边缘辅助实时应用中信息年龄感知的任务调度

王红艳，孙其博，马骁，周傲，王尚广
（北京邮电大学网络与交换技术国家重点实验室，北京 100876）

摘　要：针对无线设备资源受限导致状态提取滞后难以满足实时应用对信息新鲜度需求的问题，考虑到边缘节点处理容量的有限性，提出了一种联合考虑信息新鲜度与调度实时性的调度方法。该方法首先利用队列的系统时间和信息年龄分别刻画任务在计算之前的时延和计算之后的信息新鲜度，同时给每个卸载任务合理的截止期限，来保证任务进入计算过程之前的有效性。然后，采用最小处理速率约束方法对任务调度过程中的处理速率进行约束，保证任务调度的实时性。最后，基于Lyapunov优化技术实现优化长期任务调度决策的目的。仿真结果表明，所提方法在调度实时性和系统信息新鲜度方面均具有较好的性能。

关键词：边缘计算；信息年龄；任务调度；截止期限；Lyapunov优化
中图分类号：TP311
文献标志码：A
DOI：10.11959/j.issn.1000–436x.2024109

AoI-aware task scheduling in edge-assisted real-time applications

WANG Hongyan, SUN Qibo, MA Xiao, ZHOU Ao, WANG Shangguang
State Key Laboratory of Networking and Switching Technology, Beijing University of Posts and Telecommunications, Beijing 100876, China

Abstract: To address the issue where the resource limitations of wireless devices caused state extraction delays that cannot meet the freshness requirements of real-time applications, considering the limited processing capacity of edge nodes, a scheduling method that jointly considered information freshness and real-time performance was proposed. This method initially characterized the task delay before computation and the information freshness after computation by utilizing the system time of the queue and the age of information, respectively. Simultaneously, reasonable deadlines were assigned to each offloaded task to ensure their validity before entering the computation process. Then, the minimum processing rate constraint method was employed to restrict the processing rate during task scheduling, thereby ensuring the real-time nature of task scheduling. Finally, the objective of optimizing long-term task scheduling decisions was achieved based on Lyapunov optimization techniques. Simulation results demonstrate the good performance of the proposed method in both scheduling timeliness and system information freshness.

Keywords: edge computing, age of information, task scheduling, deadline, Lyapunov optimization

0　引言

5G[1]的商业化加速了移动通信与物联网的深度融合，有望支持多种实时性应用，包括传感器网络中的监控和响应、车载网络中的自动驾驶以及工业控制网络中的电器自动控制等，其中信息新鲜度对该类实时性应用的监测和精准控制起到了至关重要的作用。信息年龄[2-7]（AoI, age of information）作为一种增强时延的性能指标，通过计算任何给定时

收稿日期：2023–12–29；修回日期：2024–05–21
通信作者：孙其博， qbsun@bupt.edu.cn
基金项目：国家自然科学基金资助项目（No.62032003, No.U21B2016, No.62372061, No.61921003）
Foundation Items：The National Natural Science Foundation of China (No.62032003, No.U21B2016, No.62372061, No.61921003)

刻与接收最新信息的产生时刻之间的差值，来衡量目的端接收信息的新鲜度，被广泛应用于各类实时应用[8-11]中。

在一类实时应用中，如自动驾驶环境感知、面部识别、目标检测等，无线设备采集的原始数据不能直接用于系统状态信息[12]的更新，而是需要通过状态提取（计算）过程将隐含的状态信息提取出来。但是无线设备由于自身资源（如电池容量、计算资源等）的局限性，很难满足这类实时应用对 AoI 的需求。移动边缘计算[13]（MEC, mobile edge computing）的日益商业化为具有信息提取需求的实时应用提供了机遇。在 MEC 的使能下，实时应用所需的信息需要经历“接入、排队、计算、转发”4 个阶段，边缘辅助实时应用中随机到达任务的实时调度处理过程如图 1 所示。例如，在实时目标检测应用中，摄像机不断捕获周围环境的视频数据，这些视频数据并不能直接用于目标检测系统的信息更新，而是需要将其上传到边缘服务器进行目标特征信息提取，最后将目标特征信息转发给用户。在这种情况下，由于更新用户系统的信息是经过计算处理之后的特征信息，因此，用户在任意时刻的信息年龄可以通过计算给定时刻与该时刻接收到最新特征信息的产生时刻之间的差值来衡量，差值越小，信息年龄就越小，信息新鲜度越高；反之亦然。

虽然目前 MEC 能够高效地为计算资源受限的无线设备提供计算服务，但是当有大量的无线设备发起服务请求时，MEC 并不能保证用户端接收的信息具有较好的新鲜度。因为与传统中心云充足的计算资源相比，边缘服务器的计算能力通常十分有限，不同任务流之间对有限计算资源的竞争可能带来较长的排队等待[14]时延，导致目的端接收陈旧信息的可能性变大。此外，卸载到边缘服务器的任务会因长时间的等待导致其在处理之前就已经失效（如图 1 排队中所标记的过期任务包），处理失效任务会浪费边缘服务器有限的计算资源，这在一定程度上加剧了目的端接收陈旧信息的可能性。因此，在边缘辅助实时应用中，如何在计算阶段进行合理的任务调度，保证用户端接收信息的 AoI 是一个很重要的课题。

目前，围绕移动边缘计算的任务调度已有大量的研究工作，为实时应用的快速发展起到了重要的推动作用。调度实时性是指系统在严格的时间限制内响应和处理任务的能力。为了保证任务按时处

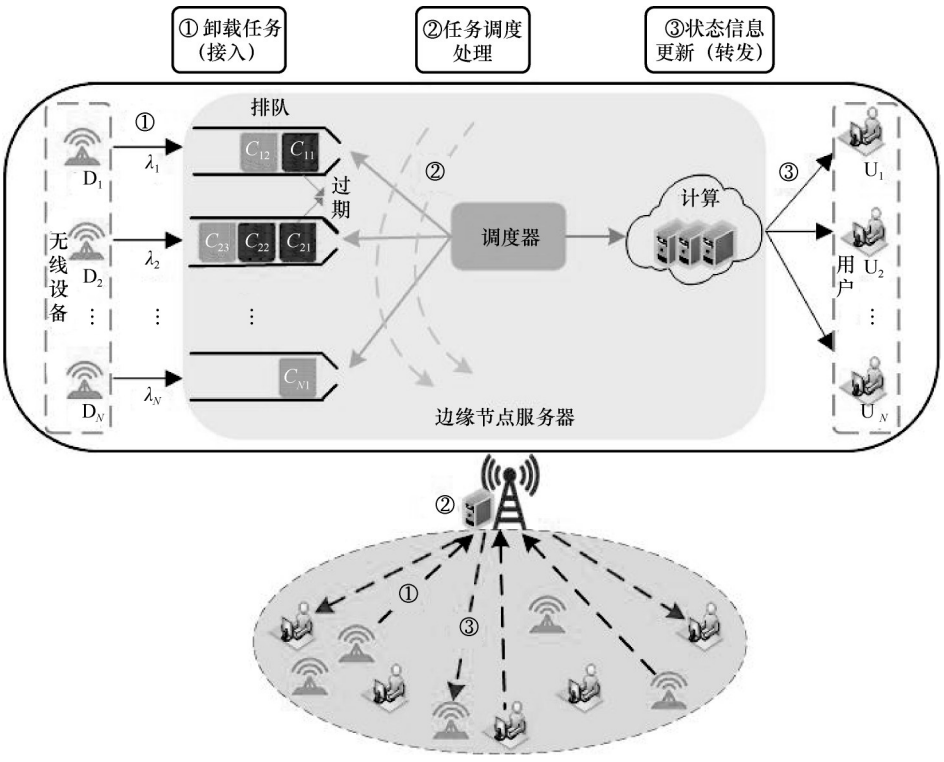

图 1　边缘辅助实时应用中随机到达任务的实时调度处理过程

理，研究主要关注时延、吞吐量和完成率等传统性能指标的性能提升。文献[15]在计算单元能耗的限制下，研究了任务并行处理过程中的时延最小化问题，提出了高效的一维搜索算法优化任务调度。文献[16]研究了基于无人机的MEC，构建了具有传输速率、任务数量和无人机飞行速率约束的吞吐量最大化问题，提出了动态任务准入算法来平衡移动效率和系统吞吐量，解决了固定基站覆盖范围有限和过载处理的问题。为了进一步保证时延敏感型任务的按时完成情况，文献[17]提出了截止期限感知的在线任务调度方法，通过贪婪地决策新任务替代已有任务，满足新任务的截止期限。

然而，针对传统性能指标的优化旨在提高任务调度实时性，忽视了任务调度过程中对信息新鲜度的保证。随着实时应用对信息新鲜度的需求日益增加，研究工作开始关注任务在传输过程中对信息新鲜度的保证。文献[18]针对对称无线网络提出了贪婪算法，来提高干扰信道传输过程中的信息新鲜度，而不考虑任务调度的实时性。对于更一般化的网络场景，文献[19]提出了具有低复杂度特性的随机算法、最大权重算法和惠特尔索引[20]算法，这些算法在保证任务调度实时性的前提下，旨在提升用户端接收信息的新鲜度。对于需要进行状态提取的信息，仅提升传输过程中信息的新鲜度并不能保证用户端接收信息的新鲜度。为了衡量计算过程对信息新鲜度的影响，文献[21]研究在给定截止期限之前传输和计算一组数据包的信息年龄最小化问题，对于适当的截止期限，提出的零等待计算调度策略可以提升任务处理之后的信息新鲜度，但其不考虑多个任务流的资源竞争。文献[22]考虑异步任务流对MEC的计算资源竞争导致用户端接收陈旧信息的问题，提出了最大权重年龄下降的贪婪算法来最大程度地提高用户端接收信息的新鲜度。然而，该算法仅适用于具有单个处理单元的调度过程，且未考虑任务的截止期限限制。综上所述，在MEC辅助信息更新的场景中，具有截止期限限制的任务流对多个并行计算资源竞争导致信息新鲜度降低和调度实时性变差的问题尚未得到充分研究。本文主要的研究工作如下。

1) 建立了具有截止期限限制的任务流并行调度模型。在满足MEC处理容量和有效任务处理速率的前提下，通过优化任务调度策略，最小化用户端接收状态信息的期望加权平均信息年龄（ewaAoI, expected weighted average AoI）问题。将该问题抽象为一个长期动态优化问题，很难通过精确算法直接求解。

2) 为了降低计算复杂度，利用处理债务的递归过程与队列的演进相结合，将长期动态优化问题转化为每个时隙的静态优化问题。进一步地，基于李雅普诺夫（Lyapunov）优化技术提出了低复杂度的信息年龄和截止期限感知的任务调度方法，以期实现调度过程中信息新鲜度与调度实时性的高效均衡，并给出了算法的理论性能下界和下界求解方法，同时对算法的计算复杂度进行了分析。

