复杂环境下信道知识图谱的更新策略

Ting Wang、Chiya Zhang、Chang Liu、Zhuoyuan Hao、Rubing Han、Weizheng Zhang 和 Chunlong He

arXiv:2512.15154v1 [cs.CE] 17 Dec 2025

摘要—**信道知识图谱（CKM）将位置信息映射为信道状态信息，借助环境知识减少第六代网络中的信令开销。然而，构建可靠的 CKM 需要大量数据和计算资源；在动态环境中，预先构建的 CKM 会逐渐过时，导致性能下降。频繁重新训练可以恢复准确性，但会造成大量浪费，因此 CKM 效能与更新开销之间存在根本性的权衡。为此，我们提出地图效能函数（MEF），同时刻画渐进老化和突发环境变化，并将更新调度问题建模为分式规划。我们开发了两种基于 Dinkelbach 的算法：Delta-P 可保证全局最优性，而 Delta-L 以近线性复杂度实现近似最优性能。对于不可预测的环境，我们推导出一种基于阈值的策略：当环境退化率超过资源消耗加速率时，立即更新是最优选择；否则，延迟更新更可取。对于可预测的环境，长期策略会有针对性地放宽这些短视规则，以最大化全局性能。在这一机制下，该策略表明，较大的初始损失和较快的衰减倾向于立即更新，而较小的初始损失和较慢的衰减则倾向于延迟更新。**

索引词—**信道知识图谱、6G无线网络、信息新鲜度、更新调度、Dinkelbach算法、动态规划、信息年龄、分式规划**

一、引言

迈向第六代（6G）无线网络的演进，以部署超大规模天线阵列并容纳海量用户为特征 [1]——[4]。这一趋势给实时获取信道状态信息（CSI）带来了前所未有的挑战，传统信道估计方案存在计算复杂度过高和信令开销过大的问题。为应对这一挑战，信道知识图谱（CKM）已成为一种很有前景的信道预测范式。通过构建无线环境的数字孪生并利用离线训练，CKM能够智能感知传播环境，并快速在线响应信道参数估计需求，从而显著降低实时测量的开销 [5]。大量研究致力于构建高保真度CKM。许多现有研究着重利用详细的环境信息（例如散射体的几何形状和材料属性）来提高预测精度 [6]。事实证明，这些方法在相对静态的环境中效果良好。

CKM的效能与存储的地图数据和实际物理传播环境之间严格对应密切相关。具体而言，CKM的准确性在很大程度上取决于数据的数量和时效性，而后者会受到CKM应用场景中环境变化的影响。无线环境通常是动态的，可能会因新建筑物的建造、大型车辆的移动，甚至植被的季节性变化而改变。这些时间变化可能导致现有CKM过时，进而严重降低预测性能 [7]。因此，需要定期进行CKM更新以维持预测精度，但这也带来了更新管理方面的新挑战：处理用于CKM更新的大规模数据集会产生高昂的计算开销。

近期研究探讨了如何 在环境变化发生时更新CKM。例如，增量学习技术已被用于在环境变化（ [7],例如旧结构消失）时使 CKM 适应新知识。然而，传统的增量学习可能会遭遇灾难性遗忘，即随着新数据不断纳入，模型对旧知识的记忆会逐渐淡化。为缓解这一问题， [8],提出了一种基于遗忘的微调机制；研究表明，该机制在适应新的环境特征之前有选择地移除过时信息，因而表现更优。尽管这些方法能够有效地使 CKM 适应新环境，何时触发此类更新，才能在性能提升与成本开销之间取得平衡，仍是一个悬而未决的问题。

尽管这些刷新机制取得了进展，但仍存在一个关键空白：何时应触发刷新——

（通讯作者：Chiya Zhang。）本研究得到中国国家自然科学基金项目（编号：62394294 和 62394290）的资助。T. Wang、C. Zhang、C. Liu、R. Han 和 W. Zhang 均隶属于哈尔滨工业大学（深圳）信息科学与技术学院，地址：中国深圳，邮编 518055。Z. Hao 隶属于哈尔滨工业大学（深圳）计算机科学与技术学院，地址：中国深圳，邮编 518055。C. He 隶属于深圳大学智能信息处理广东省重点实验室，地址：中国深圳，邮编 518060。

与侧重于自适应算法的“如何”问题不同，“何时”问题涉及资源约束下的战略决策。在典型的城市场景中，环境变化发生在多个时间尺度上，而每次CKM 刷新都需要大量训练数据。例如，针对单个基站重新训练基于神经网络的 CKM，可能需要收集数千次信道测量数据，并消耗边缘服务器上数分钟的计算资源。这造成了一个不容忽视的权衡：**频繁刷新能够保持准确性，但会造成大量浪费；相反，延迟更新可以减少浪费，却会降低预测性能，直接影响系统吞吐量和可靠性**。

信道知识图谱（CKM）更新调度问题与信息新鲜度管理有相似之处。在信息更新系统领域，信息年龄（AoI）及其变体已被广泛用作度量指标，以量化信息的新鲜度和价值 [9], [10]。这些度量指标对于设计面向时敏数据的高效调度策略至关重要。然而，传统的基于信息年龄的度量指标主要衡量自上次更新以来经过的时间，无法直接反映外部世界变化导致的系统性能下降。它们并不适用于信道知识图谱更新问题，因为无法表征环境动态对通信系统性能的影响。

为弥补这一不足，我们引入地图效能函数（MEF），将环境动态作为隐藏变量纳入考量，以评估CKM效用。基于此，我们将信道知识图谱更新调度问题建模为一个优化问题，旨在最大化CKM的长期平均MEF，同时最小化更新的平均资源消耗。该建模形成一个分式规划问题。为求解该问题，我们提出一种基于Dinkelbach方法的两层迭代算法，以找到最优更新策略。我们的工作率先提出了一种系统化方法，用于确定CKM的更新时机，在复杂无线环境中平衡其运行性能与维护成本。

A. 相关工作

1) 无线通信中的信道知识图谱：CKM 作为一个综合数据库，将地理位置与多维信道属性关联起来，从而支持面向环境的 6G 网络 [11], [12]。现有关于信道知识图谱构建的研究大致分为两类：利用物理传播原理的基于模型的方法[13] ，以及采用机器学习的数据驱动技术 [14]。然而，大多数基础性工作都假设环境近乎静态，限制了其实用性。近期研究通过统计假设检验检测环境变化 [15]，或将环境分解为近乎静态和动态组成部分 [16]，以解决信道知识图谱的维护问题。

2）信息更新系统更新策略：信息年龄（AoI）已成为量化数据时效性的关键度量指标 [17],，在各种约束条件下的调度策略方面已有大量研究 [18]–[20]。然而，经典信息年龄与内容无关，因此催生了错误信息年龄（AoII）以及上下文感知信息价值（VoCAI）等上下文感知指标 [21], [22]。对于CKM而言，这些指标存在不足，原因在于：（1）CKM的价值源于系统级预测准确率，而非源不确定性；（2）我们的MEF能够捕捉因环境变化导致的失同步所引起的性能下降；（3）CKM更新涉及相当长的停机时间，在此期间系统效用为零。

3）用于分式规划的*Dinkelbach*算法：分式规划的目标表示效率比，在无线资源分配中十分常见 [23]。Dinkelbach 算法通过迭代将此类问题转化为易于求解的相减形式，近期也已应用于AoI优化 [24]。我们采用这一框架，将CKM更新调度建模为最大化系统效用与更新成本之比。

以往研究 [25], [26]探讨了单个周期内的最优更新时间，这对应于我们第 IV 节中的短期策略。本文系统地研究在动态环境中何时 更新 CKM，并通过分式规划解决 CKM性能与重新训练成本之间的根本权衡。

B. 技术挑战

具体而言，本文应对以下技术挑战：

1）更新问题缺乏对环境动态的建模。为平衡CKM效能与更新能耗，我们定义了一个 MEF 函数，用于量化复杂环境中 CKM 随时间变化的效用，并据此将信道知识图谱更新调度问题表述为长期优化问题。

2）所构建的问题是一个带有未固定变量的分式调度问题。为解决该问题，我们提出一种基于Dinkelbach算法的带Pareto前沿长期轨迹算法（**Delta-P**），该算法将双参数 Dinkelbach算法与帕累托前沿动态规划相结合，并传播边可加三元组（F，G，C）。我们证明了该算法在离散候选集上的全局最优性、有限终止性，以及相对于连续最优解的 $O(\Delta)$ 网格近似误差。

3) 为降低复杂度，我们提出一种由 Dinkelbach 驱动的长期轨迹算法线性化方法（**Delta-L**），采用泰勒线性化将内部问题简化为单权重 DAG 上的最长路径优化，

表 I 符号与记号

符号	范围/单位	含义
T_{end}	time	时域结束时间
$\int_j = [\tau_j, \tau_{j+1})$	time	更新段 j
$f(t)$	$[0,1]$	MEF
$D(c)$	> 0	时间 c的更新阻力
$C(c)$	≥ 0	时间 c的完成成本
$S = \{c_m\}$	–	完成时间集合，路径 DAG
$\mathcal{D}(S)$	set	停机时间集合
$\mathcal{W}(S)$	set	工作时间集合
$F(S)$	–	$\int_{\mathcal{W}(S)} f(t)\ dt$
$G(S)$	time	测量($\mathcal{W}(S)$)
$C_{\mathrm{tot}}(S)$	–	$\sum_m C(c_m)$
H	time	T_{end}
$J(S)$	–	目标：平均疗效，持续时间为 wo运行时间　min平均值 cost

在可行边数量上实现近线性时间，同时保持较高的解质量。

4) 对于不可预测环境，我们利用洛必达法则推导出一种基于阈值的短视策略：当环境变化率超过更新成本的增长率时，立即更新是最优的；否则，可以计算等待时间。我们进一步刻画了由于较大的初始损失和较长的后续区段，长期策略偏离短期策略的情形。

