arXiv:2302.03006v1 [cs.IT] 6 Feb 2023

# 基于信息年龄的八卦传播中的可靠与不可靠的信息源

Priyanka Kaswan Sennur Ulukus 电气与
计算机工程系 马里兰大学，大学公园，马里兰州
20742*pkaswan@umd.edu* *ulukus@umd.edu*

**摘要—我们考虑一个由 $n$ 个节点组成的网络，这些节点旨在跟踪持续更新的过程或事件。为了向网络传播有关该事件的更新，有两个可用的信息源，其中一个信息源提供的信息比另一个更可靠。节点希望获取有关该事件的信息，这些信息不仅要最新，也要更可靠；即使可靠数据包相较于不可靠数据包稍显过时，节点仍更倾向于选择可靠数据包。我们研究这种偏好如何影响网络中拥有可靠信息的用户比例及其信息版本年龄。我们利用随机混合系统（SHS）建模，推导出用于刻画这两个量的解析方程，即长期期望的、拥有可靠数据包的节点比例以及其长期期望版本年龄，并研究它们的性质。我们还将这些结果与节点更重视信息新鲜度而非可靠性的情况进行了比较。最后，我们展示仿真结果，以验证理论结果，并进一步揭示上述量与相关变量之间的关系。**

## I. 引言

我们考虑这样一个系统：一组 $n$ 节点希望跟踪某个事件或过程（E），该事件或过程按照速率为 $\lambda_E$的泊松过程进行更新。然而，为了向节点传递有关该事件的信息，有两个信息源可供使用，其中一个比另一个更可靠，预计能够传递更准确的信息。我们将前者称为可靠（R）信息源，将后者称为不可靠（U）信息源；它们分别以总更新速率 $\lambda_R$ 和 $\lambda_U$向网络发送数据包。这种情形可能出现在多个传感器监测物理环境并向物联网发送更新，而其中一些传感器并不可靠的情况下。另一个例子是，多个网站或新闻来源向一群感兴趣的观众播报某项体育赛事的比分，其中一些信息源比其他信息源更可靠。

我们注意到，在上述情形中，感兴趣的事件本质上是动态的。对于时间敏感的动态信息，
息，网络节点通常希望获取尽可能新的信息，这可以通过文献中常用的时效性指标来量化，例如信息年龄 [1]–[4],信息版本年龄 [5], [6], 二元新鲜度 [7]–[9], 错误信息年龄 [10], 等。在本研究中，我们采用信息版本年龄指标，因为我们为有关该事件的每条信息都关联了一个版本号。如果 $V_E(t)$ 表示对应于

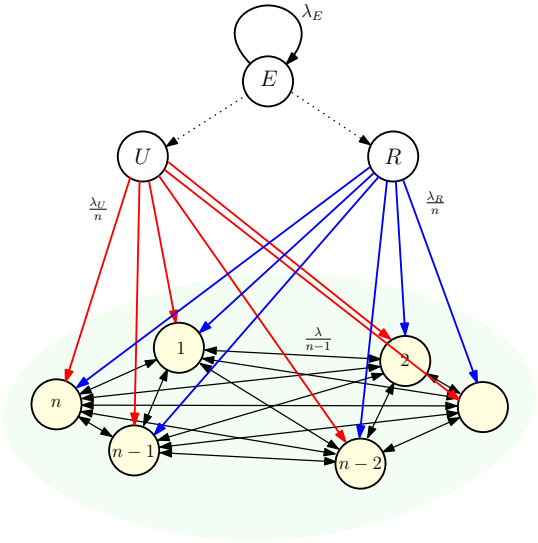



图 1. 系统模型：可靠（R）和不可靠（U）信息源观测一个以速率 $\lambda_E$ 更新的事件（E）。可靠（R）和不可靠（U）信息源分别以总速率 $\lambda_R$ 和 $\lambda_U$ 向网络传播信息。每个用户以速率 $\lambda$ 与其邻居进行八卦传播，网络为全连接网络。

事件当前状态的版本号，而 $V_i(t)$ 表示节点 $i$处有关该事件的信息的版本号，那么节点 $i$ 处的信息瞬时版本年龄定义为 $X_i(t) = V_E(t) - V_i(t)$，其中 $V_E(t)$ 每当事件更新时便递增一。