3) 进行了大规模的仿真实验。结果表明，本文所提方法在调度实时性和信息新鲜度方面均具有较好的性能。此外，相较于基准调度方法，本文所提方法在多设备节点场景下具有很好的可扩展性与适应性。

1 系统模型

本文研究边缘辅助实时应用中信息年龄感知的任务调度问题，该系统包含一个具有MEC功能的边缘节点（EN, edge node）、N个无线设备$\mathrm{D}_1,\mathrm{D}_2,\cdots,\mathrm{D}_N$和$N$个用户$\mathrm{U}_1,\mathrm{U}_2,\cdots,\mathrm{U}_N$。其中，EN通过共享的MEC服务器为多个卸载任务设备提供计算服务，将完成任务调度处理之后的状态信息更新到对应的N个用户，如图1所示。本文考虑一个具有时隙索引为$t\in\{1,2,\cdots,T\}$的离散时间系统，其中，T为系统的时间域，时隙大小为v。无线设备D_l按任务的产生顺序将需要处理的任务包通过EN卸载到MEC中。本文假设不同无线设备产生的任务包到达MEC的过程相互独立，且服从概率为λ_l的泊松分布，其中$l\in\{1,2,\cdots,N\}$。更多系统参数如表1所示。

本文假设MEC为每个无线设备的卸载任务流分别分配一个单独的缓存，记为$X_1,X_2,\cdots,X_N$，且每个缓存支持多个卸载任务的存储，任务到达缓存之后按照先进先出（FIFO, first in first out）的队列模式进行排队，即队列中最早到达的任务最先被处理。考虑到卸载到MEC的任务可能会因长时间得不到处理服务超过截止期限而被丢弃，定义$C_l(t)$为时隙t开始时队列X_l队首任务的系统时间（即卸载任务产生之后经历等待、传输及进入MEC缓存之后的计算等待的总时延[23-24]），统一将系统时间

转化为时隙数并定义为$C_l(t)=t-\frac{\tau_l^C(t)}{v}$，其中，$\tau_l^C(t)$为队首任务在无线设备$\mathrm{D}_l$的产生时刻。显然，只有当队首任务改变时，即队首任务因超过截止期限而被丢弃或被服务（计算处理），$\tau_l^C(t)$才会改变，而队列为空时的$C_l(t)$不在本文研究的考虑范围内。

表1　　系统参数

参数	含义
N、T、t、v	无线设备数量、时间域、时隙索引、时隙大小
D_l、U_l	第l个无线设备、第l个用户
X_l、λ_l	与D_l对应的缓存队列、任务到达率
$\tau_l^c(t)$	缓存队列X_l中队首任务的产生时刻
$C_l(t)$	时隙t开始时队列X_l队首任务的系统时间
$C_l'(t)$	时隙t开始时队列X_l第二个任务的系统时间
$\delta_l(t)$	时隙t开始时用户接收来自D_l更新的信息年龄
m	MEC处理单元个数
$\bar{d}_l$	D_l卸载任务的长期处理速率
θ_l	与D_l相关的信息年龄权重
s_l^j	处理来自D_l的第j个任务包所需时间
T_l^{Deadline}	D_l卸载任务允许的最大截止期限
$\mu_l^{\max}$	D_l允许的最大丢包率

与文献[22]假设MEC具有单个处理单元不同，本文假设MEC服务器被虚拟化为m个计算能力相同且相互独立的处理单元。此外，本文定义来自无线设备D_l的第j个任务包的处理时间为s_l^j，并设定了一个足够长的时隙长度v，以确保在一个时隙内可以完成任意任务的处理，即$s_l^j\leqslant v,\forall j,l$。由此可知，调度程序在任意时隙$t$最多可以同时选择$m$个用户卸载的任务进行处理。本文引入一个二进制向量$\boldsymbol{w}(t)=(w_1(t),w_2(t),\cdots,w_N(t))$，用于表示调度程序在时隙$t$是否选择处理$\mathrm{D}_l$卸载的任务。在每个时隙$t$，对于任意$\mathrm{D}_l$卸载的任务，调度决策存在2种情况，当$w_l(t)=1$时，表示在时隙$t$时调度程序选择$\mathrm{D}_l$卸载的任务进行处理；否则，$w_l(t)=0$。因此，本文对$w_l(t)$的约束为

$$\sum_{l=1}^{N}w_l(t)\leqslant m,\forall t \tag{1}$$

由于处理后的状态信息大小远小于原始更新任务的大小，而且下行带宽通常足够大，状态信息的下载时间相对于原始更新任务的上传时间和计算处理时间相差很大。因此，本文研究假设状态信息的下载时间可以忽略不计[22,25]，即处理后的状态信息可以立刻更新到相应的用户端。

1.1　任务有效性判定

考虑到同一无线设备产生的任务类型相同，本文定义与无线设备D_l产生任务的系统时间相关的最大截止期限为T_l^{Deadline}，一旦任务的系统时间超过最大截止期限T_l^{Deadline}，该任务就被丢弃。尽管后进先出队列对于信息新鲜度有很大的优势，但本文考虑的实时应用（如目标跟踪和视频监控）产生的任务在没过期之前都具有一定价值，因此本文采用最简单的先进先出的队列模式，在这种队列模式下，队首任务的系统时间最长，即未来最快将要超过最大截止期限的任务。

为了确保调度程序在任意时隙所选任务是有效的，本文在调度之前对所有队首任务的有效性进行了判定，定义了一个指示函数$I_l(t)$，用其判断时隙t开始时，队列X_l的队首任务是否有效。因此，为确保所选任务是有效的（$I_l(t)=1$），队首任务的系统时间应不大于对应无线设备任务的最大截止期限，否则，该队列的队首任务是无效的（$I_l(t)=0$），且队首任务将被丢弃。在任意时隙开始时，定义队首任务的有效性判定式为

$$I_l(t)=\begin{cases}1, & C_l(t)\leqslant T_l^{\mathrm{Deadline}}\\ 0, & \text{其他}\end{cases} \tag{2}$$

虽然基于队首任务的有效性判定可以避免调度程序选择队首任务失效的队列进行调度，确保MEC处理器始终处理的是有效任务，但很难满足用户对数据流的实时性需求。因此，本文需要为来自每个D_l的卸载任务流施加一定的处理速率约束，以确保有效任务的调度实时性。

1.2　长期处理速率

在MEC对卸载任务进行处理的过程中，本文定义了一个二进制指示函数$d_l(t)$，用来表示在时隙t时用户U_l是否接收到来自D_l的有效更新数据。在时隙t，如果MEC调度D_l卸载的任务流进行处理，并且该队列的队首任务是有效的，则$d_l(t)=1$；否则，$d_l(t)=0$。由此可知，$d_l(t)$由$w_l(t)$和$I_l(t)$共同决定，即$d_l(t)=w_l(t)I_l(t)$。基于此，定义MEC使用调度策略ξ处理来自D_l有效任务的长期处理速率为

$$\bar{d}_l^{\xi}=\lim_{T\to\infty}\frac{1}{T}\sum_{t=1}^{T}d_l(t),\forall l \tag{3}$$

本文通过衡量单位时隙内处理未超过截止期限的任务量来评估调度策略ξ的实时性。对于任意D_l，本文假设MEC对其有效任务的长期处理速率$\bar{d}_l > 0$。定义$\mu_l^{\max}$为D_l卸载任务超过截止期限允许的最大丢包率。根据每个用户对任务流的实时吞吐量需求，在计算处理阶段，MEC需要为每个D_l提供不低于$\lambda_l(1-\mu_l^{\max})$的处理速率才能满足系统需求。因此，长期处理速率应满足以下约束条件。

$$\bar{d}_l \geqslant \lambda_l(1-\mu_l^{\max}), \forall l \tag{4}$$

本文假设$\{\lambda_l(1-\mu_l^{\max})\}_{l=1}^{N}$为所需最小处理速率的可行集，即存在调度策略$\xi \in \Gamma$能够同时满足式(1)和式(4)的需求，根据文献[26]中引理5可得

$$\sum_{l=1}^{N}\lambda_l(1-\mu_l^{\max}) \leqslant m \tag{5}$$

1.3 信息年龄

本文定义$C_l'(t)$为缓存队列X_l中第2个任务的系统时间，基于$C_l(t)$的定义及其改变条件，$C_l(t)$演化过程可以表示为

$$C_l(t+1)=\begin{cases}C_l'(t)+1, & d_l(t)=1 \text{ 或 } I_l(t)=0\\ C_l(t)+1, & \text{其他}\end{cases} \tag{6}$$

定义$\delta_l(t)$为用户端在时隙t接收到来自D_l更新信息的AoI。由于MEC处理之后信息的下载时间可以忽略，因此，用户端的信息年龄即MEC处理之后的信息年龄。$\delta_l(t)$演化过程可表示为

$$\delta_l(t+1)=\begin{cases}C_l(t)+1, & d_l(t)=1\\ \delta_l(t)+1, & \text{其他}\end{cases} \tag{7}$$

为了简化系统且不失一般性，本文假设初始系统时间和目的端初始信息年龄分别为$C_l(0)=\alpha$, $\delta_l(0)=1, \forall l \in \{1,\cdots,N\}$，式(7)进一步可以表示为

$$\delta_l(t+1)=(C_l(t)+1)d_l(t)+(\delta_l(t)+1)\cdot (1-d_l(t)), \forall l \tag{8}$$

图2描述了缓存队列X_l中的系统时间$C_l(t)$和信息年龄$\delta_l(t)$的演化过程。相对于系统时间，任务需要经历处理阶段，使得用户端接收的信息年龄较系统时间更长些。设置时隙t时的系统时间为0，信息年龄为2。例如，在时隙$t+1$开始时，缓存队列中第1个任务C_l^1到达MEC服务器，假设该任务的产生时刻为τ_l^1。在时隙$t+1$结束时，缓存队列中任务的系统时间骤变为$t+1-\frac{\tau_l^1}{v}$，由于在时隙$t+1$该任务未被处理或丢弃，因此在时隙$t+2$内保持不变，并在时隙$t+2$结束时骤变为$t+2-\frac{\tau_l^1}{v}$。注意到，在时隙$t+3$时，队列中的队首任务被MEC调度处理，队首任务发生了变化，变为时隙$t+2$时到达的任务C_l^2。假设任务C_l^2的产生时刻为τ_l^2，该任务在时隙$t+4$开始时，系统时间变为$t+3-\frac{\tau_l^2}{v}$。相应地，信息年龄骤降为调度任务的系统时间和处理时间之和$t+3-\frac{\tau_l^1}{v}$。在时隙$t+8$时，系统时间发生了骤变，这是因为该队列的队首任务C_l^2因超过截止期限而被丢弃，此时，队首任务变为时隙$t+6$时到达的任务C_l^3，系统时间变为该队首任务在系统中的时延，但是信息年龄仍然会随着时间的递增而递增。基于上述对图2的分析可以发现：1) 仅当MEC对队列中的任务选择时，信息年龄才会发生变化，并且变为所选任务的系统时间和处理时间之和；2) 当队首任务的系统时间因超过截止期限被丢弃或者任务被调度处理时，系统时间才会发生变化，变为队列中第2个任务在系统中的时延。