II. 系统模型与问题构建

我们考虑在受控测试平台或周期性场景中，一座基站在有限规划时域内服务于一项固定网络任务，且环境变化可被刻画。CKM 提供信道预测以支持任务执行。CKM 必须定期通过数据重新采集和模型重新训练进行更新。关键在于，每次更新都会产生不可忽略的持续时间和运营成本，在此期间任务暂停，且不会累积任何效用。遵循 [27],每次更新均纳入所有新采集的数据这一做法，预测误差会随更新频率的提高而单调下降。我们采用这种全量更新方案，以确保信道知识图谱效用函数 $f(t)$ 随自上次更新以来的年龄 t单调递减，从而便于对单周期优化问题进行解析。

表 I 总结了所使用的重要符号。

A. *MEF*：一种新的 *CKM* 时效性与环境状态度量指标

CKM 中的训练和预测过程可以建模为一个虚拟队列，其中包含“到达”和“离开”这两个关键事件。在该模型中，CKM 的信息年龄表示当前用于 CKM 的最新信道状态信息被采集以来经过的时间。具体而言，信息年龄衡量信道状态信息数据的“新鲜度”，反映了从采集信道状态信息到其经过训练和预测处理，并最终应用于通信网络之间的延迟。

将数据包 n 的到达时间和离开时间分别记为$T^A(n)$和 $T^D(n)$。 D是包括T_{train}、 T_{wait2} 和 $T_{predict}$在内的处理时间。由于数据包大小和训练过程不同，对于特定数据包而言， D 可能不是常数。

然而，尽管信息年龄提供了自上次更新以来所经过时间的线性度量指标，但它可能不足以捕捉环境中的动态变化。我们提出使用信息年龄的函数 $f(t) \in [0,1]$，称为地图效能函数（MEF），作为我们的优化度量指标；与动态环境中的 CKM 价值相比，该函数能够更好地反映 CKM 的价值。我们将 $f(t)$定义为：使用当前的、可能已经过时的 CKM 所获得的预期任务性能比，相对于假设在时间 t刷新得到的理想 CKM 而言。

$$f(t) := \frac{\mathbb{E}[U(\pi(M_0)), \Theta_t]}{\mathbb{E}[U(\pi(M^*)), \Theta_t]} \tag{1}$$

这里， M_0 表示当前正在使用的 CKM， $M^\star$表示在时间 t瞬时刷新时所能获得的理想 CKM。映射 $\pi(\cdot)$将 CKM 转换为适用于该任务的最优网络策略，而 $U(\pi,\ \Theta)$是在环境状态 Θ_t下对应的任务效用。期望值 $\mathbb{E}[\cdot]$ 是在给定 Θ_t的条件下，对小尺度随机性取的。

$f(t)$会随着上次更新完成后的时间推移而降低。图1展示了 CKM 在若干环境区段中的信息年龄和 MEF。信息年龄随时间线性增长，并在每次更新后重置为初始值。MEF则随时间推移而降低，反映 CKM 的价值逐渐衰减；它也会在更新时重置，并在更新期间保持不变。如果没有重置，环境的突然变化会使 MEF 在延迟一段时间后按照另一种函数衰减。因此，与信息年龄相比，MEF 不仅能更好地反映信息的老化，也能更好地反映环境的变化。

B. 环境动态模型

由于基站、天气、人群等可能突然出现或消失，通信环境随之变化，CKM 在该环境中的效能损失并不总是以相同方式变化。为通信环境定义一个相关时间，在该时间内，感兴趣的时间尺度上的二阶信道统计基本保持不变。因此，规划时域根据相关时间划分为 M 个区段，

$$\int_j = [\tau_j, \tau_{j+1}), 0 = \tau_1 < \tau_2 < \cdots < \tau_{M+1} = T_{\mathrm{end}}, \tag{2}$$

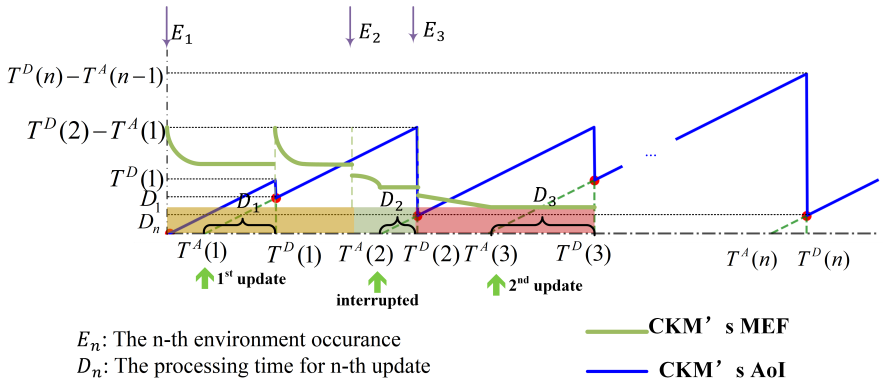

图1。信息年龄和 MEF 在更新周期中的演变。

分段边界用于模拟突变，例如 LOS↔NLOS 转换、天气变化的开始与结束，以及车辆或行人流量激增。

MEF 的公式可以通过离线采样进行拟合，但对于下文的理论分析和仿真，我们采用解析模型。我们假设，在片段 j内，老化遵循指数衰减至下限的规律，原因有二：(1) 指数衰减$e^{-\lambda_j s}$ 能够自然地描述 CKM 与不断演变的环境之间逐渐失步的过程，这在信道预测系统中十分常见，因为随着信息陈旧程度增加，估计误差会呈指数增长 [27]；(2) 下限 η_j 捕捉了即使存在时间变化仍然有效的持久环境信息（例如静态建筑几何结构），从而避免效能完全消失。在实际场景中，也可以使用其他衰减函数，而不会影响求解方法。

$$E_j(s) \;=\; \eta_j + (1-\eta_j)\, e^{-\lambda_j s}, \tag{3}$$

$$\lambda_j = \frac{\ln 2}{T_{\mathrm{half},j}}, \quad \eta_j \in [0,1), \tag{4}$$

其中，$T_{\mathrm{half},\ j}$ 表示半衰期，η_j 表示反映长期有效信息的持久下限。 $E_j(s)$描述了在 s_j开始时生成的 CKM 的老化过程。

为刻画环境更新时未能及时更新 CKM 的影响，若跨越边界 τ_ℓ时未能在该边界处完成更新，则对效能施加瞬时乘法边界冲击$s_\ell \in (0,1]$，以反映状态切换时模型失配的突然影响。多个连续未刷新的边界会产生乘法累积效应。形式化地，令 ℓ 为式（2）中定义的分段边界索引，其中 τ_ℓ 表示第 ℓ 个边界时间，定义进入损失过程

$$L(t;S) \;=\; \prod_{\ell:\ \tau_\ell \in (t_{\mathrm{last}},\, t]} s_\ell, \tag{5}$$

每当刷新恰好在边界处完成时，该值就会重置为 1。

在时间 c 完成的刷新会占用一段停机时间区间 $[c-D(c)\ \ c)$，在此期间，CKM流程无法执行该任务，因此不会产生任何效能。停机时间 $D(c)$ >0 和成本 $C(c)$ ≥ 0 可以依赖于区段，其中 $j(c)$是包含 c的区段。

调度是一个有限且严格递增的刷新完成时间集合，

$$S = \{c_1 < \cdots < c_K\} \subset [0, T_{\mathrm{end}}], \tag{6}$$

且停机时间不得重叠：

$$c_1 \geq D(c_1), c_m - c_{m-1} \geq D(c_m), \quad m = 2, \ldots, K. \tag{7}$$

任务仅在 c_m时看到新的 CKM；导致 c_m 的停机时间为 $[c_m - D(c_m)\ \ c_m)$。如果 $c_m = \tau_\ell$（恰好位于边界处），则下一分段以年龄 0 和进入损失 1 开始；如果未完成更新便跨越边界，则效能乘以 s_ℓ。

由 S 诱导的停机时间集合为

$$\mathcal{D}(S) \;=\; \bigcup_{m=1}^{K} [c_m - D(c_m),\, c_m), \tag{8}$$

而工作集合是其在规划时域内的补集，

$$\mathcal{W}(S) \;=\; [0, T_{\mathrm{end}}] \setminus \mathcal{D}(S). \tag{9}$$

结合区段内老化模型 $E_j(s)$、乘法型进入损失过程 $L(t;S)$，以及停机区间内的效能为零约束 $\mathcal{D}(S)$，我们得到概念上在 (10) 中定义的 MEF$f(t)$ 的运行形式。具体而言，自上次完成以来的年龄$s(t)$ $= t - t_{\mathrm{last}}$ 与当前区段索引 $j(t)$共同决定瞬时效能：

$$f(t) \;=\; \begin{cases} 0, & t \in \mathcal{D}(S), \\ L(t;S)\, E_{j(t)}\big(s(t)\big), & t \in \mathcal{W}(S). \end{cases} \tag{10}$$

公式（10）给出了公式（1）中概念性定义的 MEF 的操作形式。公式（1）将效能抽象地刻画为当前 CKM 与理想刷新后的

CKM 之间的预期性能比，而公式（10）则通过易处理的组成部分将这一比值参数化，从而能够对更新计划 S进行理论分析和数值优化。

*1)*假设。(i) $E_j(\cdot)$ 在 $[0,1]$ 中单调不增、连续且有界；(ii) $\{S_\ell\} \subset (0,1]$ 已知或经过预估（在线学习方法将在另一篇手稿中讨论）；(iii) $D(\cdot)$ $C(\cdot)$ 可测且依赖于区段；(iv) 存在可行计划。参数 $\{(\eta_j, T_{\mathrm{half},\ j})\}$ 可利用区段内的一些采样数据进行估算；$D,\ C$ 则来自运行时间和运行日志。