除了信息源向网络传播更新外，网络中的节点还旨在通过八卦传播 [4], [5], [9], [11]–[23] 加快传播，从而改善其信息的版本年龄和信息可靠性。八卦算法是一种分布式算法，节点会随机联系其邻居以交换数据包；从传播时间角度 [13], [15]–[17] 和及时性角度 [4]–[6], [9], [20], [21], [24] 对采用八卦协议的网络进行了广泛研究。

与本文最密切相关的研究包括 [4]–[6], [9], [20], [21], [24]。 [4], [5] 分别利用随机混合系统（SHS）框架推导了期望年龄和期望版本年龄的递归线性方程， [20] 研究了簇状八卦网络中的期望版本年龄， [9] 给出了二元新鲜度指标的类似结果， [21]尝试利用文件切片和

网络编码改进八卦网络中的年龄界，而 [6]和 [24]研究了干扰攻击和时间戳篡改对手对八卦网络的影响。上述研究均假设只有一个信息源负责向网络传输最新的信息数据包，节点交换数据包的唯一目标是提升信息新鲜度。

本文考虑两种信息源：可靠源和不可靠源。假设它们始终拥有关于该事件的最新信息，因此各自的信息版本年龄 $X_R(t)$ 和 $X_U(t)$ 始终为零。用户节点希望获取尽可能最新版本的信息，并且更偏好可靠信息，即源自可靠源的信息。令 $S_i(t)$ 表示在时刻 $t$ 节点 $i$ 上的信息数据包的可靠性状态，其中 $S_i(t)=0$ 和 $S_i(t)=1$ 分别对应可靠数据包和不可靠数据包。在时刻 $t$，如果节点 $i$ 向节点 $j$ 发送更新，节点 $j$ 将根据以下规则决定接受还是拒绝该数据包：

- 如果 $S_i(t)=1$ 且 $S_j(t)=1$，即两个节点都持有不可靠信息，那么节点 $j$ 选择信息年龄较小的数据包。
- 如果 $S_i(t)=0$ 且 $S_j(t)=0$，即两个节点都持有可靠信息，那么节点 $j$ 会再次选择信息版本年龄较低的数据包。
- 如果 $S_i(t)=0$ 且 $S_j(t)=1$，即传入数据包包含可靠信息，但节点 $j$ 持有不可靠信息，那么只要满足 $X_i \le X_j+1$，节点 $j$ 就会选择传入的可靠数据包。换言之，传入数据包的版本最多比节点 $j$ 当前持有的数据包旧一个版本。
- 如果 $S_i(t)=1$ 且 $S_j(t)=0$，即节点 $j$ 已经持有可靠信息数据包，而传入的数据包不可靠，那么只要满足 $X_j \le X_i+1$，节点 $j$ 就会继续保留其可靠数据包。换言之，节点 $j$ 持有的可靠数据包最多比传入的不可靠数据包旧一个版本。

我们首先研究，在网络按照上述协议交换文件时，网络中的用户节点平均有多大比例持有不可靠信息，以及这对用户节点处信息版本年龄的平均值有何影响。为此，我们将该问题建模为一个随机混合系统。我们还研究另一种情形：用户节点优先考虑信息新鲜度，并始终选择信息版本年龄最低的数据包。在后一种情形中，只有当被比较的两个数据包信息版本年龄相同但可靠性状态不同时，数据包的可靠性状态才会起作用；此时将选择可靠数据包。我们研究结果对各种网络参数的依赖关系，并比较这两种情形下的结果。最后，我们给出仿真结果，以验证理论结果并提供进一步的见解。

