多个老虎机（Arm）的决策（Action）并从所选老虎机获得收益（Reward）的过程。在 360 度视频速率选择问题中，其中全景场景的不同交付部分，即不同的传输速率对应于不同的手臂（ARM）。Action 是选择哪一个速率传输 360 度全景视频块。Reward 对应传输系统的一段时间内累积吞吐量。我们的最终目标是在有限的时间范围内最小化遗憾（即，最佳累积吞吐量与算法下累积吞吐量之间的差距）。

J. Chen 等人[8] 论述了两层反馈的 MAB 模型与传统的单反馈 MAB 模型的不同，并且证明了两层反馈的 MAB 模型的在 360 度视频传输问题中比单反馈 MAB 模型表现更优秀。我们的问题是两层反馈的 MAB 模型在容量发生周期性变化的信道上表现不是那么优秀。当信道容量发生改变时，两层反馈的 MAB 模型表现很迟钝，需要非常长的时隙才能输出正确的结果，即最佳传输速率。为此，我们基于两层反馈的 MAB 模型提出了多种改进算法，以便在信道容量发生改变时，能够更快适应信道并选择出更好的手臂，从而获得最小化遗憾。在本文中，AP 需要对选

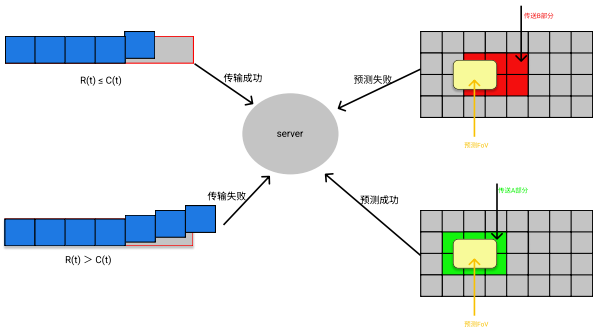

图 3-1 两层反馈示意图

择的速率做出决定，以最大化系统吞吐量。如果用户的预测成功概率和传输成功概率（即 $\{\alpha_n, \beta_n, n = 1, 2, \ldots, N\}$）都知道，则可通过解决以下优化问题来实现：

$$n^* \in \underset{n=1,2,\ldots,N}{\arg\max}\, r_n \alpha_n \beta_n,$$

其中 $\alpha_n \triangleq \Pr\{X_n(t) = 1\}$ 为预测成功率，$\beta_n \triangleq \Pr\{Y_n(t) = 1\}$ 为传输成功概率，r_n 表示全景图像传输部分的不同大小。然而预测和传输概率都是未知的，因为它们取决于许多因素，如用户行为、全景视频内容和无线环境。这要求算法不仅要学习这些统计数据，还要选择迄今为止的最佳速度。让 $I(t) \in \{1, 2, \ldots, N\}$ 表示 t 时

刻所选择的速度的下标，我们的目的是设计一种学习算法，在一定的正整数时隙内实现最大的系统吞吐量。这相当于最小化遗憾，即累积吞吐量和最佳吞吐量之间的差距：

$$\mathrm{Reg}(T) \triangleq \mathrm{T}\, r_{n^*}\alpha_{n^*}\beta_{n^*} - \mathbb{E}\left[\sum_{t=1}^{T} r_{I(t)}X_{I(t)}(t)Y_{I(t)}(t)\right].$$

如果在一个比较长的时间内遗憾过大，那就表明我们没有没有选择到最佳的传输速度。对用户而言，这往往意味着 360 度视频的传输延迟等，他/她的用户体验就大打折扣了。因此我们必须要达到一个次线性的遗憾才能够使得用户的体验最佳。

3.3 本章小结

本章对 360 度全景视频传输建模为两层反馈的 MAB 模型。在 360 度视频速率选择问题中，Arm 对应不同的速率。Action 是选择哪一个速率传输 360 度全景视频块。Reward 对应传输系统的一段时间内累积吞吐量。我们也提出了两层反馈的 MAB 模型在周期信道上的局限性。我们的最终目标是开发一种算法，可以在时变带宽下快速确定一段时间内的最佳交付部分。下一章中详细描述如何基于两层反馈的 MAB 模型来改进获得表现更优秀的算法。

第 4 章 算法设计

MAB 模型有很多解决方法，比如 Thompson Sampling 算法、UCB 算法等。在我们的测试中 Thompson Sampling 算法的实际效果更好，因此我们在选择基于 Thompson Sampling 算法进行改进从而问题。在 J. Chen 等人[8] 的研究中证明了两层反馈的 Thompson Sampling 的算法比单反馈的效果好很多。在这片文章中我们考虑到更加实际的问题，对于个人设备来说，信道容量的大小并不是不变的。造成这个情况有很多原因，包括不同时间段局域网内用户数量不同等。因此本篇文章基于两层反馈的 Thompson Sampling 算法设计了能快速适应周期变化的信道从而最大限度提高传输系统的吞吐量的多个算法。

在我们的设计算法中，$S_n^{(1)}(t)$ 表示预测成功的次数，$F_n^{(1)}(t)$ 表示预测失败的次数，$S_n^{(2)}(t)$ 表示传输成功的次数，$F_n^{(2)}(t)$ 表示传输失败的次数。这就是 360 度全景视频传输的两级反馈。我们把这两个反馈当作两个独立的反馈，每一个手臂（即传输速率）都分别维护预测和传输两个结果。

4.1 周期性重置 Thompson Sampling 算法

在周期性重置 Thompson Sampling 算法中，每当一个新的周期开始的时候，$S_n^{(1)}$、$F_n^{(1)}$、$S_n^{(2)}$、$F_n^{(2)}$ 都被置零。这个算法的思路很简单，每次周期改变时就把每一个手臂累积的成功次数和失败次数置 0。我们根据 Beta 函数的性质可以知道 Beta 函数对于最佳输出的改变并不敏感，改变最佳输出所需要的时间远大于建立最佳输出所需要的时间。因此置零算法可以称得上是一个不错的算法，但是问题是我们并不知道周期什么时候发生改变，所以在实际表现中可能不是非常如意。