与大多数考虑信息年龄在一个时隙内保持恒定的研究[7,18-19]类似，本文所考虑的信息年龄在每个时隙结束时更新，而在整个时隙内保持不变。基于此，本文定义了用户端在时间域T无限增大时的期望加权平均信息年龄，表示为

$$J(\xi)=\lim_{T\to\infty}\frac{1}{NT}\mathrm{E}\left[\sum_{t=1}^{T}\sum_{l=1}^{N}\theta_l\delta_l(t)|\delta_l(0)\right] \tag{9}$$

其中，$\mathrm{E}(\cdot)$表示由任务到达率和调度方法的随机性带来的信息年龄期望；正实数θ_l表示与无线设备D_l相关的信息年龄权重，代表了D_l任务流的重要程度，即θ_l越大，D_l任务流调度的优先级越高，反之亦然。

进一步地，为了保证任务处理的有效性和调度的实时性，本文根据任务的不同截止期限和处理速率对卸载到MEC的任务进行调度。

1.4 优化问题

在连续$T(T\to\infty)$个时隙内，本文在每个时隙开始时利用给定方法$\xi \in \Gamma_a$来衡量所有用户U_l接收信息的期望加权平均信息年龄，评估整个MEC系统的信息新鲜度。本文定义了一个信息年龄最优的调度方法$\xi^* \in \Gamma_a$来最小化系统的ewaAoI，即

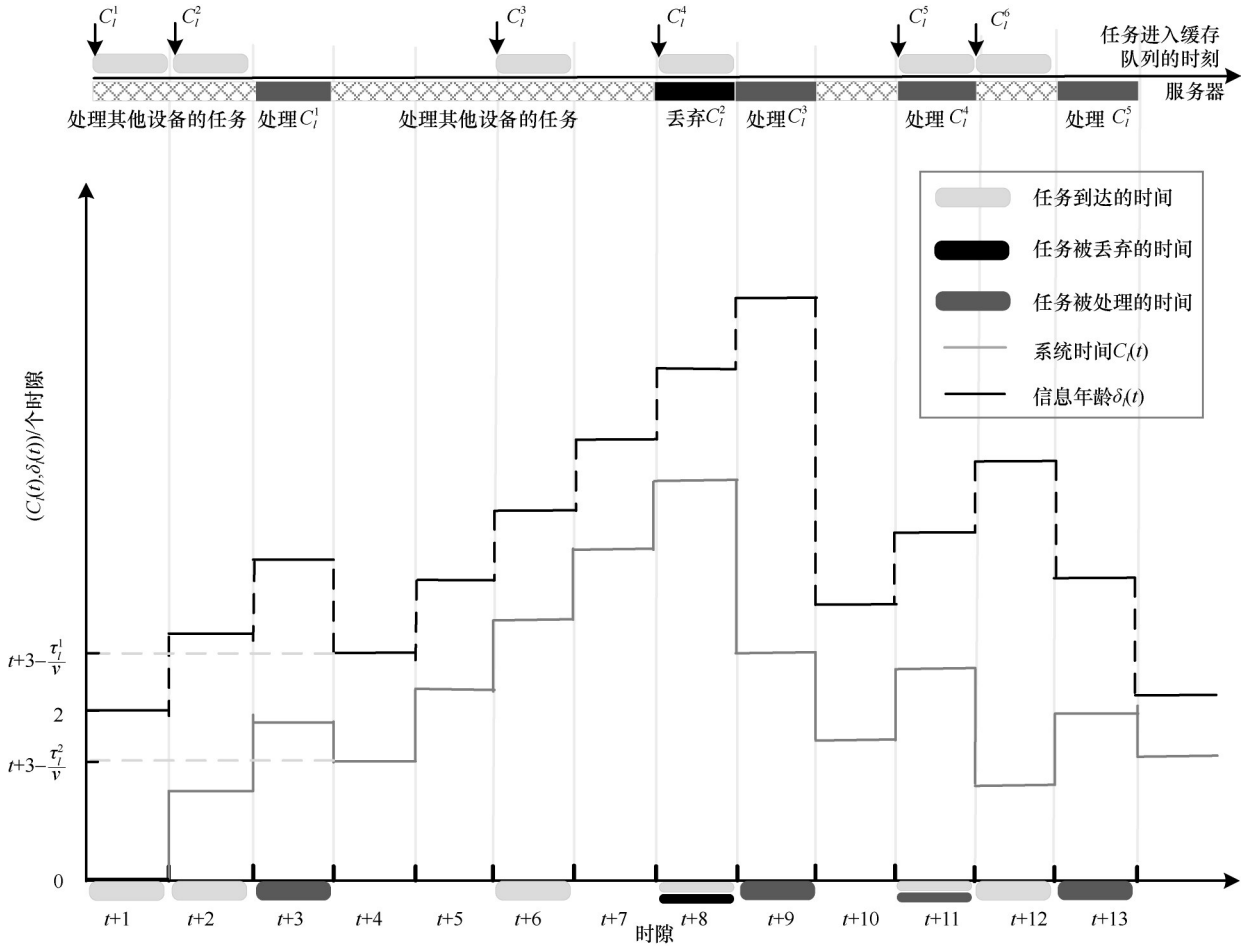

图2　系统时间和信息年龄的演化过程

$$J(\xi^*)=\min_{\xi\in\Gamma_a}J(\xi) \tag{10}$$

定义Γ_a为非预测性方法的类别，即属于该类的调度方法均基于过去和现在的状态信息来做出决策，而非依赖于未来的预测进行决策。本文旨在设计调度方法$\xi^*\in\Gamma_a$来最小化系统ewaAoI，同时满足MEC处理容量和处理速率的约束。因此，该随机优化问题可以表示为

$$\begin{aligned}&\mathrm{P}:\quad J(\xi^*)\\&\text{s.t.}\quad \mathrm{C}_1:\ \sum_{l=1}^{N}w_l(t)\leqslant m,\forall t\\&\qquad \mathrm{C}_2:\ \bar{d}_l\geqslant\lambda_l(1-\mu_l^{\max}),\forall l\\&\qquad w_l(t)\in\{0,1\},\forall l,t\end{aligned} \tag{11}$$

从上述约束中可以发现，调度方法在每个时隙t都需要满足C_1中严格的处理容量限制，同时也需要满足C_2中的长期处理速率约束。本文称由式(11)产生的调度方法为信息年龄最优方法。

2　信息年龄-截止期限感知的任务调度方法

2.1　问题分析

尽管问题P的优化函数$\delta_l(t+1)=(C_l(t)+1)d_l(t)+(\delta_l(t)+1)(1-d_l(t)),\forall l$仅存在一个决策变量$d_l(t)$，但调度方法在每个时隙$t$都需要做出$N$个决策来满足处理容量和处理速率的约束。同时，所做决策还需要尽可能地最小化函数$\sum_{l=1}^{N}\theta_l\delta_l(t)$。对于一个具有$N$个无线设备的MEC系统而言，每个时隙的决策空间为$2^N$，并且随着$N$的增加，决策空间以指数形式递增，因此，该调度问题很难通过动态规划的方法解决。通过深入分析处理速率的约束，可以将其递归性映射为队列的演化过程，由于Lyapunov优化技术[27]在排队系统中的能效和利益最大化决策方面具有较好的性能，本文所涉及的决策问题正好符合Lyapunov优化技术的应用场景。因此，基于Lyapunov优化技术，可以将长期处理

速率约束问题转化为复杂度较低的每个时隙静态优化问题来解决。

本文首先将衡量D_l卸载任务的处理滞后程度的性能指标定义为处理债务$Q_l(t)$，其在任意时隙t的演化过程为

$$Q_l(t)=\max\{(t-1)\lambda_l(1-\mu_l)-\sum_{k=1}^{t-1}d_l(k),0\} \tag{12}$$

其中，$(t-1)\lambda_l(1-\mu_l)$表示截止到时隙$t-1$，满足用户U_l服务需求所需的最少有效更新数据量；$\sum_{k=1}^{t-1}d_l(k)$表示截止到时隙$t-1$，用户U_l接收到来自D_l卸载任务流的实际有效更新数据量；操作符$\max\{\cdot\}$的使用是为了更加符合实际队列过程，即任意时刻队列都不可能为负。从式(12)中不难发现，较大的处理债务$Q_l(t)$反映了调度策略$\xi\in\Gamma_a$在满足D_l卸载任务处理需求方面的严重不足。

在分析处理债务$Q_l(t)$之前，本文首先需要解释一下处理债务与缓存队列之间的差异：1)缓存队列表示卸载任务在得不到所需处理服务后积压在缓存区的一种真实的队列形式，而处理债务是一种虚拟的队列形式；2)缓存队列是稳定的，原因是缓存队列中的部分任务可能超过截止期限而被丢弃，因此，缓存队列的长度不可能无限制增长，而处理债务$Q_l(t)$表示MEC对D_l卸载任务流的处理滞后程度，当采用不合适的调度方法时，可能导致处理债务$Q_l(t)$无限制增长。此外，如果一个调度方法是可行性最优的，则对于任意给定的最大允许丢包率向量$\boldsymbol{\mu}=(\mu_l^{\max})_{l=1}^N$，在某种程度上，任意到达过程必须在最大满足域$\Lambda(\boldsymbol{\mu},1]$内，才可以确保处理债务过程的循环迭代。

上述对处理债务$Q_l(t)$的引入并不能保证每个时隙用户U_l的最小状态更新数据量的需求大于实际状态更新数据量的阈值，而是仅仅赋予具有较大累计处理债务$Q_l(t)$的D_l具有较高的调度优先级。事实上，根据随机优化理论[27]的强稳定性定理和式(12)处理债务$Q_l(t)$的定义，可以得到处理债务$Q_l(t)$具有如下的特性。