## II. 系统模型与 SHS 特征描述

系统模型由一个可靠信息源（$R$）和一个不可靠信息源（$U$）组成，它们向一组 $n$用户节点 $\mathcal{N}=\{1,\dots,n\}$ 传输有关某个过程或事件（E）的更新，该过程或事件按速率为 $\lambda_E$的泊松过程进行更新，见图 1。假设可靠信息源和不可靠信息源始终能够获取最新版本的信息，并分别以速率 $\frac{\lambda_R}{n}$和$\frac{\lambda_U}{n}$向用户节点 $i\in\mathcal{N}$发送更新。此外，对于每个 $i,j\in\mathcal{N}$，节点 $i$ 按照速率为$\lambda_{ij}=\frac{\lambda}{n-1}$ 的泊松过程向节点 $j$ 发送更新，作为底层八卦网络的一部分。令$F$（$t$）表示在时刻 $t$持有不可靠信息数据包的用户节点比例，则

$$F(t)=\frac{S_1(t)+S_2(t)+\ldots+S_n(t)}{n} \tag{1}$$

由于 $S_i(t)=1$ 表示节点 $i$ 持有不可靠信息。我们关注当节点按照第 I 节所述协议接收数据包时，长期期望值 $F=\lim_{t\to\infty}\mathbb{E}[F(t)]$ 和 $x_i=\lim_{t\to\infty}\mathbb{E}[X_i(t)],i\in\mathcal{N}$的特征描述。注意，所有用户节点上的可靠性状态和版本年龄过程在统计上相同，因此有 $x_1=\ldots=x_n$和 $F=s_1$，其中$s_i=\lim_{t\to\infty}\mathbb{E}[S_i(t)]$。接下来，我们使用 SHS 建模 [25] 来建立一组线性方程，以求得 $s_1$ 和 $x_1$。

在这方面，我们将随机混合系统模型的连续状态选为 $(\boldsymbol{S}(t)\ \boldsymbol{X}(t))\in\mathbb{R}^{2n}$，其中 $\boldsymbol{S}(t)=[S_1(t)\dots,S_n(t)]$和 $\boldsymbol{X}(t)=[X_1(t)\dots,X_n(t)]$ 分别表示时间 $t$时 $n$ 用户节点的瞬时可靠性状态和瞬时版本年龄。当节点 $i$向节点 $j$发送更新包时，我们称转移（$i,j$）发生，其中（$E,E$）表示事件更新，（$U,i$）和（$R,i$）分别表示向用户节点 $i$ 发送来自不可靠信息源和可靠源的更新。由于 $S_i(t)$ 和 $X_i(t)$ 在转移之间保持不变，随机混合系统模型仅在一个离散状态下运行，其连续状态遵循微分方程（$\boldsymbol{S}(t)\ \boldsymbol{X}(t)$）$=\mathbf{0}_{2n}$。转移集合为

$$\begin{aligned}\mathcal{L}=\{(E,E)\}\cup\{(U,i):i\in\mathcal{N}\}\cup\{(R,i):i\in\mathcal{N}\}\\ \cup\{(i,j):i,j\in\mathcal{N}\}\end{aligned} \tag{2}$$

使得转移 $(i,j)$ 在时间 $t$ 将状态 $(\boldsymbol{S},\boldsymbol{X})$ 重置为 $\phi_{i,j}(\boldsymbol{S},\boldsymbol{X},t)\in\mathbb{R}^{2n}$ 转移后。每个转移 $(i,j)$ 的速率 $\lambda_{ij}$ 给定为

$$\lambda_{ij}=\begin{cases}\frac{\lambda_U}{n}, & i=U,j\in\mathcal{N}\\ \frac{\lambda_R}{n}, & i=R,j\in\mathcal{N}\\ \frac{\lambda}{n-1}, & i,j\in\mathcal{N}\\ \lambda_E, & i=E,j=E\end{cases} \tag{3}$$

接下来，我们定义一些稍后会用到的变量。考虑一个由 $A$ 个节点组成的集合。其中一些节点可能拥有可靠信息，另一些则可能拥有不可靠信息。令 $R(A)\subseteq A$ 表示拥有可靠信息的节点子集，并令 $U(A)\subseteq A$ 表示拥有不可靠信息的节点子集。对于一个

连续状态（$\boldsymbol{S}$, $\boldsymbol{X}$）和节点集合 $A$, 我们将集合 $A$, $X_A$ 的版本年龄定义如下:

- If $A = \varnothing$, then $X_A = \infty$.
- If $A = R(A)$ or $A = U(A)$, then $X_{\arg\min_{j\in A} X_j}$.
- If $X_{\arg\min_{j\in R(A)} X_j} \leq X_{\arg\min_{j\in U(A)} X_j} + 1$, then $X_A = X_{\arg\min_{j\in R(A)} X_j}$.
- If $X_{\arg\min_{j\in U(A)} X_j} \leq X_{\arg\min_{j\in R(A)} X_j} - 2$, then $X_A = X_{\arg\min_{j\in U(A)} X_j}$.