由4-1可知周期性重置 Thompson Sampling 算法重点在于每次周期结束后都将所有的，$S_n^{(1)}$、$F_n^{(1)}$、$S_n^{(2)}$、$F_n^{(2)}$ 置零。

算法 4.1: 周期性重置 Thompson Sampling 算法

对于每个 $t = 1, 2, ..., T$ 进行
  如果 周期开始 则
    对于每个 *rate* $r_n, n = 1, 2, ..., N$ 进行
      //重置所有速度累积的传输成功和传输失败次数
      $S_n^{(1)} \leftarrow 0$ and $F_n^{(1)} \leftarrow 0$
  对于每个 *rate* $r_n, n = 1, 2, ..., N$ 进行
    画出 $\alpha_n(t) \sim \text{Beta}\left(S_n^{(1)} + 1, F_n^{(1)} + 1\right)^2$
    画出 $\beta_n(t) \sim \text{Beta}\left(S_n^{(2)} + 1, F_n^{(2)} + 1\right)^2$
  选择满足下列条件的 rate $r_{I(t)}$

$$I(t) = \underset{n=1,2,...,N}{\arg\max}\, r_n \alpha_n(t) \cdot \beta_n(t)$$

  观察随机预测结果 $X_{I(t)}(t)$ 和随机传输结果 $Y_{I(t)}(t)$
  如果 $X_{I(t)}(t) = 1$ 则
    $S_n^{(1)} \leftarrow S_n^{(1)} + 1$
  否则
    $F_n^{(1)} \leftarrow F_n^{(1)} + 1$
  如果 $Y_{I(t)}(t) = 1$ 则
    $S_n^{(2)} \leftarrow S_n^{(2)} + 1$
  否则
    $F_n^{(2)} \leftarrow F_n^{(2)} + 1$

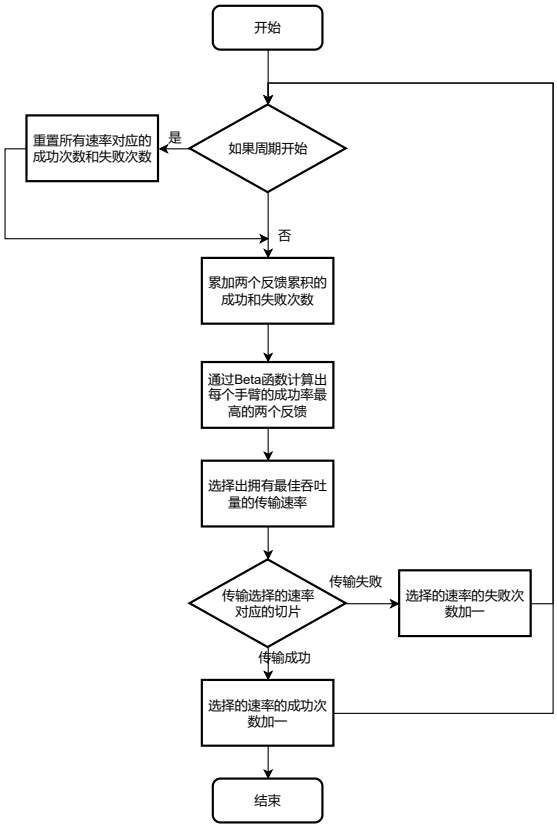

图 4-1 周期性重置 Thompson Sampling 算法流程图

4.2 基于折扣的 Thompson Sampling 算法

在一个周期改变的信道中，我们可以想象越是时间久的数据是对当前的速度选择决策越是不重要，那么我们可以使用折扣系数来对不同的时间的成功次数和失败次数赋予不同的权重。的在基于折扣的 Thompson Sampling 算法中，一个臂所对应的成功次数或者失败次数并不是简单的相加，而是使用折扣系数对累积的数据赋予不同的权重。因此在算法中我们使用一个数组存储所有的数据。越早的数据权重越小，越近的数据权重越大。折扣系数范围在 0 1 之间。每一次实验结果以元组 $\{1,2\}$ 或 $\{0,1\}$ 方式存储，其中前者表示成功，后者表示失败。比如现在队列大小为 5，队列存储的数据为 $\{1,0\},\{0,1\},\{1,0\},\{0,1\},\{1,0\}$，折扣系数为 0.9，那么输出的结果是 $\{1*0.9^5+0*0.9^4+1*0.9^3+0*0.9^2+1*0.9^1, 0*0.9^5+1*0.9^4+0*0.9^3+1*0.9^2+0*0.9^1\} = \{2.219, 1.466\}$。

算法 4.2: 基于折扣的 Thompson Sampling 算法

对于每个 *rate* $r_n, n = 1,2,...,N$ 进行
  $S_n^{(1)} \leftarrow 0$ and $F_n^{(1)} \leftarrow 0$
初始化空的数组 VEC，对于每个 $t = 1,2,...,T$ 进行
  对于每个 *rate* $r_n, n = 1,2,...,N$ 进行
    画出 $\alpha_n(t) \sim \text{Beta}\left(S_n^{(1)}+1, F_n^{(1)}+1\right)^2$
    // 从数组中取出所有的数据, 按照解释所演示的方法一样进行折扣计算并累加获得的折扣累加结果 $\{S_n^{(2)}, F_n^{(2)}\}$
    画出 $\beta_n(t) \sim \text{Beta}\left(S_n^{(2)}+1, F_n^{(2)}+1\right)^2$
  选择满足下列条件的 rate $r_{I(t)}$

$$I(t) = \underset{n=1,2,...,N}{\arg\max}\, r_n \alpha_n(t) \cdot \beta_n(t)$$

  观察随机预测结果 $X_{I(t)}(t)$ 和随机传输结果 $Y_{I(t)}(t)$
  如果 $X_{I(t)}(t) = 1$ 则
    $S_n^{(1)} \leftarrow S_n^{(1)} + 1$
  否则
    $F_n^{(1)} \leftarrow F_n^{(1)} + 1$
  如果 $Y_{I(t)}(t) = 1$ 则
    把 $\{1,0\}$ 压入 VEC
  否则
    把 $\{0,1\}$ 压入 VEC