定理1 通过保证处理债务$Q_l(t)$的强稳定性，来确保优化问题P中用户对每个任务流l的长期处理速率约束。

证明过程详见附录1。

根据优化问题P，本文注意到任意时隙$t+1$开始的调度策略依赖于系统时隙t的所有历史状态，具体介绍如下。

1)用户端的瞬时信息年龄$\delta_1(t),\delta_2(t),\cdots,\delta_N(t)$。

2)缓存队列的瞬时系统时间$C_1(t),C_2(t),\cdots,C_N(t)$。

3)D_l的处理债务$Q_1(t),Q_2(t),\cdots,Q_N(t)$。

因此，在进行调度方法设计之前，本文首先定义了时隙t开始时的系统状态$\boldsymbol{S}_t=\left(Q_l(t),C_l(t),\delta_l(t)\right)_{l=1}^N$。根据优化问题P可知，在每个时隙，系统状态$\boldsymbol{S}_t$决定了累计的处理债务、瞬时系统时间和瞬时信息年龄。这意味着下一时隙的决策$\boldsymbol{w}(t)$受当前时隙系统状态的影响。

2.2 基于Lyapunov优化的在线调度方法

基于定理1可知，求解问题P相当于在保证处理债务$Q_l(t)$强稳定性的同时最小化用户端的信息年龄，可以通过Lyapunov优化技术来解决。定义与处理债务$Q_l(t)$相关的Lyapunov函数$L(t)=\frac{R}{2}\sum_{l=1}^{N}[Q_l(t)]^2$，其中，$R>0$为处理债务的权重常数。从$L(t)$的定义可以看出，当$L(t)$很小时，累积的处理债务$Q_l(t)$很小。因此，通过保证$L(t)$尽可能小，来满足问题P中的处理速率约束问题，进而来提升所有用户终端的ewaAoI性能。定义Lyapunov漂移函数为

$$\Delta L(t)=\mathrm{E}\{L(t+1)-L(t)|\boldsymbol{S}_t\} \tag{13}$$

漂移函数旨在最小化Lyapunov函数每个时隙的增量来确保一个较小的$L(t)$。根据Lyapunov函数的定义，可以进一步得到如下的漂移函数

$$\Delta L(t)=\mathrm{E}\{L(t+1)-L(t)|\boldsymbol{S}_t\}=\frac{R}{2}\sum_{l=1}^{N}\mathrm{E}\{[Q_l(t+1)]^2-[Q_l(t)]^2|\boldsymbol{S}_t\} \tag{14}$$

下面推导表达式$\{[Q_l(t+1)]^2-[Q_l(t)]^2|\boldsymbol{S}_t\}$。为了获得与处理债务相关的表达式，本文首先使用式(12)单个时隙的递归过程来推导。

$$Q_l(t)=\max\{Q_l(t-1)-d_l(t-1)+\lambda_l(1-\mu_l),0\} \tag{15}$$

由式(15)推导可得

$$\begin{aligned}&\mathrm{E}\left\{[Q_l(t+1)]^2-[Q_l(t)]^2|\boldsymbol{S}_t\right\}=\\&\mathrm{E}\left\{[\max\{Q_l(t)+\lambda_l(1-\mu_l)-d_l(t),0\}]^2-[Q_l(t)]^2|\boldsymbol{S}_t\right\}\leqslant\\&\mathrm{E}\left\{[Q_l(t)+\lambda_l(1-\mu_l)-d_l(t)]^2-[Q_l(t)]^2|\boldsymbol{S}_t\right\}=\\&\mathrm{E}\left\{2Q_l(t)(2\lambda_l(1-\mu_l)-d_l(t))+[\lambda_l(1-\mu_l)-d_l(t)]^2|\boldsymbol{S}_t\right\}\overset{(a)}{\leqslant}\\&-2Q_l(t)I_l(t)\mathrm{E}\left\{w_l(t)|\boldsymbol{S}_t\right\}+2\lambda_l(1-\mu_l)Q_l(t)+1\end{aligned}\tag{16}$$

其中，(a)是基于不等式$|d_l(t)-\lambda_l(1-\mu_l)|\leqslant 1$推导所得。进一步，本文将式(16)代入式(14)可得漂移函数的上界为

$$\Delta L(t)\leqslant\sum_{l=1}^{N}-Q_l(t)I_l(t)\mathrm{E}\{w_l(t)|\boldsymbol{S}_t\}+\sum_{l=1}^{N}[\lambda_l(1-\mu_l)Q_l(t)]+\frac{RN}{2}\tag{17}$$

调度方法不仅需要满足问题P中处理速率的约束，还需要尽可能地降低用户端的信息年龄。因此，本文引入漂移加惩罚函数来解决上述问题。根据式(9)中最小化ewaAoI的目标函数，本文定义了与信息年龄相关的惩罚函数，如式(18)所示。

$$P(t)=\sum_{l=1}^{N}\theta_l\mathrm{E}\{\delta_l(t+1)|\boldsymbol{S}_t\}\tag{18}$$

通过式(18)可以看出，当惩罚函数较大时，用户端的信息年龄也较大。因此，在每个时隙，本文旨在最小化$\Delta L(t)+P(t)$的值。根据式(17)和式(18)，可获得漂移加惩罚函数的上界为

$$\begin{aligned}&\Delta L(t)+P(t)\leqslant\\&\sum_{l=1}^{N}-I_l(t)\left[RQ_l(t)+\theta_l(\delta_l(t)-C_l(t))\right]\mathrm{E}\left\{w_l(t)|\boldsymbol{S}_t\right\}+\\&\sum_{l=1}^{N}[\lambda_l(1-\mu_l)Q_l(t)+\theta_l(\delta_l(t)+1)]+\frac{RN}{2}\end{aligned}\tag{19}$$

直接最小化$\Delta L(t)+P(t)$比较困难，本文通过最小化不等式右侧来获得左侧上限最小值，进而达到优化左侧的目的。通过对式(19)右侧分析可知，$\sum_{l=1}^{N}[\lambda_l(1-\mu_l)Q_l(t)+\theta_l(\delta_l(t)+1)]+\frac{RN}{2}$的值仅与当前系统状态$S_t=(Q_l(t),C_l(t),\delta_l(t))$有关，与决策变量$w_l(t)$无关。本文定义$H_l(t)=I_l(t)\left[RQ_l(t)+\theta_l(\delta_l(t)-C_l(t))\right]$，在每个时隙$t$，调度策略选择前$m$个具有较大$H_l(t)$值的任务流进行调度处理。从$H_l(t)$的表达式不难发现，调度决策不仅与任务流的处理债务$Q_l(t)$、处理债务权重$R$和队首任务有效性$I_l(t)$相关，而且还与用户端信息年龄与系统时间的差值$\delta_l(t)-C_l(t)$以及信息年龄权重$\theta_l$有关。因此，在任意时隙$t$，对于到达率为$\lambda_l\in\Lambda(\boldsymbol{\mu},1]$的随机过程，该调度策略选择前$m$个具有较大$H_l(t)$值的任务流进行调度处理。具体的调度过程如算法1所示。

算法1　信息年龄–截止期限感知的实时任务调度

输入　无线设备–用户对N，处理容量m，处理债务权重R，任务到达率$\{\lambda_l\}_{l=1}^{N}$，最大丢包率$\{\mu_l^{\max}\}_{l=1}^{N}$，信息年龄权重$\{\theta_l\}_{l=1}^{N}$，最大截止期限$T_l^{\mathrm{Deadline}}$

输出 任务的处理调度决策$\boldsymbol{w}^*(t)$

1) 令初始系统时间和初始信息年龄分别为$C_l(1)=0,\delta_l(1)=1,\forall l$

2) for $t=1$ to T do

3)　while $\lambda_l>\mu_l$且$\lambda_l<1$ do

4)　记录任务到达时刻及数量

5)　获取每个队列的系统时间$\{C_l(t)\}_{l=1}^{N}$，并根据式(2)判断$I_l(t)$的值，记录丢包的集合

6)　为每一个任务流$l\in N$计算$H_l(t)$的值，计算式为

$$H_l(t)=I_l(t)\left[RQ_l(t)+\theta_l(\delta_l(t)-C_l(t))\right]$$

7)　将$H_l(t)$值进行降序排列，取前m个结果对应的l值构成一个集合L

8)　$w_l(t)=1,l\in L$，$w_l(t)=0,l\notin L$

9)　$\boldsymbol{w}^*(t)=\{w_l(t)\}_{l=1}^{N}$

10)　根据式(7)更新$\{\delta_l(t)\}_{l=1}^{N}$

11)　根据式(15)更新$\{Q_l(t)\}_{l=1}^{N}$

12)　根据到达率λ_l和式(6)更新$\{C_l(t)\}_{l=1}^{N}$

13)　end while

14) end for

在算法1中，由于任务到达率λ_l是服从泊松分布的随机变量，故变量$C_l(t)$和$Q_l(t)$也具有随机性。在更新过程中，较大的到达率λ_l可能导致较大的$\delta_l(t)-C_l(t)$差值和较大的$Q_l(t)$值，进一步可

能导致任务流较大的调度可能性。但是这并不意味着调度方法一定会优先调度具有较大λ_l的任务流，这是因为处理速率约束和信息年龄对不同任务流的重要程度不一样。此外，不同的处理速率约束、债务权重R、最大截止期限T_l^{Deadline}以及允许的最大丢包率$\mu_l^{\max}$都会影响$H_l(t)$的值，进而影响任务流的选择。

3 性能分析

3.1 与下限性能对比

由于本文考虑卸载任务在MEC采用FIFO队列规则排队，因此在任意给定网络参数$(N,m,\lambda_l,\theta_l,\mu_l^{\max})$的情况下很难给出目标函数的上界。基于此，本节利用样本路径参数的方法来推导问题P的下界P_L，基于本文所提方法与下限目标值的仿真对比，实现本文所提方法的性能分析。

首先基于任务的系统时间和到达间隔可以获得式(11)中目标函数$J(\xi)$的下限表达式，同时利用Johnson不等式对下限表达式进行变换，获得P_L为

$$\begin{aligned}&\text{P}_\text{L}\text{: LB} = \min\left\{\frac{1}{2N}\sum_{l=1}^{N}\theta_l\left(\frac{1}{\bar{d}_l}+1\right)\right\}\\&\text{C}_1\text{: }\sum_{l=1}^{N}\bar{d}_l \leqslant m,\forall t\\&\text{C}_2\text{: }\bar{d}_l \geqslant \lambda_l(1-\mu_l),\forall l\end{aligned}\tag{20}$$