接下来，我们将集合 $A$, $S_A$ 的可靠性状态定义如下:

- If $X_{R(A)} \leq X_{U(A)} + 1$, then $S_A = S_{\arg\min_{j\in R(A)} X_j}$.
- If $X_{U(A)} \leq X_{R(A)} - 2$, then $S_A = S_{\arg\min_{j\in U(A)} X_j}$.

也就是说，只要集合中最新可靠数据包的版本不比最新不可靠数据包的版本落后一版以上，拥有最新可靠数据包的节点就决定 $X_A$ 和 $S_A$。根据此定义，基于时刻 $t$ 的转移 $(i, j)$，到 $\phi_{i,j}(\boldsymbol{S}, \boldsymbol{X}, t)$ 的重置映射 $= [S'_1, \ldots, S'_n, X'_1, \ldots, X'_n] \in \mathbb{R}^{2n}$可描述为

$$S'_\ell = \begin{cases} S_{\{U,\ell\}}, & i = U, j \in \mathcal{N}, \ell = j \\ 0, & i = R, j \in \mathcal{N}, \ell = j \\ S_{\{i,\ell\}}, & i, j \in \mathcal{N}, \ell = j \\ S_\ell, & \text{otherwise} \end{cases} \tag{4}$$

and

$$X'_\ell = \begin{cases} X_\ell + 1, & i = E, j = E, \ell = j \\ \mathbb{1}_{\{X_{R(\{\ell\})}=1\}}, & i = U, j \in \mathcal{N}, \ell = j \\ 0, & i = R, j \in \mathcal{N}, \ell = j \\ X_{\{i,\ell\}}, & i, j \in \mathcal{N}_R, \ell = j \\ X_\ell, & \text{otherwise} \end{cases} \tag{5}$$

其中， $\mathbb{1}_{\{.\}}$ 表示指示函数。接下来，我们将选取一系列测试函数 $\psi: \mathbb{R}^{2n} \times [0, \infty) \to \mathbb{R}$，这些函数与时间无关，即它们关于 $t$ 的偏导数为 $\partial\psi(\boldsymbol{x},\boldsymbol{U},t)_{\partial t} = 0$，使得其长期期望值 $\mathbb{E}[\psi] = \lim_{t\to\infty} \mathbb{E}[\psi(\boldsymbol{S}(t)\ \boldsymbol{X}(t)\ t)]$ 有助于后续分析。由于测试函数仅依赖于连续状态值 $(\boldsymbol{S}, \boldsymbol{X})$，且与时间无关，为简单起见，我们省略第三个输入 $t$，并将 $\psi(\boldsymbol{S}, \boldsymbol{X}, t)$ 写作 $\psi(\boldsymbol{S}, \boldsymbol{X})$，并假设其满足 $\dot{\psi}(\boldsymbol{S}(t)\ \boldsymbol{X}(t)) = 0$。定义 $\mathbb{E}[\psi(\phi_{i,j})] = \lim_{t\to\infty} \mathbb{E}[\psi(\phi_{i,j}(\boldsymbol{S}(t)\ \boldsymbol{X}(t)\ t))]$, [25,, 由定理 1] 可得

$$0 = \sum_{(i,j)\in\mathcal{L}} (\mathbb{E}[\psi(\phi_{i,j})] - \mathbb{E}[\psi])\lambda_{ij} \tag{6}$$

这与 [4], [5], [24], 中的推导类似：由于 $d\mathbb{E}[\psi(\boldsymbol{s}(t\ \boldsymbol{X}\ t\ t]_{dt} = 0)())$在较大的 $t$ 下随着期望趋于稳定，左侧被设为零。我们将通过定义一系列适用于本分析的时不变测试函数，反复使用该方程。更多详情请参阅参考文献 [25] 和 [4]。