由4-2可知基于折扣的 Thompson Sampling 算法重点在于计算累加值时需要先对历史数据进行折扣后再累加。

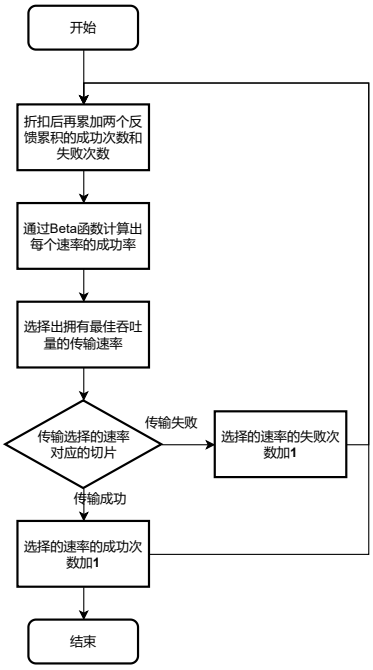

图 4-2 基于折扣的 Thompson Sampling 算法

4.3 基于滑动窗口的 Thompson Sampling 算法

不管怎么说，基于折扣的 Thompson Sampling 算法中，过去的不重要的数据还是对当前的决策有影响，那么我们可以用一个滑动窗口把最近的数据划出来，其他的数据丢弃。这么做的另一个好处是不需要一直存储着所有数据。当数据量过大时，基于折扣的 Thompson Sampling 算法在每个时隙都需要重新计算折扣累加值，这相当的耗费时间。在基于滑动窗口的 Thompson Sampling 算法中，我们维护一个固定容量的窗口，窗口的容量是指能存储的试验次数。这个窗口里面包含了所有手臂的实验结果。每一次实验结果以元组 $\{1,k\}$ 或 $\{0,k\}$ 方式存储，其中前者表示实验结果，如果是 1 表示成功，0 表示失败。，后者表示这是第 k 个手臂的实验结果。当存储的实验结果达到窗口容量时，每当存储新的结果时，需要把最早存储的结果从窗口中弹出来。获取某个手臂的累加结果时需要遍历滑动窗口来计算。

算法 4.3: 基于滑动窗口的 Thompson Sampling 算法

初始化滑动窗口 SW
对于每个 *rate* $r_n, n = 1, 2, ..., N$ **进行**
    $S_n^{(1)} \leftarrow 0$ and $F_n^{(1)} \leftarrow 0$
对于每个 $t = 1, 2, ..., T$ **进行**
    对于每个 *rate* $r_n, n = 1, 2, ..., N$ **进行**
        画出 $\alpha_n(t) \sim \text{Beta}\left(S_n^{(1)}, F_n^{(1)}\right)^2$
        // 累加滑动窗口中所有的值从滑动窗口 SW 中计算得到 r_n 的累加结果 $S_n^{(2)}$ 和 $F_n^{(2)}$
        画出 $\beta_n(t) \sim \text{Beta}\left(S_n^{(2)}, F_n^{(2)}\right)^2$
    选择满足下列条件的 rate $r_{I(t)}$

$$I(t) = \underset{n=1,2,...,N}{\arg\max}\, r_n \alpha_n(t) \cdot \beta_n(t)$$

    观察随机预测结果 $X_{I(t)}(t)$ 和随机传输结果 $Y_{I(t)}(t)$
    如果 $X_{I(t)}(t) = 1$ **则**
        $S_n^{(1)} \leftarrow S_n^{(1)} + 1$
    否则
        $F_n^{(1)} \leftarrow F_n^{(1)} + 1$
    如果 $Y_{I(t)}(t) = 1$ **则**
        把 $\{1, 0\}$ 压入 SW
    否则
        把 $\{0, 1\}$ 压入 SW
    如果 *滑动窗口* SW *大小超过容量* **则**
        弹出最早的数据

由4-3可知基于滑动窗口的 Thompson Sampling 算法重点在于积累的数据大小有一个局限，如果超出局限需要先弹出最早的数据再存储。

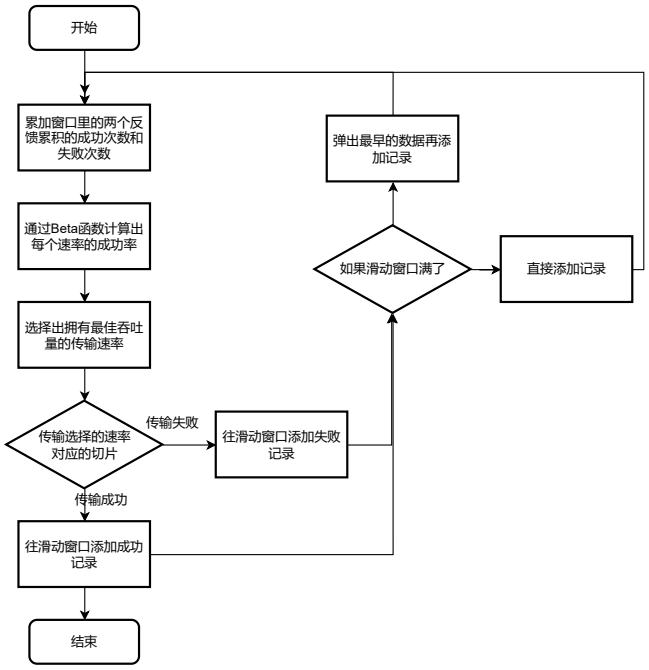

图 4-3 基于滑动窗口的 Thompson Sampling 算法

4.4 基于增加次数的滑动窗口 Thompson Sampling 算法

在基于滑动窗口的 Thompson Sampling 算法中，我们或许会碰到这种问题：周期发生变化时，算法还是不能快速反应过来选择出正确的手臂（ARM）。为什么呢？因为在周期发生变化的时候，滑动窗口里面可能大部分数据都是上一个周期的没用数据。如果我们在滑动窗口里面增加两个规则：（1）如果选择的速度超过