*2)*归一化与单位。所有时间均采用一致的单位。$E_j(0) = 1$和 $E_j(s) \downarrow \eta_j$ 均以 $s \uparrow \infty$ 表示。半衰期 $T_{\mathrm{half},\ j}$ 是超额效能 $(E_j(s) - \eta_j)$ 减半所需的时间，它提供了一种直观易懂的衰减节律。

备注：(a) 设置 $S_\ell \equiv 1$ 会移除边界冲击，这意味着只考虑 CKM 的老化。(b)

$$J(S) = \underbrace{\frac{\int_{\mathcal{W}(S)} f(t)\ dt}{\mathrm{meas}(\big(\mathcal{W}(S)\big)}}_{\text{工作时间内的平均归一化效能}} - \underbrace{\frac{\sum_{m=1}^{K} C(c_m)}{T_{\mathrm{end}}}}_{\text{整个时间段内的平均更新成本}} \tag{11}$$

第一项是 CKM 在其可用期内的平均效能，同时考虑了老化和环境变化造成的损失。第二项是按日历时间分摊的更新成本。该度量指标反映了 CKM 的长期净价值，体现了其在实际运行中效用创造与资源消耗之间的平衡，并用于确定最具成本效益的更新策略。

给定 (2)–(10) 和依赖于区段的 $D(\cdot)$ $C(\cdot)$，更新调度问题为

$$(\mathbf{P}) \quad \underset{S=\{c_m\}\subset[0,T_{\mathrm{end}}]}{\text{maximize}} \quad J(S) \ \text{ as in (11)} \quad \text{subject to} (7) \tag{12}$$

问题**(P)** 的解给出了在 $[0, T_{\mathrm{end}}]$ 上使长期效用—成本权衡最大化的完成时间集合。

III. 可预测环境下的长期更新策略

我们考虑离线预测设置，其中分段老化、进入损失参数以及依赖于区段的开销均已知。我们在 Dinkelbach 框架下提出两种算法变体：Delta-P（采用 Pareto 前沿的 Dinkelbach 方法），用于实现全局最优性；以及 Delta-L（采用泰勒线性化的 Dinkelbach 方法），以增强可扩展性。两种变体共享相同的外层分式优化结构，但内层求解器不同。

在平稳单区段状态（$M = 1$）下，问题归结为决定何时更新一次。(c) 当 $\eta_j \to 0$ 和 λ_j 较小时，对于适中的年龄，有 $E_j(s) \approx 1 - \lambda_j s$；对工作区间进行积分后，可恢复类似信息年龄的行为。(d) 可以通过禁止在指定区间内完成任务来建模硬性禁行时间窗。

C. 问题构建

我们通过长期效用—成本权衡来评估调度 S：该指标对工作期间的效能取平均，并将更新成本分摊至整个时间范围。该度量指标的设计借鉴了行业用于评估产品性能的指标。

令 meas(·) 表示勒贝格测度。所提出的目标为，

回顾式（11）中的目标函数，定义

$$F(S) = \int_{\mathcal{W}(S)} f(t)\, dt, \tag{13}$$

$$G(S) = \mathrm{meas}(\big(\mathcal{W}(S)\big)), \tag{14}$$

$$C_{\mathrm{tot}}(S) = \sum_{m=1}^{K} C(c_m), \tag{15}$$

$$H = T_{\mathrm{end}}. \tag{16}$$

问题**(P)**可改写为：

$$(\mathbf{P}) \underset{S=\{c_m\}\subset[0,T_{\mathrm{end}}]}{\text{maximize}} \quad J(S) = \frac{F(S)H - C_{\mathrm{tot}}(S)G(S)}{G(S)H} \quad \text{subject to} (7) \tag{17}$$

上述问题是一个分式规划问题，可通过 Dinkelbach算法求解[28]。

A. Delta-P：Dinkelbach赋能的长期轨迹算法—帕累托前沿

我们采用双参数 Dinkelbach 方法处理式 (11) 中的双比率目标。对于任意具有聚合统计量 $F(S)$ $G(S)$ $C_{\mathrm{tot}}(S)$，且满足 H 的可行调度 S，目标是最大化 $J(S)$。双参数 Dinkelbach 方法将 $(\lambda,\ \mu)$ 作为 $(F/G,\ C_{\mathrm{tot}}/H)$ 的估计值进行维护，并在第 k 次迭代时，在由式 (7) 导出的可行集 $\mathcal{X}$ 上求解参数化子问题：

$$\Phi_{\lambda_k,\mu_k}(S) := H\big(F(S) - \lambda_k G(S)\big) - G(S)\big(C_{\mathrm{tot}}(S) - \mu_k H\big) \longrightarrow \max_{S\in\mathcal{X}}. \tag{18}$$

引理 1 (双参数 Dinkelbach 等价性)。分式规划 ***(P)*** 可通过迭代最大化参数函数$\Phi_{\lambda,\ \mu}(S)$ 求解，该函数定义于(18)，其中(λ,μ) 按照 $\lambda \leftarrow F(S)/G(S)$ 和 $\mu \leftarrow C_{\mathrm{tot}}(S)/H$ 进行更新，直至收敛。

证明。 Dinkelbach 的经典参数化函数为

$$\begin{aligned}\Phi_\theta(F,G,C_{tot}) &= FH - C_{tot}G - \theta GH \\ &= FH - C_{tot}G - (\lambda-\mu)GH \\ &= H(F-\lambda G) - G(C_{\mathrm{tot}} - \mu H)\end{aligned} \tag{19}$$

□

尽管优化仅取决于复合参数 $\theta = \lambda - \mu$，但为了更清晰地阐释其物理意义，我们采用双参数形式。这种方法将时间边际价格 λ 与更新成本 μ的边际价格解耦，自然体现了这些不同资源之间的多目标权衡。

$\Phi_{\lambda,\ \mu}$中的乘法结构，即项 $G(S)\,C_{\mathrm{tot}}(S)$，使得单权重加法动态规划无法实现；因此，我们采用帕累托前沿机制，传播路径统计量（F，G，C_{tot}），并将标量化推迟到选择阶段。在求解子问题并得到 S_k后，参数更新如下：

$$\lambda_{k+1} = \frac{F(S_k)}{G(S_k)}, \qquad \mu_{k+1} = \frac{C_{\mathrm{tot}}(S_k)}{H}, \tag{20}$$

当 Dinkelbach 残差$|\Phi_{\lambda_k,\mu_k}(S_k)|$ 低于容差时，迭代停止。在分母为正 $G(S) > 0$、可行域非空且参数子问题可精确求解等标准前提条件下，目标值序列单调变化并收敛至$J(\cdot)$的全局最优解。

关键的算法设计选择是将外层的分式更新与内层的组合结构解耦。外层循环执行简单的双标量更新，而内层循环则在有向无环图（DAG）上求解路径问题，该图编码了可行的更新完成状态和工作区间。在每次迭代中，内层求解器在汇点构建由非支配 (F,G,C_{tot}) 元组组成的帕累托前沿，然后通过最大化得分，选出 (λ_k,μ_k) 中的最佳元组

$$\mathrm{score}(F,G,C \mid \lambda,\mu) = H\,F - G\,\big(C + (\lambda-\mu)H\big), \tag{21}$$

它在代数上等同于在（F，G，C）上计算的$\Phi_{\lambda,\ \mu}$。这支持在不重新计算整个动态规划的情况下，对任意（λ，μ）进行精确标量化，并且还

算法 1 Delta-P：Dinkelbach 增强的长期轨迹算法——帕累托前沿

输入： 片段列表 $\mathcal{S}$；停机时间 D；每次更新成本 C；规划时域 T_{end}；网格步长 Δ；容差 tol；最大值最大迭代次数 _iter

输出： 最优完成时间 $S^\star$；目标 $J^\star$

$T \leftarrow$ 构建候选时间($\mathcal{S}$，T_{end}，Δ)
($\mathcal{E}_{\mathrm{upd}}$，$\mathcal{E}_{\mathrm{term}}$) ←预计算边($T$，$\mathcal{S}$，$T_{\mathrm{end}}$，$D$)
$\mathcal{P} \leftarrow$ 帕累托前沿动态规划(T，$\mathcal{E}_{\mathrm{upd}}$，$\mathcal{E}_{\mathrm{term}}$)
$\lambda \leftarrow 0$，$\mu \leftarrow 0$，$H \leftarrow T_{\mathrm{end}}$
对于 $i = 1$ **至** 最大值 _迭代 **执行**
    (S，F，G，C) ←取最大值 $\big[H\,F - G(C + (\lambda-\mu)H)\big]$
    $\Phi \leftarrow H\,F - G(C + (\lambda-\mu)H)$
    如果 $|\Phi| <$ 容差 **则**
        $S^\star \leftarrow S$；$J^\star \leftarrow F/G - C/H$；**跳出**
    $\lambda \leftarrow F/G$；$\mu \leftarrow C/H$
返回 $S^\star$，$J^\star$

允许直接在前沿上计算 $J = F/G - C/H$，用于诊断和提前停止。

初始化 $(\lambda_0,\ \mu_0)$ 可以设为零，这样安全且简单。一种更好的热启动方法是利用最佳静态基线计算初始比率。停止规则$|\Phi_{\lambda,\ \mu}(S)|$ <容差具有稳健性；实际应用中，我们还会检查 J 的绝对变化，并通过对工作时间设置一个较小的下限来避免 G 退化。对于得分相同的调度方案，平局打破规则是优先选择较小的 C，然后选择较大的 G。