## III. 可靠性与版本年龄分析

由于所有用户节点的版本年龄和可靠性状态演化过程都相同，我们用 $A_k$ 表示典型的 $k$ 用户节点集合。我们的第一个测试函数为 $\psi(\boldsymbol{S}, \boldsymbol{X}) = S_{A_k}$，在转移 $(i, j)$ 后修改为 $\psi(\phi_{i,j}(\boldsymbol{S}, \boldsymbol{X}, t)) = S'_{A_k}$，并可用式 (4)、(5) 表征如下，

$$S'_{A_k} = \begin{cases} S_{A_k\cup\{U\}}, & i = U, j \in A_k \\ 0, & i = R, j \in A_k \\ S_{A_{k+1}}, & i = \mathcal{N}\backslash A_k, j \in A_k \\ S_{A_k}, & \text{otherwise} \end{cases} \tag{7}$$

定义 $a_k = \lim_{t\to\infty} \mathbb{E}[S_{A_k}(t)]$ 和 $b_k = \lim_{t\to\infty} \mathbb{E}[S_{A_k\cup\{U\}}(t)]$, , 并使用 (6)，可得:

$$\begin{aligned} 0 =& (b_k - a_k)\frac{k\lambda_U}{n} + (0 - a_k)\frac{k\lambda_R}{n} \\ &+ (a_{k+1} - a_k)\frac{k(n-k)\lambda}{n-1} \end{aligned} \tag{8}$$

接下来，我们研究测试函数 $\psi(\boldsymbol{S}, \boldsymbol{X}) = S_{A_k\cup\{U\}}$，其 $(i, j)$ 转移映射如下:

$$S'_{A_k\cup\{U\}} = \begin{cases} 1 - \mathbb{1}_{\{X_{R(A_k)}=0\}}, & i = E, j = E \\ 0, & i = R, j \in A_k \\ S_{A_{k+1}\cup\{U\}}, & i = \mathcal{N}\backslash A_k, j \in A_k \\ S_{A_k\cup\{U\}}, & \text{otherwise} \end{cases} \tag{9}$$

定义 $c_k = \lim_{t\to\infty} \mathbb{E}[\mathbb{1}_{\{X_{R(A_k)}(t)=0\}}]$, 由 (6) 可得:

$$\begin{aligned} 0 =& (1 - c_k - b_k)\lambda_E + (0 - b_k)\frac{k\lambda_R}{n} \\ &+ (b_{k+1} - b_k)\frac{k(n-k)\lambda}{n-1} \end{aligned} \tag{10}$$

最后，我们研究测试函数 $\psi(\boldsymbol{S}, \boldsymbol{X}) = \mathbb{1}_{\{X_{R(A_k)}=0\}}$，其 $(i, j)$ 转移映射如下，

$$\mathbb{1}'_{\{X_{R(A_k)}=0\}} = \begin{cases} 0, & i = E, j = E \\ 1, & i = R, j \in A_k \\ \mathbb{1}_{\{X_{R(A_{k+1})}=0\}}, & i = \mathcal{N}\backslash A_k, j \in A_k \\ \mathbb{1}_{\{X_{R(A_k)}=0\}}, & \text{otherwise} \end{cases} \tag{11}$$

应用 (6) 后可得,

$$\begin{aligned} 0 =& (1 - c_k)\lambda_E + (1 - c_k)\frac{k\lambda_R}{n} \\ &+ (c_{k+1} - c_k)\frac{k(n-k)\lambda}{n-1} \end{aligned} \tag{12}$$

方程 (8)、(10)、(12) 可改写如下,

$$a_k = \frac{b_k\frac{k\lambda_U}{n} + a_{k+1}\frac{k(n-k)\lambda}{n-1}}{\frac{k\lambda_U}{n} + \frac{k\lambda_R}{n} + \frac{k(n-k)\lambda}{n-1}} \tag{13}$$

$$b_k = \frac{(1 - c_k)\lambda_E + b_{k+1}\frac{k(n-k)\lambda}{n-1}}{\lambda_E + \frac{k\lambda_R}{n} + \frac{k(n-k)\lambda}{n-1}} \tag{14}$$

$$c_k = \frac{\frac{k\lambda_R}{n} + c_{k+1}\frac{k(n-k)\lambda}{n-1}}{\lambda_E + \frac{k\lambda_R}{n} + \frac{k(n-k)\lambda}{n-1}} \tag{15}$$