了阈值（比如占用了滑动窗口容量的一半或者更多），而且这个速度在传输过程中失败了，那么我们不再是只增加一次失败次数 $F_n^{(1)}(t)$，而是增加更多。因为出现这种可能性时往往表示信道容量发生变化了。（2）如果选择的手臂（ARM）在传输过程中成功了而且有其他手臂超过了阈值（比如占用了滑动窗口容量的一半或者更多），那么我们会给这个手臂对应的成功次数增加更多。在这个算法中，滑动窗口将存储每一次成功的“次数”和失败的“次数”，因此压入滑动窗口的值的形式为 $[arm_index, success_number, fail_number]$。

算法 4.4: 基于增加次数的 Thompson Sampling 算法

初始化滑动窗口 SW
对于每个 *rate* $r_n, n = 1, 2, ..., N$ **进行**
    $S_n^{(1)} \leftarrow 0$ and $F_n^{(1)} \leftarrow 0$
设置阈值 $threshold$ **对于每个** $t = 1, 2, ..., T$ **进行**
    对于每个 *rate* $r_n, n = 1, 2, ..., N$ **进行**
        画出 $\alpha_n(t) \sim \text{Beta}\left(S_n^{(1)}, F_n^{(1)}\right)^2$
        从滑动窗口 SW 中计算得到 r_n 的累加结果 $S_n^{(2)}$ 和 $F_n^{(2)}$
        画出 $\beta_n(t) \sim \text{Beta}\left(S_n^{(2)}, F_n^{(2)}\right)^2$
    选择满足下列条件的 rate $r_{I(t)}$

$$I(t) = \underset{n=1,2,...,N}{\arg\max}\, r_n \alpha_n(t) \cdot \beta_n(t)$$

    观察随机预测结果 $X_{I(t)}(t)$ 和随机传输结果 $Y_{I(t)}(t)$
    如果 $X_{I(t)}(t) = 1$ **则**
        $S_n^{(1)} \leftarrow S_n^{(1)} + 1$
    否则
        $F_n^{(1)} \leftarrow F_n^{(1)} + 1$
    // 如果窗口里面有速度对应的成功数量大于阈值, 则增加成功的次数
    如果 $Y_{I(t)}(t) = 1$ *AND* 有非 $I(t)$ 的 $threshold \geq armCounts$ **则**
        把 $\{I(t), add_number, 0\}$ 压入 SW
    // 如果窗口里面选择的速度 r_n 对应的成功数量大于阈值, 则增加失败的次数
    如果 $Y_{I(t)}(t) = 0$ *AND* $I(t)$ 的 $threshold \geq armCounts$ **则**
        把 $\{I(t), 0, sub_number\}$ 压入 SW
    如果 滑动窗口 SW 大小超过容量 **则**
        弹出最早的数据

由4-4可知基于增加次数的滑动窗口 Thompson Sampling 算法重点在于每次累积的数据不止一个，这需要根据信道特性进行设置，以便获取最佳吞吐量。

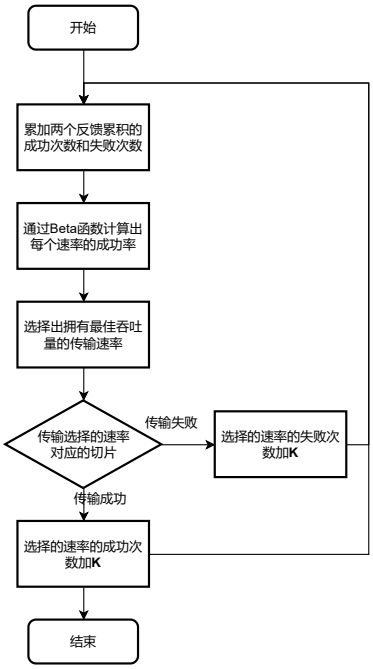

图 4-4 基于增加次数的滑动窗口 Thompson Sampling 算法

4.5 基于折扣、增加次数和滑动窗口的 Thompson Sampling 算法

在上述的除了周期性重置 Thompson Sampling 算法外，我们是否可以把所有的有点结合起来呢？当然可以，这就是基于折扣、增加次数和滑动窗口的 Thompson Sampling 算法。

算法 4.5: 基于滑动窗口的 Thompson Sampling 算法

初始化滑动窗口 SW **对于每个** *rate* $r_n, n = 1, 2, ..., N$ **进行**
  $S_n^{(1)} \leftarrow 0$ and $F_n^{(1)} \leftarrow 0$
设置阈值 $threshold$ **对于每个** $t = 1, 2, ..., T$ **进行**
  对于每个 *rate* $r_n, n = 1, 2, ..., N$ **进行**
    画出 $\alpha_n(t) \sim \text{Beta}\left(S_n^{(1)}, F_n^{(1)}\right)^2$
    从滑动窗口 SW 中计算得到 r_n 的折扣累加结果 $S_n^{(2)}$ 和 $F_n^{(2)}$
    画出 $\beta_n(t) \sim \text{Beta}\left(S_n^{(2)}, F_n^{(2)}\right)^2$
  选择满足下列条件的 rate $r_{I(t)}$

$$I(t) = \underset{n=1,2,...,N}{\arg\max}\, r_n \alpha_n(t) \cdot \beta_n(t)$$

  观察随机预测结果 $X_{I(t)}(t)$ 和随机传输结果 $Y_{I(t)}(t)$
  如果 $X_{I(t)}(t) = 1$ **则**
    $S_n^{(1)} \leftarrow S_n^{(1)} + 1$
  否则
    $F_n^{(1)} \leftarrow F_n^{(1)} + 1$
  如果 $Y_{I(t)}(t) = 1$ *AND* **有非** $I(t)$ **的** $armCounts \geq threshold$ **则**
    把 $\{I(t), add_number, 0\}$ 压入 SW
  如果 $Y_{I(t)}(t) = 0$ *AND* $I(t)$ **的** $armCounts \geq threshold$ **则**
    把 $\{I(t), 0, sub_number\}$ 压入 SW
  如果 *滑动窗口* SW *大小超过容量* **则**
    弹出最早的数据