整个算法的伪代码如算法 1 所示。在每次迭代中，帕累托前沿动态规划会为有向无环图上的所有可行调度方案生成非支配三元组 $(F,\ G,\ C)$。当前迭代值 $(\lambda,\ \mu)$ 将分式目标函数线性化为 $score(F,\ G,\ C\ |\lambda,\ \mu)$；随后，算法选择使该得分最大化的前沿元组，根据所选调度更新 $\lambda \leftarrow F/G$ 和 $\mu \leftarrow C/H$，并在残差 $|\Phi|$ 低于容差时停止。前沿可以在各次迭代间重复使用，而标量化仅随 $(\lambda,\ \mu)$ 改变。

B. 参数化子问题与帕累托前沿动态规划求解器

我们将可行调度方案编码为有向无环图上的路径，其顶点集 T 收集了所有候选时间，包括分段边界 $\{\tau_\ell\}$ 以及步长为 Δ的均匀网格。时间 0 处设有源点，时间 T_{end}处设有汇点。若在 v 完成更新是可行的，则存在一条边 $(u\to v)$：如果该边表示一次更新，则必须满足不重叠约束 $v - u \geq D(v)$。我们将边分为更新边和终止边。只保留满足约束的边，便将可行性纳入图中；因此，任意一条源汇路径都对应一个停机事件互不重叠且具有明确任务集的可行调度方案

。目标是将调度问题转化为图上的轨迹优化问题，以便后续进行动态规划处理。

对于更新边 $(u\to v)$，

$$\Delta F_{u\to v}=\int_{u}^{v-D(v)} f(t)\,dt, \tag{22}$$

$$\Delta G_{u\to v}=(v-D(v))-u, \tag{23}$$

$$\Delta C_{u\to v}=C(v). \tag{24}$$

对于终端边 $(u\to T_{\text{end}})$，

$$\Delta F_{u\to T_{\text{end}}}=\int_{u}^{T_{\text{end}}} L_{u\to T_{\text{end}}}(t)\,E_{j(t)}(t-u)\,dt, \tag{25}$$

$$\Delta G_{u\to T_{\text{end}}}=T_{\text{end}}-u, \tag{26}$$

$$\Delta C_{u\to T_{\text{end}}}=0. \tag{27}$$

因此，对于任意路径 S，

$$(F,G,C_{\text{tot}})=\sum_{e\in S}(\Delta F_e,\Delta G_e,\Delta C_e), \tag{28}$$

也就是说，这三个准则在各条边上仍然是可加的。在停机区间内，根据定义，$f(t)=0$，既不计入 ΔF，也不计入 ΔG。根据当前路径语义，这些规则使 $(\Delta F,\ \Delta G,\ \Delta C)$ 具有马尔可夫性。

因此，内层问题通过多准则动态规划求解一次：在每个顶点 v 处，维护一个集合 $\mathcal{P}[v]$，其中包含偏序下的非支配元组 $(F,\ G,\ C)$。

动态规划按拓扑序进行。将 $\mathcal{P}[\text{src}]=\{(0,0,0)\}$ 初始化，并为其他顶点设置 $\mathcal{P}[v]=\emptyset$。对于每条边 $(u\to v)$，其增量为 $(\Delta F,\ \Delta G,\ \Delta C)$；对于每个元组 $(F_u,\ G_u,\ C_u)\in\mathcal{P}[u]$，构造 $(F,\ G,\ C)=(F_u+\Delta F,\ G_u+\Delta G,\ C_u+\Delta C)$。将 $(F,\ G,\ C)$ 插入 $\mathcal{P}[v]$，前提是它未被支配；并移除它所支配的任何元组。保留回溯指针以便恢复调度。在汇点处，将 $\mathcal{P}[\text{sink}]$ 传递给外层循环，并通过最大化 $(F-\lambda G)-(C-\mu H)$ 选出最佳元组，或者等价地，通过最大化带有 $G>0$ 的 $J=F/G-C/H$ 来选出。为控制前沿集合的大小，我们可选择应用 ε-支配：若满足 $F_1\ge(1-\varepsilon)F_2$、$G_1\le(1+\varepsilon)G_2$ 且 $C_1\le(1+\varepsilon)C_2$，则 $(F_1,\ G_1,\ C_1)$ ε-支配 $(F_2,\ G_2,\ C_2)$。这种剪枝带来了可调节的复杂度–精度权衡，并使内存使用量保持可预测。

C. 最优性

外层 Dinkelbach 最优性的前提是内层算法已达到最优。

1) 内层动态规划的最优性：

a) 充分条件：

•(C1) 可加性与非负性：$(\Delta G_e,\ \Delta C_e,\ \Delta F_e)$具有可加性且为非负（参见 (22)–(25)）。

• (C2)可行性编码与有向无环图：所有约束均通过可行边编码；该图无环，因此复杂度可控。

算法 2 帕累托前沿动态规划：用于以下对象的多准则动态规划 $(F,\ G,\ C_{\text{tot}})$

输入： 有向无环图 $(T,\mathcal{E})$; 边增量 $(\Delta F,\Delta G,\Delta C)$
输出： 前沿集合 $\{\mathcal{P}[v]\}_{v\in T}$ 及其回溯指针

$\mathcal{P}[\text{src}]\leftarrow\{(0,\ 0,\ 0)\}$; $\mathcal{P}[v]\leftarrow\emptyset$ 对于 $v\neq$src
对于 按拓扑序排列的顶点 u**执行**
  对于 每个 $(u\to v)\in\mathcal{E}$（其值为 $(\Delta F,\Delta G,\Delta C)$）**执行**
    对于每个 $(F_u,G_u,C_u)\in\mathcal{P}[u]$ **执行**
      $(\hat{F},\hat{G},\hat{C})\leftarrow(F_u+\Delta F,\ G_u+\Delta G,\ C_u+\Delta C)$
      如果 NOTDOMINATED($(\hat{F},\hat{G},\hat{C}),\ \mathcal{P}[v]$) **then**
        插入 $(\hat{F},\hat{G},\hat{C})$，并附带回溯指针；在 $\mathcal{P}[v]$中剪除被支配的元组
  $\mathcal{P}[v]\leftarrow$ ε剪枝$\big(\mathcal{P}[v],\ \ \varepsilon\big)$
返回 $\{\mathcal{P}[v]\}$

•(C3) 单调标量化：最终评分$\Phi_{\lambda,\ \mu}$ 关于 F 单调不减，关于 G单调不增，C，其中 $G>0$。

b) 路线图： 步骤 1：在 (C1)–(C2) 条件下，动态规划计算汇点处的完整帕累托前沿。步骤 2：在 (C3) 条件下，任何单调标量化 J 和$\Phi_{\lambda,\ \mu}$都能在该前沿上达到其全局最优解。步骤 3：利用 Dinkelbach恒等式，当内层前沿精确时，外层循环收敛至全局最优解。

c) 设定： 在由离散候选时间构成的有向无环图上，每条可行路径 S 都具有一个可加且非负的资源向量

$$\mathbf{u}(S)=(G(S),\,C(S),\,F(S))=\sum_{e\in S}(\Delta G_e,\Delta C_e,\Delta F_e),$$

$$(\Delta G_e,\Delta C_e,\Delta F_e)\in\mathbb{R}^3_+ \tag{29}$$

定义1 （支配、非支配、帕累托前沿）。 定义偏序

$$\mathbf{u}_1\preceq\mathbf{u}_2\iff\big(G_1\le G_2,\ C_1\le C_2,\ F_1\ge F_2\big), \tag{30}$$

至少有一个严格不等式成立时，我们称$\mathbf{u}_1$ 支配$\mathbf{u}_2$。

令终端帕累托前沿为

$$\mathcal{L}_{sink}=\text{ParetoMin}\big\{\mathbf{u}(S):S\ \textit{is a path to sink}\big\} \tag{31}$$

动态规划在每个节点 i 维护一个帕累托标签集 $\mathcal{L}_i$。对于任意边 e：$i\to k$，将 $(g,c,f)\in\mathcal{L}_i$通过 $(g+\Delta G_e,\ c+\Delta C_e,\ f+\Delta F_e)$ 扩展到 $\mathcal{L}_k$，然后剪除被支配的标签。按拓扑序遍历一次。

d) 目标与标量化：对于单条路径向量 (g,c,f)，定义

$$\Psi(g,c,f)\ :=\ \frac{f}{g}-\frac{c}{H},\quad(g>0,\ H>0). \tag{32}$$

$$\Phi_{\lambda,\mu}(g,c,f)=Hf-g\big(c+(\lambda-\mu)H\big). \tag{33}$$

引理2 (扩展下的支配关系保持)。 若在节点 i, $\ell_1 = (g_1,\ c_1,\ f_1) \preceq \ell_2 = (g_2,\ c_2,\ f_2)$，则对于任意边 $e: i \to k$,

$$\ell_1 + \Delta_e \ \preceq\ \ell_2 + \Delta_e, \quad \Delta_e = (\Delta G_e, \Delta C_e, \Delta F_e). \tag{34}$$

证明。 根据（C1），由可加性和分量非负性可得 $g_1 + \Delta G_e \le g_2 + \Delta G_e$, $c_1 + \Delta C_e \le c_2 + \Delta C_e$，以及 $f_1 + \Delta F_e \ge f_2 + \Delta F_e$，其中至少一个为严格不等式。参见多目标最短路径标签方法的标准性质 [29]–[31]。 □

引理3 （支配剪枝的安全性）。 在同一节点 i，若 $\ell_1 \preceq \ell_2$，则删除 ℓ_2 不会丢弃任何可能在 T 处成为帕累托最优或在任何（*C3*）类型标量化下成为最优的路径。