请注意， $a_1 = s_1 = F$，因此，在计算

$F$时，我们首先根据（15）求解 $c_k$ ，从 $k = n$ 开始，依次代入 $k = n-1, \ldots, 1$进行迭代计算。计算出所有 $c_k$ 后，我们可以类似地根据（14）反向迭代，计算出所有 $b_k$。最后，我们可以利用 $b_k$ 和归纳方程（13）计算出所有$a_k$，从 $a_n$ 开始，最终得到 $a_1 = F$ 。

从方程（13）、（14）和（15）中可以得到一些观察结果。首先，如果 $\lambda_R$ 很大，那么将 $\lambda_R = \infty$ 代入（13）可得 $a_1 = 0$，即所有节点始终都拥有可靠信息。另一方面，如果 $\lambda_U \to \infty$，那么 $a_1 \approx b_1$，其中 $b_1$ 完全独立于 $\lambda_U$ ，是一个取决于其他转移率和网络规模 $n$ 的正值。因此，当 $\lambda_U$ 很大时，预计仍有非零比例的节点拥有可靠信息。这可以归因于以下事实：即使这意味着恢复到信息的较旧版本，节点仍偏好可靠信息；只要 $\lambda_R$ 非零，一些节点就能够成功丢弃不可靠数据包，转而采用可靠数据包。此外，如果八卦传播速率 $\lambda$ 较高，那么 $a_1 \approx \ldots \approx a_n$，从而导致

$$\begin{aligned} a_1 \approx a_n \approx b_n \frac{\lambda_U}{\lambda_U + \lambda_R} &\approx \frac{(1-c_n)\lambda_E}{\lambda_E + \lambda_R} \frac{\lambda_U}{\lambda_U + \lambda_R} \\ &\approx \left(\frac{\lambda_E}{\lambda_E + \lambda_R}\right)^2 \frac{\lambda_U}{\lambda_U + \lambda_R} \end{aligned} \tag{16}$$

人们可能会认为，较高的流言传播速率会使新鲜可靠信息在网络中快速传播，并使 $F$趋于 0；然而，如式 (16) 所示， $F$在这种情况下仍取决于其他因素，例如可靠源在事件更新后多久能够开始发送数据包，以及不可靠源在这方面是否更快。

接下来，我们刻画用户节点处的版本年龄。我们定义 $e_k = \lim_{t\to\infty} \mathbb{E}[X_{A_k}(t)]$ 和 $d_k = \lim_{t\to\infty} \mathbb{E}[\mathbb{1}_{\{X_{R(A_k)}(t)=1\}}]$。我们需要两个新的测试函数 $X_{A_k}$ 和 $\mathbb{1}_{\{X_{R(A_k)}}$,, 它们经转移 ($i$, $j$) 修改如下，

$$X'_{A_k} = \begin{cases} X_{A_k} + 1, & i = E, j = E \\ \mathbb{1}_{\{X_{R(A_k)}=1\}}, & i = U, j \in A_k \\ 0, & i = R, j \in A_k \\ X_{A_{k+1}}, & i = \mathcal{N}\backslash A_k, j \in A_k \\ X_{A_k}, & \text{otherwise} \end{cases} \tag{17}$$

$$\mathbb{1}'_{\{X_{R(A_k)}=1\}} = \begin{cases} \mathbb{1}_{\{X_{R(A_k)}=0\}}, & i = E, j = E \\ 0, & i = R, j \in A_k \\ \mathbb{1}_{\{X_{R(A_{k+1})}=1\}}, & i = \mathcal{N}\backslash A_k, j \in A_k \\ S_{A_k\cup\{U\}}, & \text{otherwise} \end{cases} \tag{18}$$

利用式 (6)，可得线性方程，

$$\begin{aligned} 0 =& (e_k + 1 - e_k)\lambda_E + (d_k - e_k)\frac{k\lambda_U}{n} + (0 - e_k)\frac{k\lambda_R}{n} \\ &+ (q_{k+1} - e_k)\frac{k(n-k)\lambda}{n-1} \end{aligned} \tag{19}$$

$$\begin{aligned} 0 =& (c_k - d_k)\lambda_E + (0 - d_k)\frac{k\lambda_R}{n} \\ &+ (d_{k+1} - d_k)\frac{k(n-k)\lambda}{n-1} = 0 \end{aligned} \tag{20}$$