4.6 本章小结

在周期信道中，我们基于 Thompson Sampling 算法又开发出五个新的算法，分别周期性重置 Thompson Sampling 算法、基于折扣的 Thompson Sampling 算法、基于滑动窗口的 Thompson Sampling 算法为和基于折扣滑动窗口的 Thompson Sampling 算法。这些算法是我们根据周期性容量变化的信道的特点而开发的，那么这些算法是否有效呢？在本文的下一章将通过仿真实验来验证。

第 5 章 仿真实验

在上一节中，我们基于双反馈 Thompson Sampling 算法开发了 5 个针对周期信道的优化算法（下面统称为 TS 算法）。在下面的仿真实验中，我们假设不知道信道的周期。如果知道了信道的周期，那么显然我们不再需要这些改进的双反馈 TS 算法。因为我们可以在每次周期改变的时候切换到上一次周期积累的数据。另外在考虑如何设定传输概率组数时，我们选择设定两组数据，根据我们的实验数据，两组数据就能大致展示出实验结果，且实验结果与 10 组、25 组等更多的组也大致相同。在模拟中，我们将时间范围设置为 10^4 个时隙，并运行 5000 个实验以确保平均遗憾是足够精确的。表 5-1 表示仿真实验参数设置。

表 5-1 仿真实验参数设置

	Rate1	Rate2	Rate3	Rate4	Rate5
传输速率 r_n	2	3	5	7	9
预测概率 α_n	0.1	0.3	0.6	0.7	0.9
传输概率 $\beta_n^{(1)}$	0.9	0.8	0.65	0.63	0.1
平均吞吐量 1	0.18	0.72	1.95	3.087	0.81
传输概率 $\beta_n^{(2)}$	0.99	0.85	0.75	0.15	0.01
平均吞吐量 2	0.198	0.765	2.25	0.735	0.081

5.1 置零时机对 TS 算法的影响

在这小节，我们研究置零时机对 TS 算法的影响。我们的实验是基于并不知道信道容量周期的条件进行的，因此很难把握在周期变化的时候置零，因此我们探索置零时机对 TS 算法的影响。下图中的置零偏差是指置零的时机与周期长度之比。图 5-1 显示出了结果。可以看出置零时机对置零结果影响重大. 只有准确的在周期发生改变时进行置零是最好的，其他情况下的结果都是非常差的。但是现实中我们难以把握这个时机，因此置零 TS 算法的实际应用效果应该没有那么好。

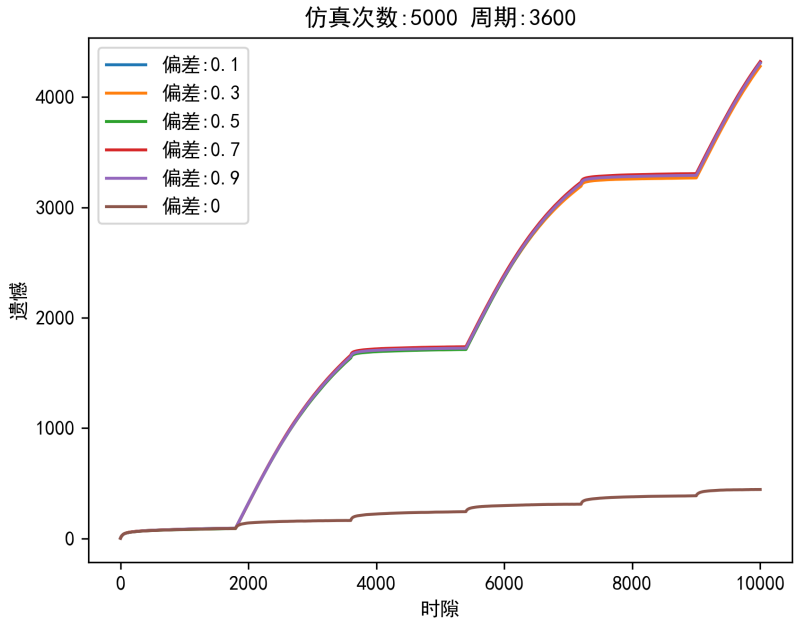

图 5-1 置零时机对 TS 算法的影响

5.2 总体比较

在本小节中，我们将比较七个算法的遗憾性能。这七个算法分别是单反馈的 Thompson Sampling 算法、两层反馈的 Thompson Sampling 算法、周期性重置 Thompson Sampling 算法、基于折扣的 Thompson Sampling 算法、基于滑动窗口的 Thompson Sampling 算法、基于增加次数的滑动窗口 Thompson Sampling 算法和基于折扣、增加次数和滑动窗口的 Thompson Sampling 算法。图 5-2可以看出经过改进的 TS 算法表现都非常优秀。