证明。 对于从 i 到 $sink$ 的任意后缀路径 P，逐边应用引理2: $\ell_1 + \Delta(P) \preceq \ell_2 + \Delta(P)$。（C3）中的单调性给出 $\Psi(\ell_1 + \Delta(P)) \ge \Psi(\ell_2 + \Delta(P))$ 且 $\Phi_{\lambda,\mu}(\ell_1 + \Delta(P)) \ge \Phi_{\lambda,\mu}(\ell_2 + \Delta(P))$。因此，可以安全地删除被支配的标签。 □

引理4 （完备性不变量）。 按拓扑序， $\mathcal{L}_i$ 等于所有到达 i的可行标签所构成的帕累托前沿。

证明。 在源点处， $\mathcal{L}_{\mathrm{src}} = \{(0,\ 0,\ 0)\}$。假设对于 i的所有前驱节点，该结论均成立。沿可行边扩展它们的前沿，即可覆盖所有到达 i的标签；根据引理3进行剪枝后，即得到i处的帕累托前沿。 □

定理1 （终端前沿的完备性）。在满足（*C1*）–（*C2*）且有向无环图有限的条件下，算法终止时，$\mathcal{L}_T$ 恰好是所有到达 T的可行路径所构成的帕累托前沿。

证明。 在 $i = sink$处应用引理4。有限性和无环性确保算法在一次拓扑遍历中终止。 □

定理2 （在前沿上达到全局最优性）。 在满足（*C3*）且 $G > 0$,

$$\max_S J(S) = \max_{\mathbf{u}\in\mathcal{L}_T} \Psi(\mathbf{u}), \ \max_S \Phi_{\lambda,\mu}(S) = \max_{\mathbf{u}\in\mathcal{L}_T} \Phi_{\lambda,\mu}(\mathbf{u}) \tag{35}$$

证明。 单调性：若$\mathbf{u}_1 \preceq \mathbf{u}_2$ 且 $g_1,\ g_2 > 0$，则

$$\frac{f_1}{g_1} - \frac{f_2}{g_2} = \frac{f_2(g_2 - g_1) + (f_1 - f_2)g_2}{g_1 g_2} \ \ge\ 0, \tag{36}$$

$$-\frac{c_1}{H} \ \ge\ -\frac{c_2}{H}, \tag{37}$$

因此 $J(\mathbf{u}_1) \ge J(\mathbf{u}_2)$，对于 $\Phi_{\lambda,\ \mu}$ 也同样如此。任何非帕累托标签都被一个得分不差的帕累托标签支配；因此，最大值在$\mathcal{L}_T$上取得。 □

*2）Dinkelbach*算法的最优性： $H > 0$和 $G(S) > 0$ 对所有可行路径 S均成立。根据 $J(S)$和 $\Phi_{\lambda,\ \mu}(S)$的定义，其中 $\theta := \lambda - \mu$,

$$\Phi_{\lambda,\mu}(S) = G(S)H\big(J(S) - \theta\big). \tag{38}$$

引理5 （符号等价性）。 对于任意（λ、 μ） $\in \mathbb{R}^2$以及任意可行的 S、 $\Phi_{\lambda,\ \mu}(S) \ge 0 \iff J(S) \ge \lambda - \mu$。

Proof. Immediate from (38) and $G(S)H > 0$. □

引理6 （残差非负性与精确停止）。给定（λ_k， μ_k），令

$$S_k \in \arg\max_S \Phi_{\lambda_k,\mu_k}(S), \qquad r_k := \Phi_{\lambda_k,\mu_k}(S_k), \tag{39}$$

并更新 $\lambda_{k+1} = F(S_k)/G(S_k)$、 $\mu_{k+1} = C(S_k)/H$（等价地， $\theta_{k+1} = J(S_k)$）。则 $r_k \ge 0$，并且 $r_k = 0$ 当且仅当 S_k 是 J的全局最大化解。

证明。 根据第 $k-1$次迭代的更新， $\theta_k = \lambda_k - \mu_k = J(S_{k-1})$。因此由（38）可得，$\Phi_{\lambda_k,\ \mu_k}(S_{k-1}) = 0$，所以 $r_k = \max_S \Phi_{\lambda_k,\ \mu_k}(S) \ge 0$。此外， $r_k = 0$ 当且仅当$\Phi_{\lambda_k,\ \mu_k}(S) \le 0$ 对所有 S 成立，且等号在 S_k处成立。根据引理5，这等价于 $J(S) \le \theta_k$ 对所有 S成立，且 $J(S_k) = \theta_k$，即 S_k 取得 J的全局最大值。 □

引理7 （单调上升）。 序列 $\{J(S_k)\}$单调不减，且在$r_k > 0$时严格递增。

证明。 由（38）， $r_k = G(S_k)H\big(J(S_k) - \theta_k\big) \ge 0$，因此$J(S_k) \ge \theta_k = J(S_{k-1})$。若 $r_k > 0$，则 $J(S_k) > \theta_k$，故严格上升。 □

引理8 （离散性下的有限终止）。若可行集有限，则迭代在有限步内终止，并满足 $r_k = 0$ 且 $J(S_k)$ =最大$_S$ $J(S)$。

证明。由引理7， $\{J(S_k)\}$ 是一个取值于有限集的单调不减序列，因此会在有限步内达到稳定；由引理6，稳定性意味着 $r_k = 0$ 以及全局最优性。 □

定理3 （双参数 Dinkelbach 方案的全局最优性）。令 $J^\star =$ 最大$_S$ $J(S)$。在每次迭代中进行精确的内层最大化时，序列$\{J(S_k)\}$ 单调不减，并收敛至 $J^\star$，其中$r_k \to 0$。此外，若 J 在可行集上仅取有限个值，则算法将在有限步内终止，并得到 J的一个全局最大化解。

证明。 根据引理7， $\{J(S_k)\}$ 单调不减，且有上界 $J^\star$，因此收敛到某个 $J \le J^\star$。若 $J < J^\star$，则精确内层最大化结合式（38）会使严格为正的残差无限次出现，这与收敛性矛盾；因此 $J = J^\star$ 且$r_k \to 0$。有限终止的情形由引理8得出。 □

表 I 复杂度与最优性：Delta-P 与 Delta-L

方法	内部时间	总时间	空间	外部复用	最优性
Delta-P	$O(\lvert\mathcal{E}\rvert\,\Gamma)$	$O(N_{\text{outer}}+\lvert\mathcal{E}\rvert\,\Gamma)$	$O(\lvert T\rvert\,\Gamma)$	复用前沿；选择于 $O(\Gamma)$	全局
Delta-L	$O(\lvert\mathcal{E}\rvert)$	$O(N_{\text{outer}}\,N_{\text{TL}}\,\lvert\mathcal{E}\rvert)$	$O(\lvert T\rvert)$	重新加权每个 TL 步骤	近似

D. 收敛性

Dinkelbach 算法的超线性收敛源于这样一个事实：它等价于对一个由分式目标函数导出的凸函数应用牛顿法 [23]。

定理 4 (内层动态规划收敛性)。在边权重非负的有限有向无环图$(\Delta F,\Delta G,\Delta C)$上，算法 *2* 在一次拓扑遍历中即可计算出精确的帕累托前沿 $\mathcal{P}$[汇点]，并在有限步内终止。

证明。任一路径至多包含 $|V|-1$ 条边；每个 $(F,\ G,\ C)$ 都是非负增量的有限和，因此标签空间是有限的。□

E. 复杂度

Delta-P 的时间复杂度与边数和平均前沿大小的乘积成正比。在包含 $|T|$ 个顶点和 $|\mathcal{E}|$ 条边的网格上，最坏情况下的开销为 $O(|\mathcal{E}|\,\Gamma)$，其中 Γ 是剪枝后每个顶点前沿基数的上界；在实践中，在现实参数和粗到中等分辨率的网格下，Γ 保持适中。该方法的开销随 Dinkelbach 迭代次数线性增长，而迭代次数通常较少。

F. Delta-L：*Dinkelbach* 启用的长轨迹算法线性化

当 Γ 增大时（网格更精细或约束更多），Delta-P 就会成为瓶颈。为降低复杂度，我们提出 Delta-L，它对式 18 中的乘法项采用泰勒展开，该项包含路径层级的非加性乘积 $G\,C_{\text{tot}}$。

在内迭代 k 中，给定当前的调度统计量 $(F^{(k)},\ G^{(k)},\ C^{(k)})$，使用一阶泰勒展开，在当前迭代点 $(G^{(k)},\ C^{(k)})$ 处将乘积 $f(G,\ C)\ =\ G\,C$ 线性化：

$$G\,C\ \approx\ f(G^{(k)},C^{(k)})+\frac{\partial f}{\partial G}\Big|_{(k)}(G-G^{(k)})+\frac{\partial f}{\partial C}\Big|_{(k)}(C-C^{(k)}) \tag{40}$$
$$=C^{(k)}G+G^{(k)}C-G^{(k)}C^{(k)}.$$

去掉常数项后，路径得分变为边可加形式，每边权重为

$$w_k(e)\ =\ H\,\Delta F_e-\big(C^{(k)}+(\lambda-\mu)H\big)\,\Delta G_e-G^{(k)}\,\Delta C_e. \tag{41}$$

因此，Delta-L 中的一步泰勒线性化（TL）求解的是有向无环图上的单权重最长路径动态规划。总体时间复杂度如表 II 所示。

示例 令

$T=\{t_0=0<t_1=2<t_2=5<t_3=9<t_4=12=T_{\text{end}}\}$，(42)

并包含一条边 $u\ \to\ v$ 当且仅当 $v\ >\ u$ 和$v-u\ \geq\ D(v)\ =\ 2$。按照（22）–（25）预先计算边增量（$\Delta F,\ \Delta G,\ \Delta C$）。然后，可行边包括

$$\mathcal{E}(t_0)=\{t_0\to t_1,\ t_0\to t_2,\ t_0\to t_3,\ t_0\to t_4\}, \tag{43}$$
$$\mathcal{E}(t_1)=\{t_1\to t_2,\ t_1\to t_3,\ t_1\to t_4\}, \tag{44}$$
$$\mathcal{E}(t_2)=\{t_2\to t_3,\ t_2\to t_4\}, \tag{45}$$
$$\mathcal{E}(t_3)=\{t_3\to t_4\}, \tag{46}$$