从式(20)中不难发现，下界问题P_L仅依赖于MEC系统对任意无线设备D_l有效任务的长期处理速率$\{\bar{d}_l\}_{l=1}^{N}$，原问题P中的约束C_1和C_2仍然是必须的，只是原问题单个任务流的$\bar{d}_l$变为了总任务流的处理速率$\sum_{l=1}^{N}\bar{d}_l$。由于$\bar{d}_l \geqslant \lambda_l(1-\mu_l)$，且有$\lambda_l \in \Lambda(\boldsymbol{\mu},1],\forall l$，则有$\lambda_l(1-\mu_l) \leqslant \bar{d}_l \leqslant 1$，原问题P中的约束$\text{C}_1$等价于$\text{P}_\text{L}$问题中的$\text{C}_1$。

定理2 对于具有参数为$(N,m,\lambda_l,\theta_l,\mu_l^{\max})$的任意给定网络，问题$\text{P}_\text{L}$的最优目标值是式(11)最优值的一个下限，即式(21)的优化问题可以为原问题提供信息年龄的下限，则有$J(\xi^*) \geqslant \text{LB}$。

定理2的证明过程可以根据文献[28]中定理1的证明思路推导获得，详见附录2。为了获得问题P_L的唯一解，本文使用拉格朗日松弛和KKT（Karush-Kuhn-Tucker）条件进行分析，详细的分析过程见附录3，实现过程如算法2所示。

算法2 下限问题P_L的解

输入 无线设备–用户对N，处理容量m，任务到达率$\{\lambda_l\}_{l=1}^{N}$，最大丢包率$\{\mu_l\}_{l=1}^{N}$，信息年龄权重$\{\theta_l\}_{l=1}^{N}$

1) 根据$\gamma_l = \dfrac{\theta_l}{2N(\lambda_l)^2(1-\mu_l)^2},\forall l$，计算$\gamma_l$

2) 根据$\gamma = \max_l\{\gamma_l\}$，计算$\gamma$

3) 根据$w_l = \lambda_l(1-\mu_l)\max\left\{1;\sqrt{\dfrac{\gamma_l}{\gamma}}\right\},\forall l$，计算$w_l$

4) 根据$S = \sum_{l=1}^{N}\lambda_l(1-\mu_l)$，计算$S$

5) while $S < m$且$\gamma > 0$ do

6) 微调降低γ

7) 重复步骤3) 和步骤4) 来更新w_l和S

8) end while

9) 输出$\gamma^* = \gamma$和$\bar{w}_l^* = \bar{w}_l$

本文所提方法的性能与下限的差距可通过第4节仿真实验来进行评估。

3.2 时间复杂度分析

算法1的复杂度主要由for循环（步骤8)）和确定缓存集合L控制。for循环是对T个时隙进行迭代，因此，其迭代数为T。集合L的确定可以使用排序算法获得，其时间复杂度为$O(N\log N)$，因此，算法1的时间复杂度可以表示为$O(TN\log N)$。

4 实验验证及分析

本文所有实验均运行在相同的软硬件环境中，通过提供多种实验来验证本文所提方法的性能。首先，根据系统模型和算法需要，设置相关的模型和算法参数；其次，基于本文所要解决的问题，选出4种比较相关的基准方法；最后，通过对比本文所提方法和基准方法在不同的设备–用户对数、任务到达率以及债务权重参数的ewaAoI和处理债务性能，来验证本文所提方法的性能。

4.1 实验设置

实验场景如图1所示，一个配置有MEC服务器的EN覆盖半径为50 m的范围，N个无线设备随机地部署在该范围内，具有m个独立处理单元的MEC服务器配置有N个相同的缓存区，无线设备将顺序产生的任务以速率λ_l转发到MEC。实验的

一些基本参数设置如表2所示，此外，实验中可调参数分别为N、λ_l、$\mu_l^{\max}$、T_l^{Deadline}和R，这些参数会根据不同的实验目的在实验中进行设置。

表2　　实验参数设置

参数名称	值
独立处理单元数量m/个	3
时间域T/个时隙数	$N \times 10^5$
迭代次数I/次	10

为了评估本文所提调度方法的性能，选择4种通用的调度方法与其进行性能对比，分别为文献[19]中提出的平稳随机（RD, random）调度方法、漂移加惩罚（DPP, drift plus penalty）方法、最大权重债务优先（LWDF, largest weighted debt first）方法以及单任务队列的最大权重信息年龄下降[29]（LWAR-STQ, largest weighted AoI reduction with single task queue）方法。其中前3种采用与本文相同的FIFO队列模式，且考虑处理速率约束，最后一种采用单任务的队列模式，即队列中新到达的任务包会替代队列中已有的任务包，仅存储单个任务，且4种调度方法均不考虑截止期限限制。

1) RD。顾名思义就是不考虑系统时间和目的端AoI的状态，在满足处理速率约束的情况下，每个时隙调度方法随机地选择m个D_l卸载任务流进行处理。该调度方法在一定程度上兼顾了调度公平性和吞吐量需求的特性。

2) DPP。在每个时隙开始时，计算MEC处理之后的AoI与其对应的系统时间的加权减少量与处理债务之和$\theta_l(\delta_l(t)-C_l(t))+RQ_l(t)$。然后对所有加权减少量从高到低排序，调度方法选择前m个最大值对应的D_l卸载任务流进行处理。

3) LWDF。在每个时隙开始时，对处理债务长度按照从大到小排序，调度方法选择前m个D_l卸载任务流进行处理。

4) LWAR-STQ。在每个时隙开始时，新到达的任务替换队列中已有的任务，计算MEC处理之后信息的AoI与其对应的系统时间的加权减少量$\theta_l(\delta_l(t)-C_l(t))$，然后对所有加权减少量从高到低排序，调度方法选择前m个最大值对应的卸载任务流进行处理。

4.2　仿真结果分析

系统性能是根据系统整体的信息新鲜度和调度实时性来评估的，其中信息新鲜度由用户端接收信息的期望加权平均信息年龄来表征，而调度实时性则利用最大处理债务的大小来表征。此外，由于本文所提方法计算处理之后的信息年龄和系统时间差值，因此还增加了对任务系统时间的测量。考虑到设备–用户对数的激增会对系统ewaAoI、系统时间和处理债务产生影响，以及任务到达率的递增会对系统时间和系统ewaAoI产生影响。下面首先从这2个方面来分析ewaAoI、平均系统时间和最大处理债务的性能，之后分析债务权重参数对ewaAoI性能的影响。

1) 设备–用户对数的影响。图3~图8分别展示了不同设备–用户对数$N \in \{5,8,10,12,14,16,18,20,22,24,26,28,30\}$对系统ewaAoI、处理债务和系统时间产生的影响。图3~图5展示了在同构网络下设备–用户对数对系统ewaAoI、平均系统时间和最大标准处理债务的影响。其中同构网络考虑所有任务流具有相同的网络配置$\theta_l=1$、$\lambda_l=0.05$、$T_l^{\text{Deadline}}=10$和$\mu_l^{\max}=0.03$，$\forall l$。图6~图8展示了在异构网络下设备–用户对数对系统ewaAoI、平均系统时间和最大标准处理债务的影响，其中异构网络设置为：当$l \in \left\{1,2,\cdots,\frac{N}{2}\right\}$时，$\theta_l=1$，$\lambda_l=0.05$，$\mu_l^{\max}=0.03$，$T_l^{\text{Deadline}}=8$；当$l \in \left\{\frac{N}{2}+1,\frac{N}{2}+2,\cdots,N\right\}$时，$\theta_l=4$，$\lambda_l=0.09$，$\mu_l^{\max}=0.04$，$T_l^{\text{Deadline}}=10$。此外，设置债务权重$R=1$。

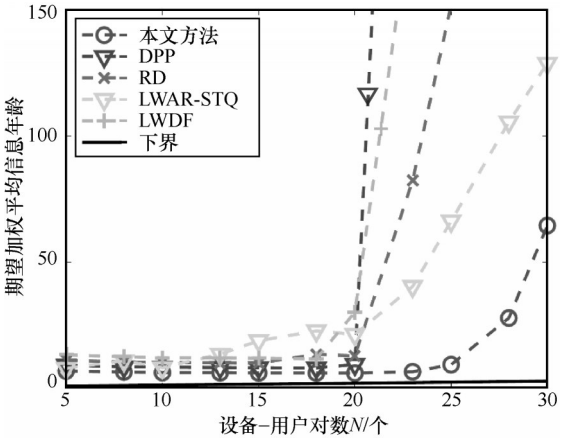

图3　在同构网络下设备–用户对数对系统ewaAoI的影响

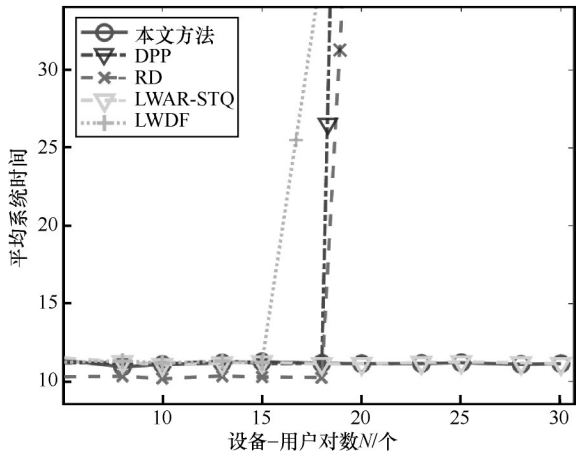

图4　在同构网络下设备–用户对数对平均系统时间的影响

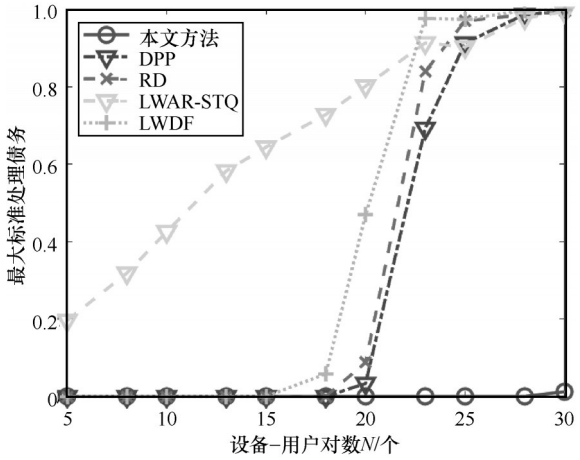

图5　在同构网络下设备–用户对数对最大标准处理债务的影响

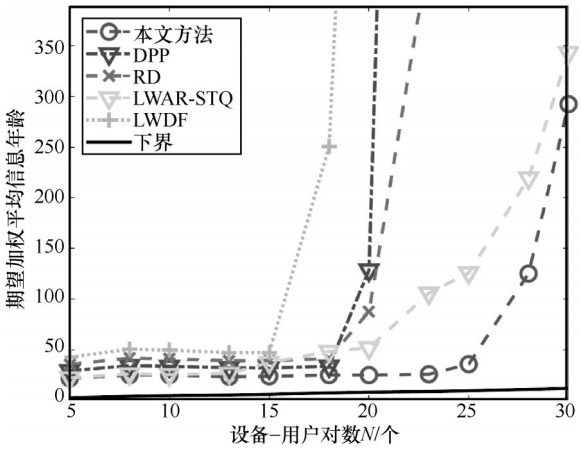

图6　在异构网络下设备–用户对数对系统ewaAoI的影响

从图3和图6的结果可以发现，随着设备–用户对数的增加，所有调度方法在$N \leqslant 16$时系统ewaAoI的性能变化不大，但当$N > 16$时，采用不同调度方法的ewaAoI的性能均出现不同程度的骤增。相较于其他调度方法，本文方法在同等网络设置下能够容忍更多的设备–用户对数。