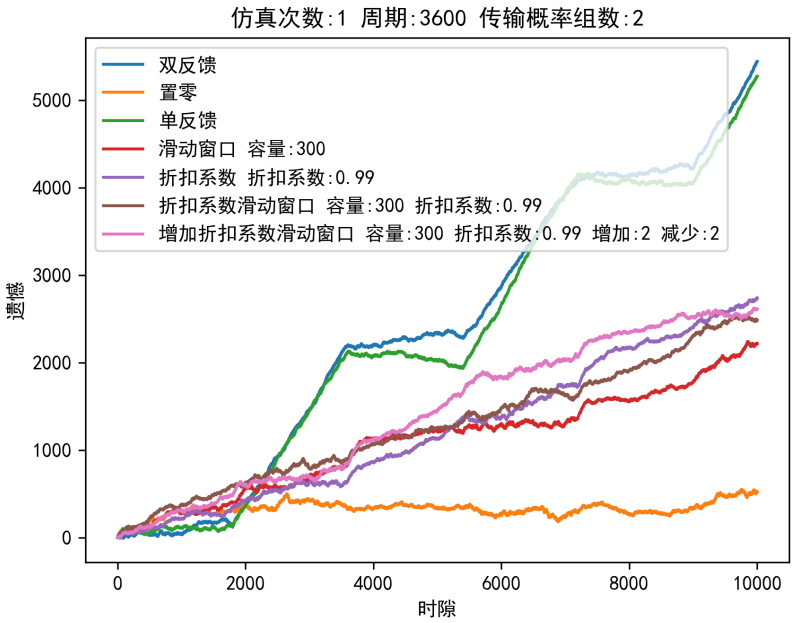

图 5-2 七个算法小周期的效果比较图

5.3 窗口容量对 TS 算法的影响

在这小节，我们选定基于滑动窗口的 TS 算法来研究窗口容量对 TS 算法的影响。窗口容量的单位设定为时间，即窗口容量表示能存储多少时隙的结果。我们设定了 8 个窗口容量值，100、250、300、500、800、1000 和 2000。图 5-3显示窗口容量值处于 250-500 的表现都非常好。

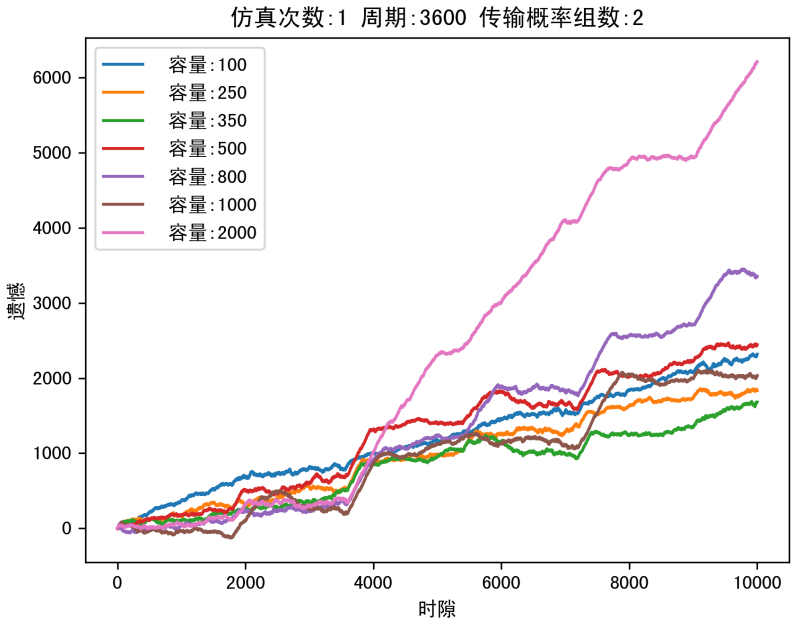

图 5-3 窗口容量对 TS 算法的影响

5.4 折扣系数对 TS 算法的影响

在这小节，我们选定基于折扣的 Thompson Sampling 算法来研究折扣系数对 TS 算法的影响。我们设定了 5 个折扣系数值，分别为 0.1、0.5、0.8、0.9、0.95 和 1。令人惊讶，图 5-4显示出折扣系数为 1 的算法表现更佳。折扣系数对 TS 算法的影响和窗口容量关系比较大，很可能是测试的这个窗口容量条件导致的结果。

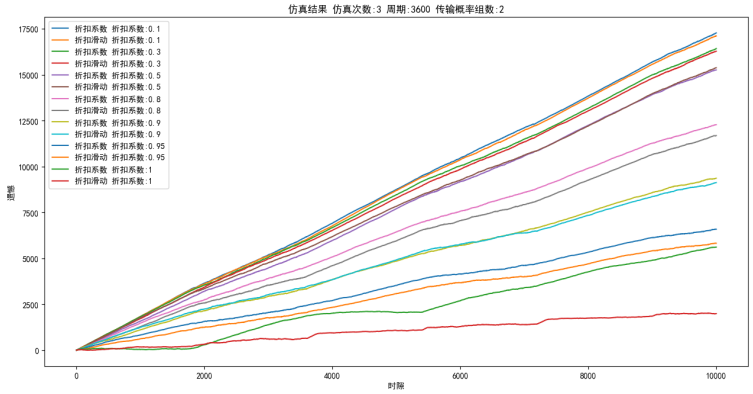

图 5-4 折扣系数对 TS 算法的影响

5.5 增加失败次数对 TS 算法的影响

在这小节，我们选定基于折扣的 Thompson Sampling 算法来研究折扣系数对 TS 算法的影响。我们设定了 5 个折扣系数值，分别为 0.1、0.5、0.8、0.9、0.95 和 1。令人惊讶，图 5-5显示出失败次数对结果没有影响。这说明在某一个手臂失败的时候它被选择的次数并没有超过窗口。