并且每个 $u\in T$ 也可能有一条直接的终端边$u\to T_{\text{end}}$。

动态规划按拓扑序遍历顶点

$$t_0\ \to\ t_1\ \to\ t_2\ \to\ t_3\ \to\ t_4(=T_{\text{end}}), \tag{47}$$

在源点 $P[t_0]=\{$**(0，0，0)**$\}$处初始化前沿。对于此顺序中的每个顶点 u 以及每条出边（$u\to v$），其增量为（$\Delta F,\ \Delta G,\ \Delta C$），传播候选项

$$(\hat{F},\hat{G},\hat{C})=(F+\Delta F,\ G+\Delta G,\ C+\Delta C) \tag{48}$$

从 $P[u]$ 到 $P[v]$，仅插入非支配元组。处理完 t_4 后，从 $P[t_4]$ 中选出使当前标量化值最大的元组，并通过回溯指针恢复相应路径。

一次 TL 迭代按如下步骤进行：

1) 给定 $(G^{(k)},\ C^{(k)})$，为每条边 e 赋予权重 $w_k\ e\ =\ H\,\Delta F_e-\big(C^{(k)}+\ \lambda-\mu\ H\big)\Delta G_e-(\,)(\,)$。$G^{(k)}\Delta C_e$

2) 按拓扑序 $t_0\to t_1\to t_2\to t_3\to t_4$运行单权重最长路径动态规划：

$$\mathrm{dp}[t_0]=0,\qquad \mathrm{dp}[v]=\max_{u\to v}\ \big(\mathrm{dp}[u]+w_k(u\to v)\big), \tag{49}$$

保留回溯指针以恢复路径 $S^{(k+1)}$。3) 对 $S^{(k+1)}$ 上的边增量求和，以更新 $(F^{(k+1)},\ G^{(k+1)},\ C^{(k+1)})$。

4) 若 $|\Phi_{\lambda,\ }\Phi_{\lambda\ \mu}\ S^{(k)}\ |\ <\ \ _{\text{in}}\ \ S^{(k+1)}\ =\ S^{(k)}$ $(\ S^{(k+1)})\ .\ (\,)$ tol 或，则停止内层循环；否则设置 $k\leftarrow k+1$ 并重复。

在外层双参数 Dinkelbach 循环中 ，更新 $\lambda\leftarrow F/G$和 $\mu\leftarrow C/H$，并在$|\Phi_{\lambda,\ \mu}(S)|$ <时停止 tol. C比较 d with the Delta-P, the Delta-L 代理通过少数几个 $O(|\mathcal{E}|)$最长路径求解，替代前沿维护，在{v1}较大的情况下，以精确性换取速度 其中 Γ 很大。

IV. 不可预测环境下的短期更新策略

在无法预测的情境中，未来的环境变化尚不可知，因此更新策略具有短期性。更新决策仅基于当前观测到的效能衰减和已知开销。令 $D > 0$ 为更新停机时间，$C \geq 0$ 为其成本。问题归结为寻找单次更新的最佳时间 t，以最大化净效用，公式如下：

$$\max_t g(t) = \frac{\int_0^t f(x)dx}{t} - \frac{C}{t+D} \tag{50}$$

对 $g(t)$ 进行优化，可以根据当前观测值确定最优更新时间点。由于我们不知道每种环境会持续多久，因此每次优化只决定进行一次更新。

根据 $f(t)$ 的性质，在不同情况下结果有所不同，但总体而言，可以得出最优更新时间具有阈值结构：当 $\frac{C}{D^2} \leq -f'(0)_2$ 时，最优更新时间为 $t = 0$；否则，最优更新时间是首个满足 $g'(t^*) = \frac{tf(t) - \int_0^t f(x)dx}{t^2} + \frac{C}{(t+D)^2} = 0$ 的 t^*。

$g(t)$ 的导数由两部分相加而成，其中一部分显然为正或零，另一部分显然为负或零。如果这两项之和为零，则 t 就是最优更新时间。如果 $f(t)$ 为常数，$g(t)$ 会随时间递增，这意味着始终没有必要更新。如果 $C = 0$，也就是说更新免费，那么 $g(t)$ 会随时间递减，最优更新时间为 $t = 0$，这意味着自环境变化后即可进行更新。在其他常见情况下，$g'(t)$ 由一部分正值和一部分负值相加而成。

令 $h(t) = tf(t) - \int_0^t f(x)dx$，由上文可知，$h(t) < 0$ 以及 $h'(t) < 0$。令 $p(t) = \frac{C}{(t+D)^2}$，显然，$p(t)$ 是一个递减函数，从 $\frac{C}{D^2}$ 开始下降，并渐近趋于零。要讨论 $g'(t)$ 的正负形态，我们只需考察 $h(t)$ 和 $p(t)$ 的变化率。

首先，研究 $h(t)$ 在 t=0 时的取值。当 t 接近 0 时，$h(t)$ 和 t^2 也接近 0，因此应用洛必达法则：

$$\begin{aligned}\lim_{t\to 0} H(t) = \lim_{t\to 0} \frac{h(t)}{t^2} &= \lim_{t\to 0} \frac{tf(t) - \int_0^t f(x)dx}{t^2} \\ &= \lim_{t\to 0} \frac{tf'(t)}{2t} \\ &= \lim_{t\to 0} \frac{f'(t)}{2}\end{aligned} \tag{51}$$

正值部分始终是一个从正值递减至接近零的函数；负值部分则可能是一个从负值或零递减至 $-\infty$ 或某个负值的函数，也可能是一个从负值或 $-\infty$ 递增至某个负值或零的函数。

基于上述分析，我们可以得出最终结论，如图2所示。

我们可以从实际角度解读最优更新策略的阈值结构。比值 $\frac{C}{D^2}$ 可以理解为更新过程中计算资源消耗的变化率，它反映了相对于更新延迟，系统计算能力被消耗的速度。该比值充当一个关键阈值，用于决定立即更新是否为最优选择。

当环境快速变化时（其特征是 $f'(0)_2$ 的值较大且为负），信息价值会随着时间迅速下降，因此，尽管需要付出计算成本，维持信息的新鲜度仍更有价值。在这种情况下，最优策略是立即更新（$t^* = 0$），因为维持信息新鲜度所带来的收益超过了计算开销。这很容易理解：在快速变化的环境中，使用过时信息进行决策的代价，远高于频繁更新所需的计算成本。反之，当环境变化缓慢，即 $-f'(0)_2$ 的值较小时，信息在更长时间内仍有价值，系统可以延长两次更新之间的等待时间，以尽量减少计算资源消耗。因此，阈值结构提供了一种方法，可根据环境变化与资源消耗的相对速率，在信息新鲜度和计算效率之间进行权衡。

V. 仿真

A. 仿真设置

我们考虑时长为 $T_{\text{end}} = 300$ 分钟的规划时域，每个案例包含 $M = 6$ 个区段。我们采用分层网格：在整个范围内使用间隔为 $\Delta_{\text{coarse}} = 5$ 分钟的粗网格，并在每个边界$\{0, \tau_j, T_{\text{end}}\}$两侧 ± 12 分钟范围内增设间隔为 $\Delta_{\text{fine}} = 0.25$ 分钟的细网格，同时始终纳入边界本身。完成初步求解后，我们执行一次局部细化与重求解步骤：在求得的完成时间附近 ± 6 分钟范围内，以 0.2 分钟为步长加入额外候选项，然后重新运行一次求解器。这是一种成本低廉的一次性自适应细化方法，可使连续最优解与离散网格对齐。

为进一步控制复杂度而不牺牲最优性，我们通过施加 $t_k - t_i \leq 90$ 分钟的限制来剪除过长的边。所有边增量（$\Delta F, \Delta G, \Delta C$）均通过闭式边界感知积分精确计算一次，并在内外迭代中重复使用，从而避免重复进行数值积分。

为验证我们的算法在不同环境中的特性，我们针对表III中总结的三种环境类型分别绘制了 N 个独立同分布的案例；

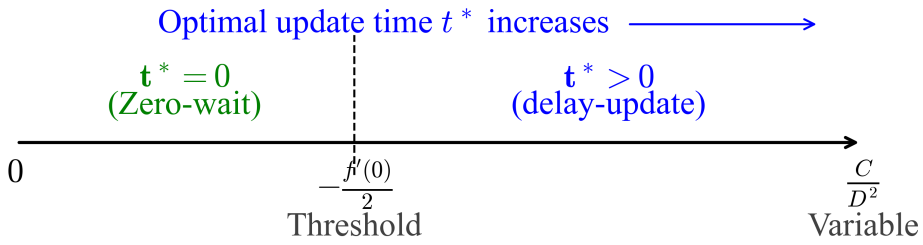

图 2. 基于阈值的更新策略示意图。横轴表示**计算资源消耗**的加速度（$\frac{C}{D^2}$），而竖直虚线表示由**环境变化率**（$-\frac{f'(0)}{2}$）确定的阈值。仅当计算资源消耗加速度低于环境退化速率时，零等待策略才是最优的。