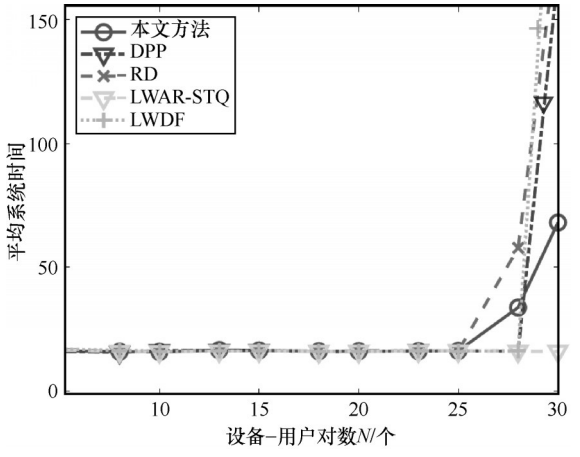

图7　在异构网络下设备–用户对数对平均系统时间的影响

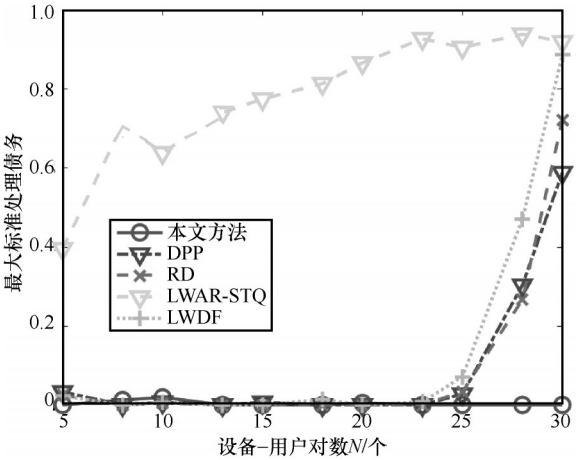

图8　在异构网络下设备–用户对数对最大标准处理债务的影响

图4和图7结果表明，在不同网络设置下，设备–用户对数对平均系统时间的影响存在很大差异。图4结果显示，当$N > 15$时，除了本文方法与具有单任务队列模式的方法LWAR-STQ，其他方法的平均系统时间都会出现骤增现象。这是因为随着无线设备–用户对数增加到一定数量，调度程序对队列中的任务调度处理不及时，导致队列中等待处理的任务不断增多，任务等待的时间也随着任务的增多而增大。而本文方法在调度之前会过滤系统时间较大的任务，因此在一定程度上会降低任务的系统时间。图7结果显示，当$N > 25$时，除了具有单任务队列模式的方法LWAR-STQ，其他方法的平均系统时间都会出现骤增现象，其中本文方法骤增的幅度最小。这进一步验证了本文方法通过调度前的过滤可以降低任务的系统时间，但同时也说明，在任意网络设置下，设备–用户对数的递增对采用FIFO队列模式方法

的系统时间产生的影响较大，只是相对于其他采用FIFO队列模式的方法，本文方法的系统时间骤增时刻更滞后。LWAR-STQ方法采用单任务包队列模式，不存在排队等待的过程，因此其系统时间最小，但是该方法不管队列中已有任务是否失效，都对其进行调度处理，也会出现处理过时任务的情况，导致有效任务调度实时性变差。同时，只要有新任务到达就替换队列中已有任务，导致丢包率变大，调度实时性进一步恶化。这一影响进一步在图5和图8中的最大标准处理债务中得到验证。

从图5和图8中可以发现，LWAR-STQ方法的调度实时性最差，相较于调度方法RD、DPP及LWDF，本文方法的处理债务（调度实时性）性能对设备–用户对数的递增敏感度滞后。

2) 任务到达率对系统性能的影响。图9和图10分别展示了不同任务到达率对系统ewaAoI和平均系统时间的影响。考虑网络具有$N=30$的设备–用户对数，设置任意任务流的到达率为$\lambda_l=\frac{N-l+1}{N}\lambda,\forall l$，其中，$\lambda\in\{0.01,0.02,\cdots,0.30\}$；设置任务流的年龄权重和丢包率分别为$\theta_l=4$、$\mu_l^{\max}=0.04$、$l\in\left\{1,2,\cdots,\frac{N}{2}\right\}$和$\theta_l=1$、$\mu_l^{\max}=0.03$、$l\in\left\{\frac{N}{2}+1,\frac{N}{2}+2,\cdots,N\right\}$；设置债务权重参数和最大截止期限分别为$R=1$和$T_l^{\text{Deadline}}=10$。

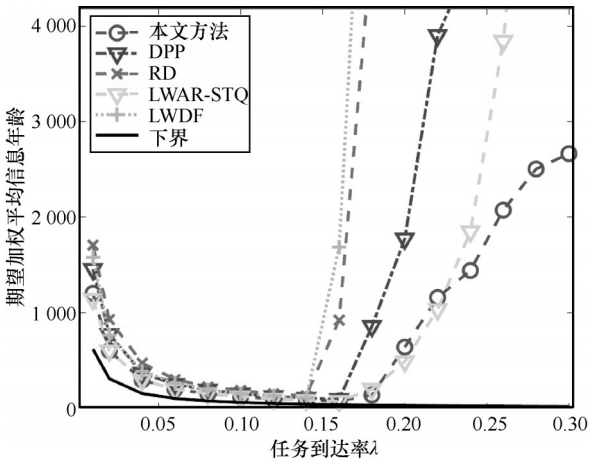

图9　任务到达率对系统ewaAoI的影响

从图9可以看出，在任务到达率超过一定值之后，所有基准调度方法下的ewaAoI性能均出现骤增现象，比如在$\lambda>0.14$时，LWDF方法最早出现ewaAoI骤增，而LWAR-STQ方法最晚出现ewaAoI骤增。尽管本文方法比LWAR-STQ方法提前发生ewaAoI递增的趋势，但是本文方法的ewaAoI性能呈现比较缓慢的增长趋势，而且系统ewaAoI增加幅度也没有LWAR-STQ方法的ewaAoI增加幅度大，因此，在同等网络设置下，本文方法可以更好地适应任务流变化带来的性能影响。

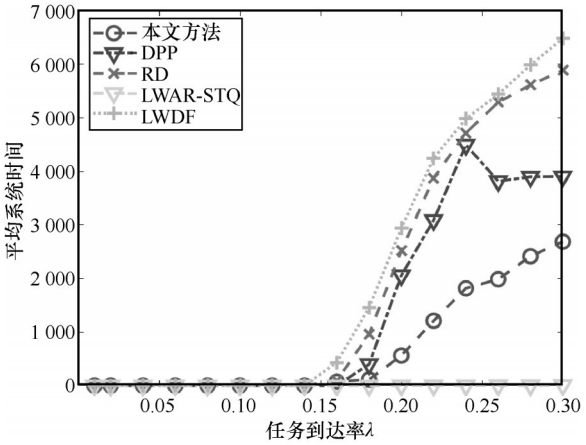

图10　任务到达率对平均系统时间的影响

从图10中可以看出，当任务到达率$\lambda\geqslant0.14$时，除了LWAR-STQ方法，所有方法的系统时间会出现不断增加的趋势，也包括本文方法。这是因为随着任务到达率的增大，有限处理资源来不及处理不断增加的任务，使得任务在队列中的等待时间变长，而本文方法尽管对任务进行截止期限判断，但是仅仅在调度过程中对所调度的任务流中的任务进行截止期限判断，未对所有任务流中的任务进行截止期限判断，导致未调度任务流中任务的平均系统时间不断增加。值得一提的是，由于LWAR-STQ方法采用单个任务包队列规则，因此其平均系统时间最小。尽管LWAR-STQ方法能保持最小的平均系统时间，但是该方法下的ewaAoI性能还会出现骤增的现象，增长幅度比本文方法要大很多。这进一步证实了本文方法在适应任务流变化所带来的ewaAoI性能方面具有更好的表现。

3) 债务权重对本文方法的ewaAoI性能影响。本文利用债务权重参数R来权衡处理约束和ewaAoI在调度过程中所占比重，R越大，表示处理约束的占比越大，即处理的实时性越重要，反之亦然。图11展示了债务权重参数R对本文方法的ewaAoI的性能

影响。考虑设备–用户对数为$N=30$的网络规模，设置$\theta_l=\frac{l}{N}$，$\mu_l^{\max}=0.35\lambda_l$，$\lambda_l=\frac{0.2(N-l+1)}{N}$，$T_l^{\text{Deadline}}=10,\forall l$，$T\in\left\{10^4,2\times10^4,\cdots,10\times10^4\right\}$和$R\in\{1,3,5,8,10,50,100,500\}$。

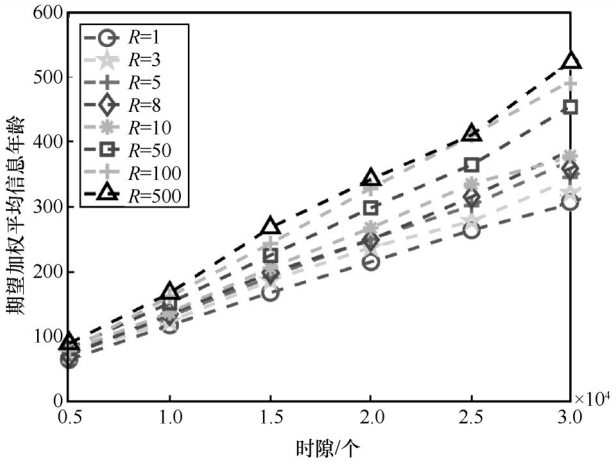

图11 债务权重参数R对本文方法的ewaAoI的性能影响

从图11可以看出，随着时间的不断增长，不同债务权重参数对ewaAoI性能的影响大致呈现线性增大的趋势，说明债务权重参数对ewaAoI的影响随着时间增长不断积累，这一点与采用后进先出队列规则的权重参数影响[19]具有很大区别；其次，随着权重参数R的增大，ewaAoI也不断变大，这也正好验证了参数R的本质作用，即R越大，越忽略对ewaAoI性能的优化，从而造成更大的ewaAoI增长；最后，当$R=1$时，债务权重对ewaAoI和实时处理需求最公平，此时，对ewaAoI性能的影响也最小。