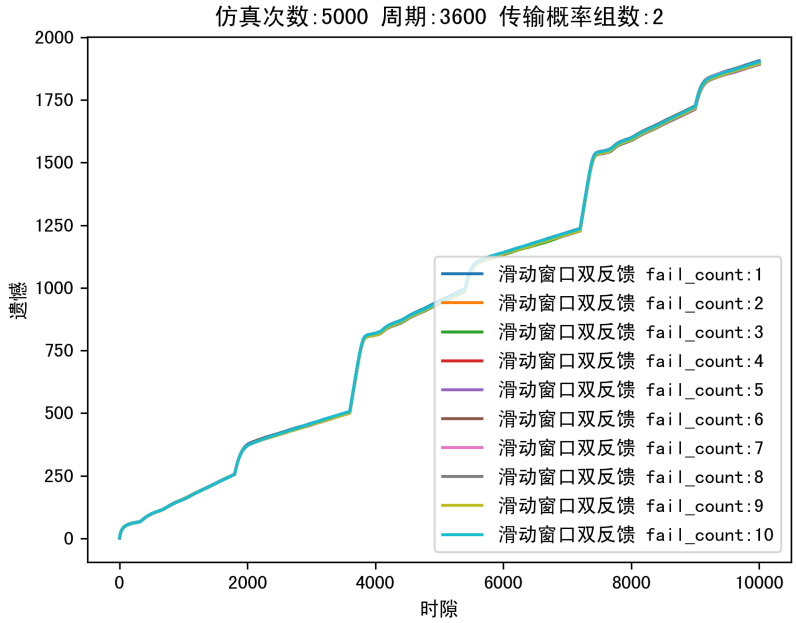

图 5-5 增加失败次数对 TS 算法的影响

5.6 增加成功次数对 TS 算法的影响

在这小节，我们选定基于折扣的 Thompson Sampling 算法来研究折扣系数对 TS 算法的影响。我们设定了 5 个折扣系数值，分别为 0.1、0.5、0.8、0.9、0.95 和 1。令人惊讶，图 5-6显示出增加成功次数到 1 和 2 的效果最好。

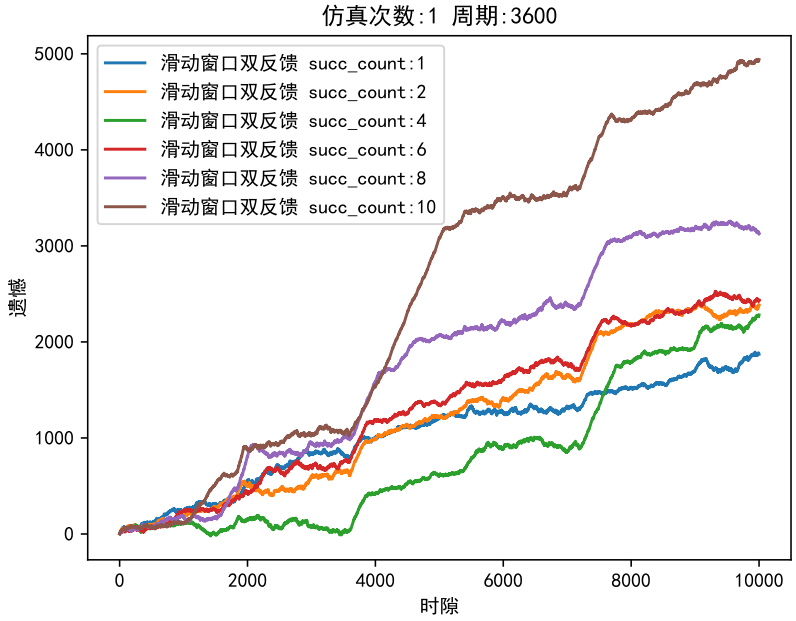

图 5-6 增加成功次数对 TS 算法的影响

5.7 折扣滑动窗口 TS 算法深入探索

在这一小节，我们深入探索折扣滑动窗口 TS 算法

5.7.1 不同周期下折扣滑动窗口 TS 算法效果比较

不同周期对算法的表现还是有很大的影响的，从图 5-7可以看出周期越大，不同周期下折扣滑动窗口 TS 算法效果越好，这很容易理解，因为周期改变导致 TS 算法转变所占用的时间变少了。

5.7.2 多种传输概率变换下折扣滑动窗口 TS 算法效果比较

我们上面的实验全部都是两组传输概率变换的实验，现在我们使用五十组传输概率来测试折扣滑动窗口 TS 算法。从图 5-8可以看出折扣滑动窗口 TS 算法的表现依然胜过两层反馈 TS 算法