表III 仿真中使用的环境类型和采样区间

Type	S_{entry}	$T_{1/2}$（分钟）	η	$(D,\ C)$（典型）
A：小入口、快速	$[0.2, 0.5]$	$[2.5, 5]$	$[0, 0.08]$	$D \in \{0.8, 1.0, 1.2, 1.5\},\ C \in [0.05, 0.2]$
B：大入口、缓慢	$[0.9, 1.0]$	$[50, 80]$	$[0.2, 0.35]$	$D \in [2.0, 3.5],\ C \in [0.8, 1.8]$
C：混合非对角元素	$\{[0.6, 0.8]$ 或 $[0.9, 1.0]\}$	$\{[50, 80]$ 或 $[8, 18]\}$	$[0.05, 0.30]$	$D \in [2.0, 3.5],\ C \in [0.8, 1.8]$

在所有类型中，每次更新的停机时间 D 和成本 C 都依赖于区段。

对于数值求解器，我们将外层 Dinkelbach 容差设为 10^{-6}，最多迭代 60 次；内层泰勒线性化（TL）求解器的容差设为 10^{-6}，每个外层步骤最多迭代 25 次。网格以 $\Delta_{\text{coarse}} = 3$ 分钟为间隔初始化，并在分段边界及已获得的完成时间附近进行局部细化，间隔为 $\Delta_{\text{fine}} = 0.2$ 分钟，确保将所有分段边界 $\{\tau_j\}$ 纳入候选时间。为减少边数，我们剪除跨越 90 分钟以上的边。所有边增量（$\Delta F,\ \Delta G,\ \Delta C$）均通过对式 (22)–(25) 进行闭式积分预先计算，并在迭代过程中重复使用。

B. 仿真结果

我们首先将近似最优策略与另外三种策略进行比较：**(1)零等待更新**：环境进入新的分段后立即更新。**(2)固定 10 分钟**：从初始时刻起每 10 分钟更新一次。**(3)固定 25 分钟**：从初始时刻起每 25 分钟更新一次。

为诊断局部单周期激励，我们针对每个区段 j 计算初始衰减斜率及其短期阈值

$$T_j = -\frac{f_j'(0)}{2} = \frac{(1-\eta_j)\lambda_j}{2}, \tag{52}$$

并在轨迹图中与 $(D_j,\ C_j)$ 一同显示。短期充分规则“若 $R_j = C_j/D_j^2 \le T_j$ 则等待时间为零”作为参考线。

图 3 展示了各种 CKM 更新策略相对于帕累托前沿的表现，该前沿

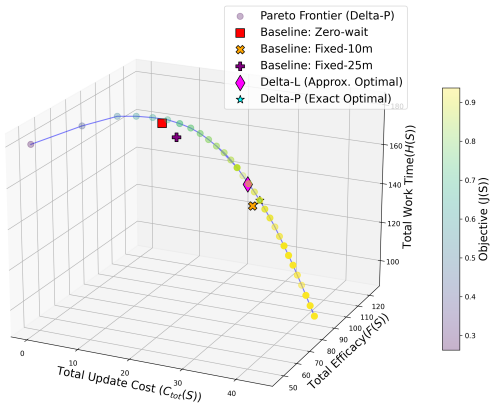

图 3. 工作时间 G、效能 F 与更新成本 C_{tot} 的帕累托前沿。

代表理论性能上限，界定了总更新成本、效能得分与工作时间之间的最优权衡。结果验证了我们提出的 Delta-P（青色星形标记）能够精准达到这一最优前沿，证实了其理论最优性。尤为重要的是，低复杂度的 Delta-L 算法（洋红色菱形标记）所得解与前沿极为接近，展现出近乎最优的性能。相比之下，所有基准策略均表现为次优，其性能点明显落在该边界之内。

基准策略暴露了常见的陷阱：零等待策略过于保守，最大限度延长了工作时间，却取得了最差的效能，与 Delta-P 相比差距达 33.9%。

表 IV Delta-P/L 与基准策略的性能

策略	效能	工作时间	更新成本	目标	计算时间
Delta-P	151	167	31.5	**0.7674**	63.62s
Delta-L	153	173	28.5	**0.7628**	0.45s
零等待	130	215	7.5	0.5731	0.00秒
固定 10 分钟	143	164	33	0.7308	0.00秒
固定 25 分钟	145	203	13.5	0.6563	0.00秒

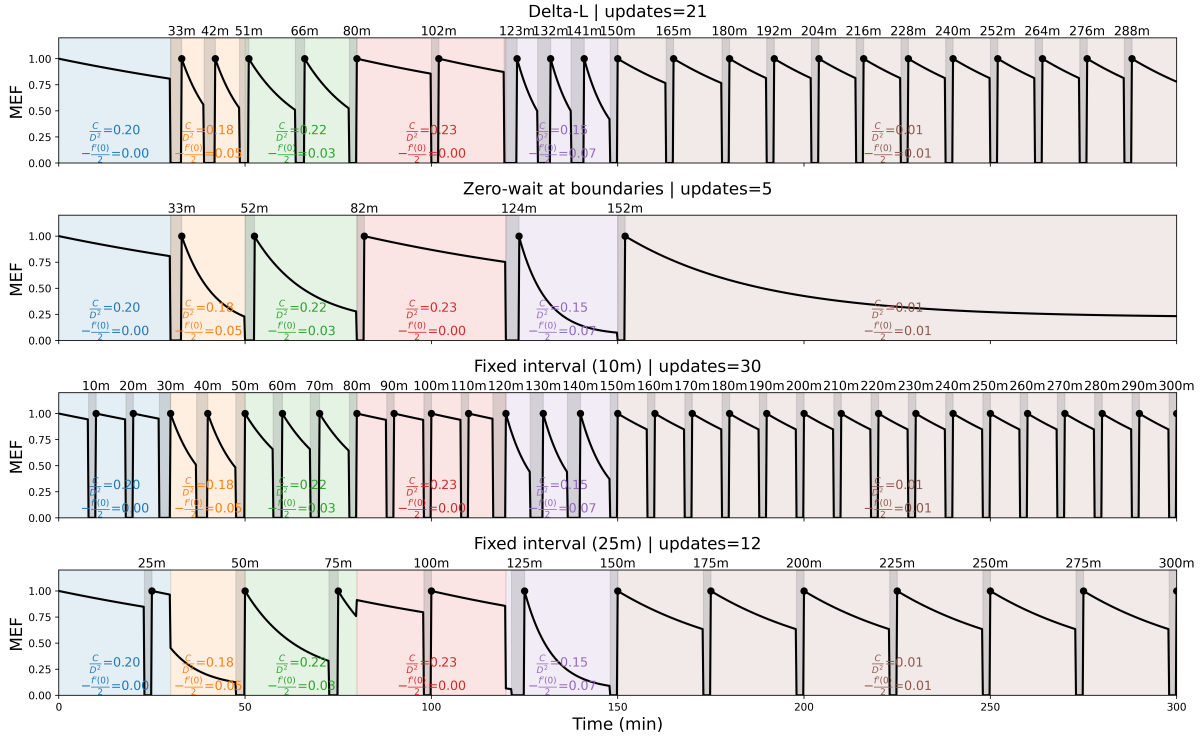

图4. Delta-L 与基线更新策略的比较。Delta-L 将更新时机与分段边界和快速衰减时段进行最优对齐，取得了高于零等待、固定 10 分钟和固定 25 分钟基线的目标值 J 。

相反，固定 10 分钟策略过于激进，成本最高，平均疗效却平平，这表明没有理论依据地频繁更新并不可行。这些结果证实，考虑环境且注重成本的调度策略显著优于基准策略。

图4展示了Delta-L调度与三个基线方案相比的具体更新措施。最优轨迹将若干次完成时间安排在 T_j 较大、 S_{entry} 较小的边界处；当停机时间 D_j 较长，或稍作调整即可更好地与下一区段对齐时，则会引入短暂延迟。零等待基线仅在边界处采取行动，因此错过了有利可图的区段内刷新；固定间隔基线则忽略了异质性，常在收益较低的区间进行刷新，导致停机时间增加，却未能带来相应的效能提升。在不同实例中，Delta-L的 J 均高于所有基线，同时将更新次数维持在适中水平；更新集中在快速衰减分段以及较大初始损失之前。

为进一步理解这些结果背后的决策机制，我们在图4中将 $\frac{C}{D^2}$ 与 $-\frac{f'(0)}{2}$进行比较，发现短期规则是在年龄为0时采用的局部单区间判据。在某些环境分段中，长期策略的行动并不符合短期策略的理论。短期规则孤立地考察一个区间，并判断在该区间内多进行一次更新是否能带来即时收益。它将当前状态视为全新状态，并忽略此次更新对后续区间起始状态和预算的影响。长期目标具有全局性：它会最大化整个规划时域内的单位成本平均收益，并对所有更新采用相同的收益—成本衡量标准。由于每次更新都会重置状态并改变未来区间的时间安排，最优方案会将更新移至成本较低或边际收益较高的时段，并可能延迟或跳过那些局部看来颇具吸引力的更新。

尽管长期最优策略可能偏离短期规则，但短期直觉仍能很好地指明选择方向。以此作为

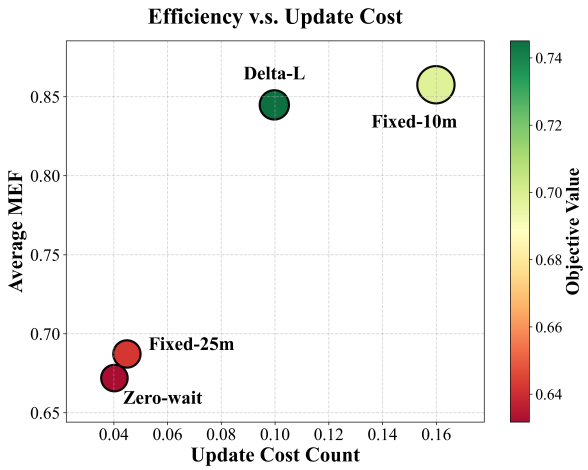

图5. 平均 MEF 与更新成本之间的权衡，其中每个点的颜色表示其目标值。

参考，我们将环境分为以下几类，并基于大量仿真，报告每类环境中最常被选择的更新时间。

类型A（小入口、快速，例如在火车站台上）。在初始损失较大且区段内衰减较快的情况下，零等待出现的概率并不低，但在我们的默认抽样中，延迟更新仍占主导。这是因为全局最优解有时倾向于短暂延迟，以便与后续区段对齐。