5 结束语

本文对边缘辅助状态更新应用中实时获取新鲜信息的优化策略进行研究，考虑边缘服务器的处理容量和处理速率的约束，构建处理速率约束的最小化信息年龄的任务调度模型。利用处理债务的递归过程与队列演化过程相结合，将长期动态优化问题转化为每时隙的静态优化问题，并基于Lyapunov优化技术提出了一种复杂度较低的信息年龄–截止期限感知的任务调度方法，给出了算法的性能下界，并分析了算法的计算复杂度。仿真实验结果表明，本文所提方法具有较好的调度实时性和信息新鲜度性能。

附录1 处理债务强稳定性的证明

首先根据处理债务的演化过程式(15)可知，下一时隙D_l任务流的处理债务的长度与当前时隙的队列长度满足以下不等式约束。

$$Q_l(t+1)\geqslant Q_l(t)-d_l(t)+\lambda_l(1-\mu_l) \tag{21}$$

对式(21)两边分别在时间上进行迭代可得

$$\begin{aligned}&Q_l(t+1)\geqslant Q_l(t)-d_l(t)+\lambda_l(1-\mu_l)\\&Q_l(t)\geqslant Q_l(t-1)-d_l(t-1)+\lambda_l(1-\mu_l)\\&\quad\vdots\\&Q_l(2)\geqslant Q_l(1)-d_l(1)+\lambda_l(1-\mu_l)\end{aligned} \tag{22}$$

对式(22)左右两边分别求T个时隙和重排列，可得

$$Q_l(T+1)-Q_l(1)\geqslant T\lambda_l(1-\mu_l)-\sum_{t=1}^{T}d_l(t) \tag{23}$$

再对式(23)两边分别除以T并取极限可得

$$\lim_{T\to\infty}\frac{Q_l(T+1)-Q_l(1)}{T}\geqslant\lambda_l(1-\mu_l)-\lim_{T\to\infty}\frac{1}{T}\sum_{t=1}^{T}d_l(t) \tag{24}$$

根据队列速率稳定性的定义可知，要保证处理债务$Q_l(t)$的强稳定性，需要满足

$$\lim_{T\to\infty}\frac{Q_l(t)}{T}=0 \tag{25}$$

同时也满足

$$\lim_{T\to\infty}\frac{Q_l(0)}{T}=0 \tag{26}$$

则式(26)可以进一步推导为

$$\lim_{T\to\infty}\frac{1}{T}\sum_{t=1}^{T}w_l\geqslant\lambda_l(1-\mu_l) \tag{27}$$

由式(27)可以看出，确保处理债务$Q_l(t)$的强稳定性等价于确保处理速率的约束需求。证毕。

附录2 问题下限推导的证明

根据文献[19]中所使用的样本路径方法，在T个时隙构成的时间域内，考虑MEC网络中一个满足处理容量和处理速率约束的调度方法$\xi\in\Gamma_a$。利用样本路径的方法，对于任意给定的样本路径s，在时隙T内MEC处理D_l任务的总和为$\Omega_l(T)=\sum_{t=1}^{T}w_l(t)$，同时任意2个任务的处理时间间隔为$v_l[s]$。处理时间间隔$v_l[s]$表示MEC处理来自任务流$l$的任务$s-1$与任务$s$之间的时隙数，其中$s\in\{1,2,\cdots,\Omega_l(T)\}$，剩余未使用的时隙数为$M_l$。因此，根据以上分析有

$$T=\sum_{s=1}^{\Omega_l(T)}v_l[s]+M_l,\forall l\in\{1,2,\cdots,N\} \tag{28}$$

根据信息年龄的演化过程$\delta_l(t+1)=(C_l(t)+1)d_l(t)+(\delta_l(t)+1)(1-d_l(t))$可知，$\delta_l(t+1)$的值将按照序列$\{C_l(s-1)+1,C_l(s-1)+2,\cdots,C_l(s-1)+v_l[s]\}$进行演化，这种演化模式在整个时间域$T$内不断重复，因此，目的端接收来自任务流$l$的平均信息年龄为

$$\frac{1}{T}\sum_{t=1}^{T}\delta_l(t)=\frac{1}{T}\left[\sum_{s=1}^{\Omega_l(T)}\frac{(2C_l[s-1]+1+v_l[s])v_l[s]}{2}+C_l[\Omega_l(T)]M_l+\frac{(M_l+1)M_l}{2}\right]=\frac{1}{2}\left[\frac{\Omega_l(T)}{T}\frac{1}{\Omega_l(T)}\sum_{s=1}^{\Omega_l(T)}(v_l^2[s]+2C_l[s-1]v_l[s])+\frac{2C_l[\Omega_l(T)]M_l+M_l^2}{T}+1\right] \tag{29}$$

定义任意集合s的样本期望的算子为

$$\bar{\mathbb{Z}}[v_l]=\frac{1}{\Omega_l(T)}\sum_{s=1}^{\Omega_l(T)}v_l[s] \tag{30}$$

$$\bar{\mathbb{Z}}[v_l^2]=\frac{1}{\Omega_l(T)}\sum_{s=1}^{\Omega_l(T)}v_l^2[s] \tag{31}$$

将$\bar{\mathbb{Z}}[v_l^2]$的值代入式(29)，考虑到式(29)中$2C_l[s-1]v_l[s]$和$2C_l[\Omega_l(T)]M_l$非负的特性，利用Johnson不等式$\bar{\mathbb{Z}}[v_l^2]\geqslant(\bar{\mathbb{Z}}[v_l])^2$，从而可得如下的不等式

$$\frac{1}{T}\sum_{t=1}^{T}\delta_l(t)\geqslant\frac{1}{2}\left(\frac{\Omega_l(T)}{T}(\bar{\mathbb{Z}}[v_l])^2+\frac{M_l^2}{T}+1\right) \tag{32}$$

根据式(28)和式(30)样本期望定义，可得如下等式

$$\frac{T}{\Omega_l(T)}=\frac{\sum_{s=1}^{\Omega_l(T)}v_l[s]+M_l}{\Omega_l(T)}=\bar{\mathbb{Z}}[v_l]+\frac{M_l}{\Omega_l(T)} \tag{33}$$

将式(33)中的$\bar{\mathbb{Z}}[v_l]$代入式(32)中，可得不等式

$$\frac{1}{T}\sum_{t=1}^{T}\delta_l(t)\geqslant\frac{1}{2}\left(\frac{1}{T}\frac{(T-M_l)^2}{\Omega_l(T)}+\frac{M_l^2}{T}+1\right) \tag{34}$$

通过对式(34)右侧变量M_l进行分析并最小化可得

$$\frac{1}{T}\sum_{t=1}^{T}\delta_l(t)\geqslant\frac{1}{2}\left(\frac{M_l}{\Omega_l(T)+1}+1\right) \tag{35}$$

对式(35)取期望并再一次利用Johnson不等式对其简化可得

$$\frac{1}{T}\sum_{t=1}^{T}\mathrm{E}[\delta_l(t)]\geqslant\frac{1}{2}\left(\frac{1}{\left[\frac{\Omega_l(T)}{T}\right]+\frac{1}{T}}+1\right) \tag{36}$$

对式(36)两端的T取极限可得

$$\lim_{T\to\infty}\frac{1}{T}\sum_{t=1}^{T}[\delta_l(t)]\geqslant\frac{1}{2}\left(\frac{1}{\bar{w}_l}+1\right) \tag{37}$$

将式(37)代入式(9)可得

$$\lim_{T\to\infty}\frac{1}{NT}\left[\sum_{t=1}^{T}\sum_{l=1}^{N}\theta_l\delta_l(t)\right]=\lim_{T\to\infty}\frac{1}{N}\theta_l$$

$$\mathrm{E}\left[\frac{1}{T}\sum_{t=1}^{T}\delta_l(t)\right]\geqslant\frac{1}{2N}\sum_{l=1}^{N}\theta_l\left(\frac{1}{\bar{w}_l}+1\right) \tag{38}$$

证毕。

附录3　定理2的证明

对于下限问题$\mathrm{P_L}$，其唯一解可以表示为$\bar{w}_l^{\mathrm{LB}}=\min\left\{\lambda_l,\sqrt{\frac{\theta_l}{2N\gamma^*}}\right\},\forall l$，其中$\gamma^*$的值可以通过算法2获得。本文首先利用拉格朗日松弛方法定义如下的拉格朗日函数

$$L(\bar{w}_l,\beta_l,\gamma)=\frac{1}{2N}\sum_{l=1}^{N}\theta_l\left(\frac{1}{\bar{w}_l}+1\right)+\sum_{l=1}^{N}\beta_l(\lambda_l(1-\mu_l)-\bar{w}_l)+\gamma\left(\sum_{l=1}^{N}w_l-m\right) \tag{39}$$

其中，$\{\beta_l\}_{l=1}^{N}$是与松弛$\bar{w}_l\geqslant\lambda_l(1-\mu_l)$相关的KKT乘数，$\gamma$是与松弛$\sum_{l=1}^{N}w_l(t)\leqslant m$相关的KKT乘数。同时定义$L(\bar{w}_l,\beta_l,\gamma)=+\infty$，然后根据KKT条件，有以下特性。

1) 平稳性：$L_{\bar{w}_l}(\bar{w}_l,\beta_l,\gamma)=0$。

2) 互补松弛：$\beta_l(\lambda_l(1-\mu_l)-\bar{w}_l)=0$；$\gamma\left(\sum_{l=1}^{N}\bar{w}_l-m\right)=0$。

3) 原始可行性：$\bar{w}_l\geqslant\lambda_l(1-\mu_l),\forall l$和$\sum_{l=1}^{N}w_l(t)\leqslant m,\forall t$。

4) 对偶可行性：$\beta_l\geqslant0,\forall l$和$\gamma\geqslant0$。

考虑到速率约束$\lambda_l(1-\mu_l)$是严格正实数，由于$L(\bar{w}_l,\beta_l,\gamma)$是一个凸函数，如果存在一个向量$(\{\bar{w}_l^*\}_{l=1}^{N},\{\beta_l^*\}_{l=1}^{N},\gamma^*)$能够满足KKT条件，则该向量是唯一的。因此，优化$\mathrm{P_L}$问题的调度策略$\xi^*\in\Gamma_a$也是唯一的，并且可以用$\{\bar{w}_l^*\}_{l=1}^{N}$来刻画，从而可以得到唯一的向量$(\{\bar{w}_l^*\}_{l=1}^{N},\{\beta_l^*\}_{l=1}^{N},\gamma^*)$。