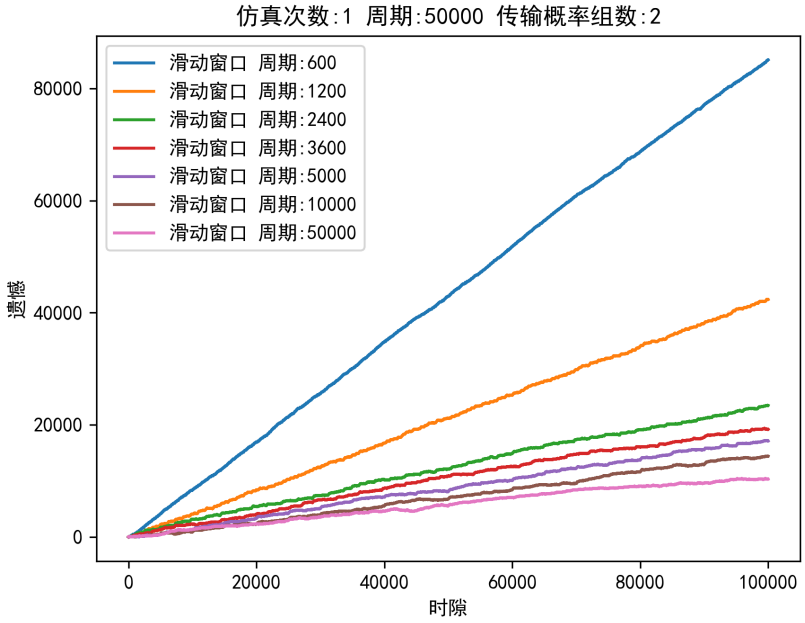

图 5-7 不同周期下折扣滑动窗口 TS 算法效果比较

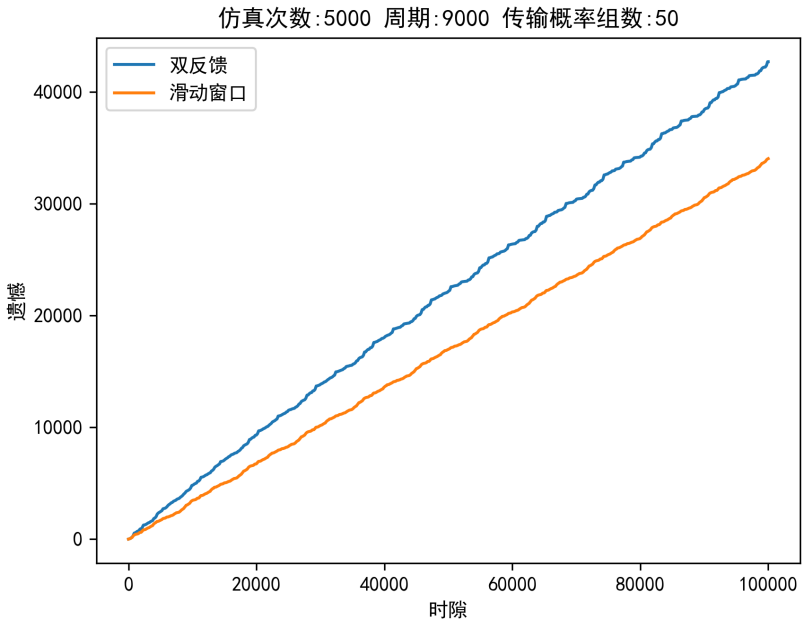

图 5-8 多种传输概率变换下折扣滑动窗口 TS 算法效果比较

5.8 总结

从上述仿真实验可以看出，我们改进的算法都是非常有效的。

第 6 章 总结与展望

6.1 工作总结

在本文中，我们考虑 360 度全景视频流的速度选择问题，并将其表述为具有两级反馈信息多臂老虎机（MAB）问题，其中所选的臂在每次播放后都有预测结果和传输结果。直观地说，选择速度越大，预测成功的概率就越大。但代价是的传输失败的概率增加。我们的目标是适当地确定每个时隙中选择的速度，目标是在有限的范围内最大化系统吞吐量。为此，我们提出了多种改进的 Thompson Sampling 算法，这些算法都有效地利用了两级反馈信息，并证明了其性能优于未改进的两级反馈的 Thompson Sampling 算法。

6.2 研究展望

注意，在本文中，我们设定传输概率的分布为伯努利分布，并没有考虑更加复杂的情况，比如高斯分布等。而且设定的传输成功概率形式很简单，并不能完全反应出实际情况。同时在周期长度比较短的情况下这些改进的算法效果并不明显。这些问题留待将来的工作解决。

相关的科研成果目录

1. 发表论文

   [1] XXX

   [2] XXX

   [3] XXX

2. 发明专利

   (1) XXX

   (2) XXX

   (3) XXX

3. 获奖

   (1) XXX

   (2) XXX

   (3) XXX

致谢

（谢辞应以简短的文字对课题研究与论文撰写过程中曾直接给予帮助的人员(例如指导教师、答疑教师及其他人员) 表示对自己的谢意，这不仅是一种礼貌，也是对他人劳动的尊重，是治学者应当遵循的学术规范。内容限一页。）

姓名

2022 年 3 月 15 日

附录 A 补充更多细节

附录 B　多附录

附 B.1　多附录

附录 C 参考文献

[1] Nan Jiang, Viswanathan Swaminathan, and Sheng Wei. 2017. Power Evaluation of 360 VR Video Streaming on Head Mounted Display Devices. In Proceedings of the 27th Workshop on Network and Operating Systems Support for Digital Audio and Video. ACM, 55–60.

[2] Xing Liu, Qingyang Xiao, Vijay Gopalakrishnan, Bo Han, Feng Qian, Matteo Varvello:360° Innovations for Panoramic Video Streaming. HotNets 2017: 50-56

[3] J. Chen, B. Li and R. Srikant, Thompson-Sampling-Based Wireless Transmission for Panoramic Video Streaming, 2020 18th International Symposium on Modeling and Optimization in Mobile, Ad Hoc, and Wireless Networks (WiOPT), 2020, pp. 1-3.

[4] Philbin J, Chum O, Isard M, et al. Lost in quantization: Improving particular object retrieval in large scale image databases, Computer Vision and Pattern Recognition, 2008. CVPR 2008, IEEE Conference on, IEEE, 2008: 1-8.

[5] Tang M , Wong V . Online Bitrate Selection for Viewport Adaptive 360-Degree Video Streaming[J]. 2020.

[6] L. Xie, X. Zhang, and Z. Guo, “CLS: A cross-user learning based system for improving QoE in 360-degree video adaptive streaming,” in Proc. ACM Int’l Conf. on Multimedia (MM), Seoul, Republic of Korea, Oct. 2018.

[7] Harsh Gupta, Jiangong Chen, Bin Li, R. Srikant, Online Learning-Based Rate Selection for Wireless Panoramic Video Streaming.

[8] J. Chen, B. Li and R. Srikant, Thompson-Sampling-Based Wireless Transmission for Panoramic Video Streaming, 2020 18th International Symposium on Modeling and Optimization in Mobile, Ad Hoc, and Wireless Networks (WiOPT), 2020, pp. 1-3.

毕业论文(设计)成绩评定记录

指导教师评语： 成绩评定： 指导教师签名:　　年　月　日
答辩小组或专业负责人意见： 成绩评定： 签名:　　年　月　日
院系负责人意见： 成绩评定： 签名（章）： 年　月　日