将其重新排列后，再结合式 (15)，可得以下方程组，

$$e_k = \frac{\lambda_E + d_k\frac{k\lambda_U}{n} + e_{k+1}\frac{k(n-k)\lambda}{n-1}}{\frac{k\lambda_U}{n} + \frac{k\lambda_R}{n} + \frac{k(n-k)\lambda}{n-1}} \tag{21}$$

$$d_k = \frac{c_k\lambda_E + d_{k+1}\frac{k(n-k)\lambda}{n-1}}{\lambda_E + \frac{k\lambda_R}{n} + \frac{k(n-k)\lambda}{n-1}} \tag{22}$$

$$c_k = \frac{\frac{k\lambda_R}{n} + c_{k+1}\frac{k(n-k)\lambda}{n-1}}{\lambda_E + \frac{k\lambda_R}{n} + \frac{k(n-k)\lambda}{n-1}} \tag{23}$$

与之前一样，我们首先计算从 $k = n$ 开始的所有 $c_k$ ，随后便可迭代计算所有 $d_k$ ，进而计算所有 $e_k$，最终得到版本年龄$x_1 = e_1$。同样，如果 $\lambda_R$ 很大，就会导致 $e_1 \approx 0$，因为网络中充满了新鲜可靠数据包。另一方面，如果 $\lambda_U \to \infty$，那么 $e_1 \approx d_1$，其中 $d_k$ 是一个与 $\lambda_U$ 无关的正数。也就是说，用户节点的信息版本年龄并不保证在不可靠源以较高速率发送更新时降至零，因为年龄为一的可靠数据包比年龄为零的不可靠数据包更受青睐。

### IV. 更高优先级的新鲜信息

为了进一步探讨，我们现在研究用户节点将新鲜信息置于最高优先级的情形。这意味着，当用户节点收到数据包时，其首要目标是保留信息版本年龄最低的数据包；只有当两个数据包的版本年龄相同但可靠性状态不同时，节点才会优先选择可靠数据包而非不可靠数据包。假设网络拓扑和转移率与第二节相同，我们将 $X_A$ 的定义修改如下：

- If $A = \varnothing$, then $X_A = \infty$, else $X_A = X_{\arg\min_{j\in A} X_j}$.

我们将 $S_A$ 的定义修改如下：

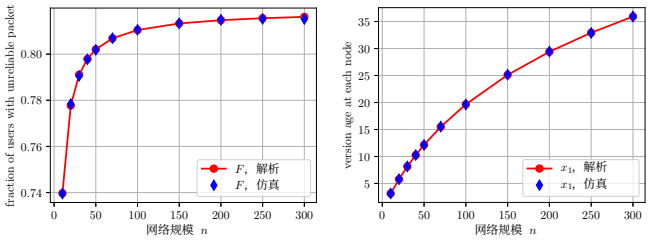
- If $X_{R(A)} \le X_{U(A)}$, then $S_A = S_{\arg\min_{j\in R(A)} X_j}$, else $S_A = S_{\arg\min_{j\in U(A)} X_j}$.

所关注的测试函数为 $S_{A_k}$ 和 $S_{A_k\cup\{U\}}$ ，其转移映射为（$i$, $j$），

$$S'_{A_k} = \begin{cases} S_{A_k\cup\{U\}}, & i = U, j \in A_k \\ 0, & i = R, j \in A_k \\ S_{A_{k+1}}, & i = \mathcal{N}\backslash A_k, j \in A_k \\ S_{A_k}, & \text{otherwise} \end{cases} \tag{24}$$

and

$$S'_{A_k\cup\{U\}} = \begin{cases} 1, & i = E, j = E \\ 0, & i = R, j \in A_k \\ S_{A_{k+1}\cup\{U\}}, & i = \mathcal{N}\backslash A_k, j \in A_k \\ S_{A_k\cup\{U\}}, & \text{otherwise} \end{cases} \tag{25}$$




(a) (b)