为了评估平稳性，首先需要计算$L(\bar{w}_l,\beta_l,\gamma)$对$\bar{w}_l$的偏导数，可得如下等式

$$\frac{\theta_l}{2N(\bar{w}_l)^2}+\beta_l=\gamma \tag{40}$$

根据互补松弛$\gamma\left(\sum_{l=1}^{N}\bar{w}_l-m\right)=0$可知，$\gamma=0$或者$\sum_{l=1}^{N}\bar{w}_l=m$。然而根据式(40)，只有当$\beta_l=0$且$\bar{w}_l\to\infty$时，$\gamma$才可能为0，但这与$\bar{w}_l\in(\lambda_l(1-\mu_l),1]$相违背，因此有

$$\gamma\geqslant0\text{ 且 }\sum_{l=1}^{N}\bar{w}_l=m \tag{41}$$

基于对偶可行性$\beta_l\geqslant0$，本文可以将任务流$l\in\{1,2,\cdots,N\}$划分为两类。1) $\beta_l>0$的任务流，根据互补松弛$\beta_l(\lambda_l(1-\mu_l)-\bar{w}_l)=0$可知$\bar{w}_l=\lambda_l(1-\mu_l)$，将$\bar{w}_l$的值代入式(41)有不等式$\beta_l=\gamma-\gamma_l>0$，其中定义常数为

$$\gamma_l=\frac{\theta_l}{2N(\lambda_l)^2(1-\mu_l)^2} \tag{42}$$

2) $\beta_l=0$的任务流，根据式(40)可得

$$\gamma = \gamma_l \left(\frac{\lambda_l (1 - \mu_l)}{\bar{w}_l} \right)^2 \Rightarrow \bar{w}_l = \lambda_l (1 - \mu_l) \sqrt{\frac{\gamma_l}{\gamma}}, \gamma - \gamma_l < 0 \quad (43)$$

因此，对于固定值$\gamma \geqslant 0$，如果$\gamma \geqslant \gamma_l$，则任务流$l$属于第一类，否则任务流$l$属于第二类。两类中与任务流$l$相关的$\beta_l$和$\bar{w}_l$值可以表示为

$$\beta_l = \max \{ 0 ; \gamma - \gamma_l \} , \forall l \quad (44)$$

$$\bar{w}_l = \lambda_l (1 - \mu_l) \max \left\{ 1 ; \sqrt{\frac{\gamma_l}{\gamma}} \right\}, \forall l \quad (45)$$

基于上面的分析可知，当$\gamma < \max \{ \gamma_l \}$时，所有的任务流都属于第二类，并且有$\bar{w}_l > \lambda_l (1 - \mu_l), \forall l$。通过不断降低$\gamma$的值，使每个任务流的处理速率$\bar{w}_l$保持恒定或增加。本文设计的目的是找到最优的$\gamma^*$使由此获得的$\{ \bar{w}_l^* \}_{l=1}^N$能够满足$\sum_{l=1}^{N} \bar{w}_l^* = m$约束。基于以上分析，本文设计算法2来获得KKT条件唯一解向量$(\{ \bar{w}_l^* \}_{l=1}^N, \{ \beta_l^* \}_{l=1}^N, \gamma^*)$，从而获得$\mathrm{P_L}$问题的唯一解。证毕。

参考文献：

[1] ANDREWS J G, BUZZI S, CHOI W, et al. What will 5G be?[J]. IEEE Journal on Selected Areas in Communications, 2014, 32(6): 1065-1082.

[2] KAUL S, GRUTESER M, RAI V, et al. Minimizing age of information in vehicular networks[C]//Proceedings of the 2011 8th Annual IEEE Communications Society Conference on Sensor, Mesh and Ad Hoc Communications and Networks. Piscataway: IEEE Press, 2011: 350-358.

[3] YATES R D, SUN Y, BROWN D R, et al. Age of information: an introduction and survey[J]. IEEE Journal on Selected Areas in Communications, 2021, 39(5): 1183-1210.

[4] KOSTA A, PAPPAS N, ANGELAKIS V. Age of information: a new concept, metric, and tool[J]. Foundations and Trends® in Networking, 2017, 12(3): 162-259.

[5] YATES R D, KAUL S K. The age of information: real-time status updating by multiple sources[J]. IEEE Transactions on Information Theory, 2019, 65(3): 1807-1827.

[6] WANG H Y, SUN Q B, WANG S G. A survey on the optimisation of age of information in wireless networks[J]. International Journal of Web and Grid Services, 2023, 19(1): 1-33.

[7] CHEN X R, GATSIS K, HASSANI H, et al. Age of information in random access channels[J]. IEEE Transactions on Information Theory, 2022, 68(10): 6548-6568.

[8] DAS A K, ROY S, BANDARA E, et al. Securing age-of-information (AoI) -enabled 5G smart warehouse using access control scheme[J]. IEEE Internet of Things Journal, 2023, 10(2): 1358-1375.

[9] WANG S H, CHEN M Z, YANG Z H, et al. Distributed reinforcement learning for age of information minimization in real-time IoT systems[J]. IEEE Journal of Selected Topics in Signal Processing, 2022, 16(3): 501-515.

[10] LI J, WANG J P, CHEN Q, et al. Digital twin-enabled service satisfaction enhancement in edge computing[C]//Proceedings of the IEEE Conference on Computer Communications. Piscataway: IEEE Press, 2023: 1-10.

[11] 张宇明, 徐连明, 印思源, 等. 面向信息年龄的应急无人机网络低能耗信息采集和传输调度机制[J]. 通信学报, 2023, 44(7): 1-13.
ZHANG Y M, XU L M, YIN S Y, et al. AoI-oriented low-energy-consumption information collection and transmission scheduling mechanism for emergency UAV networks[J]. Journal on Communications, 2023, 44(7): 1-13.

[12] KAUL S, YATES R, GRUTESER M. Real-time status: how often should one update? [C]//2012 Proceedings IEEE Conference on Computer Communications. Piscataway: IEEE Press, 2012: 2731-2735.

[13] HU Y C, PATEL M, SABELLA D, et al. Mobile edge computing—a key technology towards 5G[J]. ETSI White Paper, 2015, 11(11): 1-16.

[14] DEMERS A, KESHAV S, SHENKER S. Analysis and simulation of a fair queueing algorithm[J]. ACM SIGCOMM Computer Communication Review, 1989, 19(4): 1-12.

[15] LIU J, MAO Y Y, ZHANG J, et al. Delay-optimal computation task scheduling for mobile-edge computing systems[C]//Proceedings of the 2016 IEEE International Symposium on Information Theory (ISIT). Piscataway: IEEE Press, 2016: 1451-1455.

[16] NING Z L, DONG P R, WEN M W, et al. 5G-enabled UAV-to-community offloading: joint trajectory design and task scheduling[J]. IEEE Journal on Selected Areas in Communications, 2021, 39(11): 3306-3320.

[17] MENG J Y, TAN H S, LI X Y, et al. Online deadline-aware task dispatching and scheduling in edge computing[J]. IEEE Transactions on Parallel and Distributed Systems, 2020, 31(6): 1270-1286.

[18] KADOTA I, UYSAL-BIYIKOGLU E, SINGH R, et al. Minimizing the age of information in broadcast wireless networks[C]//Proceedings of the 2016 54th Annual Allerton Conference on Communication, Control, and Computing (Allerton). Piscataway: IEEE Press, 2016: 844-851.

[19] KADOTA I, SINHA A, MODIANO E. Optimizing age of information in wireless networks with throughput constraints[C]//Proceedings of the IEEE Conference on Computer Communications. Piscataway: IEEE Press, 2018: 1844-1852.

[20] WHITTLE P. Restless bandits: activity allocation in a changing world[J]. Journal of Applied Probability, 1988, 25(A): 287-298.

[21] GONG J, KUANG Q B, CHEN X. Joint transmission and computing scheduling for status update with mobile edge computing[C]//Proceedings of the IEEE International Conference on Communications (ICC). Piscataway: IEEE Press, 2020: 1-6.

[22] ZHONG J, ZHANG W Y, YATES R D, et al. Age-aware scheduling for asynchronous arriving jobs in edge applications[C]//Proceedings of the IEEE Conference on Computer Communications. Piscataway: IEEE Press, 2019: 674-679.

[23] ZHU T X, SHI T, LI J Z, et al. Task scheduling in deadline-aware mo-

bile edge computing systems[J]. IEEE Internet of Things Journal, 2019, 6(3): 4854-4866.

[24] LOU J, TANG Z Q, ZHANG S L, et al. Cost-effective scheduling for dependent tasks with tight deadline constraints in mobile edge computing[J]. IEEE Transactions on Mobile Computing, 2023, 22(10): 5829-5845.

[25] WANG F, XU J, WANG X, et al. Joint offloading and computing optimization in wireless powered mobile-edge computing systems[J]. IEEE Transactions on Wireless Communications, 2018, 17(3): 1784-1797.

[26] HOU I H, BORKAR V, KUMAR P R. A theory of QoS for wireless[C]// Proceedings of the IEEE Conference on Computer Communications. Piscataway: IEEE Press, 2009: 486-494.

[27] NEELY M J. Stochastic network optimization with application to communication and queueing systems[M]. Berlin: Springer, 2010.

[28] ALABBASI A, AGGARWAL V. Joint information freshness and completion time optimization for vehicular networks[J]. IEEE Transactions on Services Computing, 2022, 15(2): 1118-1129.

[29] SUN J Z, WANG L H, JIANG Z Y, et al. Age-optimal scheduling for heterogeneous traffic with timely throughput constraints[J]. IEEE Journal on Selected Areas in Communications, 2021, 39(5): 1485-1498.

[作者简介]

王红艳（1987– ），女，内蒙古呼和浩特人，北京邮电大学博士生，主要研究方向为边缘计算、信息年龄。

孙其博（1975– ），男，河南郑州人，博士，北京邮电大学研究员，主要研究方向为网络服务与智能、物联网应用技术。

马骁（1990– ），女，山东德州人，博士，北京邮电大学讲师，主要研究方向为移动云计算、移动边缘计算。

周傲（1987– ），女，湖南衡阳人，博士，北京邮电大学副教授，主要研究方向为边缘计算、云计算。

王尚广（1982– ），男，河南周口人，博士，北京邮电大学教授，主要研究方向为服务计算、移动边缘计算与智能、5G/6G核心网等。