类型B（大入口、缓慢，例如在街上行走）。延迟更新占据压倒性优势；零等待很少见，不更新则零星出现。这符合直觉：-较小的初始损失（$S_{\text{entry}} \approx 1$）和较慢的衰减会降低立即刷新的边际收益，而停机时间和成本依然存在。

类型C（其他）。呈现出混合模式：当抽样落入“小入口 + 中等/快速”区间时，零等待更常见；否则，延迟更新占主导。不更新主要出现在“大入口 +缓慢”的角落区域。

为评估性能与成本之间的详细权衡，图5展示了平均更新成本、平均 MEF 和更新次数。尽管激进的固定 10 分钟策略取得了最高的 MEF（0.86），但代价是更新频率最高。相比之下，Delta-L 实现了更优的平衡：以减少 37.5% 的更新成本和 33.3% 的更新次数为代价，达到最高 MEF 的 98.6%。这凸显了我们算法的主要优势：通过智能分配资源，在显著降低运营成本和系统停机时间的同时，实现近乎最优的 CKM 质量，证明了其在实际网络部署中的效能。

六、结论

本文探讨了一个关键问题：在动态环境中，何时更新信道知识图谱。通过引入地图效能函数并将更新调度建模为分式规划，我们提出了追求全局最优性的 Delta-P 和具有近线性可扩展性的 Delta-L。研究结果表明，最优更新决策取决于信息衰减与资源消耗之间的竞争。对于不可预测环境，我们推导出一个闭式阈值：当 $-f'(0)/2 > C/D^2$ 时，立即更新是最优选择。这体现了一种直观的权衡——当信息价值的衰减速度快于资源消耗速度时，保持信息新鲜度的收益超过计算节省；否则，审慎延迟更新更为可取。对于可预测环境，较大的初始损失和较快的信息衰减有利于立即更新，以避免性能迅速下降；而较小的初始损失和较慢的信息衰减则允许延迟更新，从而节约资源。值得注意的是，长期策略会合理地偏离短期规则，将更新安排在分段边界，并在整个规划时域内重新分配预算，以优先保障全局性能，而非局部收益。

未来的工作包括多小区协同更新、环境动态的在线学习，以及与资源分配的集成。

参考文献

[1] D. Wen、Y. Zhou、X. Li、Y. Shi、K. Huang 和 K. B. Letaief，《感知、通信与计算一体化综述》，*IEEE Communications Surveys & Tutorials*，第 1–1 页，2024 年。[2] A. Fayad、T. Cinkler 和 J. Rak，《迈向 6G 光前传：赋能技术与研究展望综述》，*IEEE Communications Surveys & Tutorials*，第 27 卷，第 1 期，第 629–666 页，2025 年。[3] P. Yang、Y. Xiao、M. Xiao 和 S. Li，《6G 无线通信：愿景与潜在技术》，*IEEE Network*，第 33 卷，第 4 期，第 70–75 页，2019 年。[4] Z. Z hao、Q. Du、D. Wang、X. Tang 和 H. Song，《面向 6G 网络的服务感知无线接入前景概述》，*Electronics*，第 11 卷，第 8 期，第 1262 页，2022 年。[5] L. ´Arvai 和 S. Homolya，《面向室内定位的高效无线电地图更新算法》，载于 *2020 21th International Carpathian Control Conference (ICCC)*，2020 年，第 1–5 页。[6] Z. Li、X. Zhu 和 J. Cao，《增强位置感知：信息年龄与定位精度的视角》，*IEEE Journal on Selected Areas in Communications*，第 42 卷，第 10 期，第 2704–2718 页，2024 年。[7] C. Zhang、T. Wang 和 C. He，《通过信道知识图补偿无人机通信中的定位误差》，*IEEE Internet of Things Journal*，第 12 卷，第 11 期，第 16 977–16 989 页，2025 年。[8] Y. Gong、X. Wang、T. Wang 和 C. Zhang，《结合机器遗忘的信道知识图更新》，载于 *2025*年 *IEEE/CIC*中国通信国际会议 *(ICCC Workshops)*，2025 年，第 1–6 页。[9] A. Maatouk、M. Assaad 和 A. Ephremides，《错误信息年龄：语义赋能通信的推动因素》，*IEEE Transactions on Wireless Commu- nications*，第 22 卷，第 4 期，第 2621–2635 页，2023 年。[10] G. He、S. Zhang、M. Feng、S. Li 和 T. Ji ang，《面向分布式多智能体系统的错误信息年龄感知数据分发》，*IEEE Transactions on Wireless Commu- nications*，第 23 卷，第 10 期，第 15 705–15718 页，2024 年。[11] Y. Zeng 和 X. Xu，《通过信道知识图实现环境感知 6G 通信》，*IEEE Wireless Communications*，第 28 卷，第 3 期，第 84–91 页，2021 年。[12] Y. Zeng、J. Chen、J. Xu、D. Wu、X. Xu、S. Jin、X. Gao、D. Gesbert、S. Cui 和 R. Zhang，《基于信道知识图的 6G 环境感知通信教程》，

IEEE Communications Surveys & Tutorials，第 26 卷，第 3 期，第 1478–1519 页，2024年第三季度。[13] X. Xu 和 Y. Zeng，《信道知识图构建需要多少数据？》，*IEEE Transactions on Wireless Communications*，第 23 卷，第 10 期，第 13 011–13 021 页，2024 年。[14] W. Liu 和 J. Chen，《利用环境语义的无人机辅助无线电地图构建》，*IEEE Transactions on Wireless Communications*，第 22 卷，第 9 期，第 6341–6355 页，2023 年 9 月。[15] K. Katagiri，《基于统计推断的无线电地图更新与补偿》，博士论文，电气通信大学，日本东京，2022年3月，哲学博士论文。[16] D. Wu、Y. Qiu、Y. Zeng 和 F. Wen，《融合信道知识图与动态感知信息的环境感知信道估计》，*IEEE WirelessCommunications Letters*，2024 年。[17] R. D. Yates、Y. Sun、D. R. Brown、S. K. Kaul、E. Modiano 和 S. Ulukus，《信息年龄：导论与综述》，*IEEE Journal on Selected Areas in Communications*，第 39 卷，第 5 期，第 1183–1210 页，2021 年。[18] S. Sun 、 W. Wu 、 C. Fu 、 X. Qiu、J. Luo 和 J. Wang，《随机能量收集模型下多源更新网络系统的信息年龄优化》，*IEEE Journal on Selected Areas in Communications*，第 42 卷，第 3187 页，2024 年 11 月，第 3172 页，期号与页码信息原文顺序异常。[19]V. Tripathi 和 E. Modiano，《利用 Whittle 指数方法最小化信息年龄函数》，*IEEE/ACM Transactionson Networking*，第 32 卷，第 5158 页，2024 年 6 月，第 5144 页，期号与页码信息原文顺序异常。[20] I. Kadota、A. Sinha 和 E. Modiano，《面向吞吐量约束无线网络中信息年龄优化的调度算法》，*IEEE/ACM Transactionson Networking*，第 27 卷，第 4 期，第 1359–1372 页，2019 年。[21] F. Peng、X. Wang 和 X. Chen，《随机时延信道上状态更新的目标导向通信》，载于 *IEEE INFOCOM 2024 - IEEE Conference on Computer Communi- cations Workshops (INFOCOM WKSHPS)*，2024 年，第 1–6 页。[22] O. Li、F. Zhang、C. He、C. Xu 和 X. Wang，《物联网系统中状态更新的上下文感知信息价值》，载于 *2024*年*IEEE/CIC*中国通信国际会议 *(ICCC)*。IEEE，2024 年，第 1745–1750 页。[23] J.-P. Crouzeix 和 J. A. Ferland，《广义分式规划算法》，*Mathematicalprogramming*，第 52 卷，第 1 期，第 191–207 页，1991 年。[24] Y. Xiao、Q. Du、W. Cheng 和 W. Zhang，《面向元宇宙信息年龄最小化的自适应采样与传输》，*IEEE Journal on Selected Areas in Communications*，第 42 卷，第 3 期，第 588–602 页，2023 年。[25] R. Han、C. Zhang 和 T. Wang，《信息视角下的信道知识图更新分析》，载于 *2025*年*IEEE/CIC*中国通信国际会议 *(ICCC Workshops)*，2025 年，第 1–6 页。[26] C. Ferguson 和 L. Kleinrock，《包含信息年龄在内的陈旧信息指标的最优更新时间》，*IEEE Journal on Selected Areas in Information Theory*，第 4 卷，第 734–746 页，2023 年。[27] M. K. C. Shisher 和 Y. Sun，《数据新鲜度如何影响实时监督学习？》，载于第二十三届移动网络与移动计算理论、算法基础及协议设计国际研讨会论文集，2022 年，第 31–40 页。[28] W. Dinkelbach，《关于非线性分式规划》，*Man- agement Science*，第 13 卷，第 7 期，第 492–498 页，1967 年。[29] M. Ehrgott，《多准则优化》，第 2 版。柏林：Springer，2005 年。[30] E. Q. V. Martins，《关于多准则最短路径问题》，*European Journalof OperationalResearch*，第 16 卷，第 2 期，第 236–245 页，1984 年。[31] P. Hansen，《双准则路径问题》，*Journal of Optimiza- tion Theory and Applications*，第 31 卷，第 133 页，1979 年，第 1 期，第 109 页，期号与页码信息原文顺序异常。