图 2. 对比分析结果与仿真结果：(a) 信息不可靠用户的预期比例 $F$。(b) 期望版本年龄 $x_1$。

定义 $\bar{a}_k = \lim_{t\to\infty}\mathbb{E}[S_{A_k}(t)]$ 和 $\bar{b}_k = \lim_{t\to\infty}\mathbb{E}[S_{A_k\cup\{U\}}(t)]$，，并对上述测试函数应用 (6)，得到

$$\bar{a}_k = \frac{\bar{b}_k\frac{k\lambda_U}{n} + \bar{a}_{k+1}\frac{k(n-k)\lambda}{n-1}}{\frac{k\lambda_U}{n} + \frac{k\lambda_R}{n} + \frac{k(n-k)\lambda}{n-1}} \tag{26}$$

$$\bar{b}_k = \frac{\lambda_E + (\bar{b}_{k+1})\frac{k(n-k)\lambda}{n-1}}{\lambda_E + \frac{k\lambda_R}{n} + \frac{k(n-k)\lambda}{n-1}} \tag{27}$$

在这种情况下，计算用户节点的版本年龄更简单，只需要一个测试函数 $X_{A_k}$ with $\bar{e}_k = \lim_{t\to\infty}\mathbb{E}[X_{A_k}(t)]$。相应的转移映射和线性方程为

$$X'_{A_k} = \begin{cases} \bar{X}_{A_k}+1, & i=E, j=E \\ 0, & i=U, j\in A_k \\ 0, & i=R, j\in A_k \\ X_{A_{k+1}}, & i=\mathcal{N}\backslash A_k, j\in A_k \\ X_{A_k}, & \text{otherwise} \end{cases} \tag{28}$$

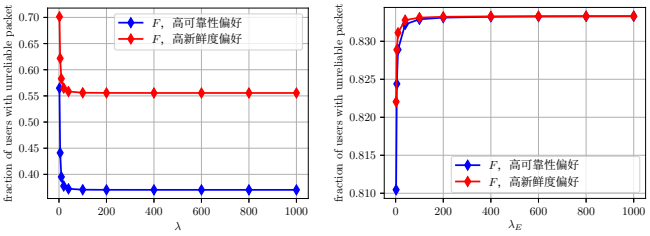
and

$$\bar{e}_k = \frac{\lambda_E + \bar{e}_{k+1}\frac{k(n-k)\lambda}{n-1}}{\frac{k\lambda_U}{n} + \frac{k\lambda_R}{n} + \frac{k(n-k)\lambda}{n-1}} \tag{29}$$

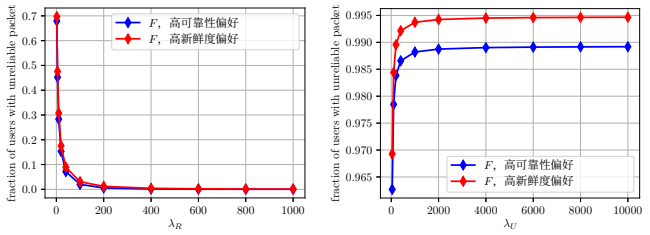
将前一设置中的 $\bar{e}_k$ 和 $\bar{a}_k$ 与 $e_k$ 和 $a_k$ 进行比较，可以看到，从 $k=n$开始迭代时，$\bar{e}_k \geq e_k$ 和$\bar{b}_k \leq b_k$，从而导致 $\bar{a}_k \leq a_k$。因此，信息可靠性与版本年龄之间存在权衡。不过，当 $\lambda_E$很大时，根据式 (22) 和 (23)，$d_k \to 0$ 和 $c_k \to 0$，因此对于较大的 $\lambda_E$，$\bar{e}_k \approx e_k$。这是因为事件更新得太快，网络中的所有数据包都很快过时，网络中不再有版本年龄较小（为 1）的可靠数据包，导致 $d_k \approx 0$。

## V. 数值结果

我们对不同网络规模 $n$ 的全连接网络进行仿真，参数为 $\lambda_E = 2$、$\lambda_U = 5$、$\lambda_R = 1$ 和 $\lambda = 0.1$，仿真总时长最长为 $10^6$，并将其作为 $t\to\infty$的代理变量。我们在图 2 中，将 $F$ 和 $x_1$ 的仿真点（蓝点）绘制在由式 (13)、(14)、(15)、(21)、(22)得到的曲线（红线）上。实时仿真点与通过迭代计算得到的方程曲线相吻合，为理论结果提供了支持，并表明 $F$ 和 $x_1$均随网络规模 $n$增大。

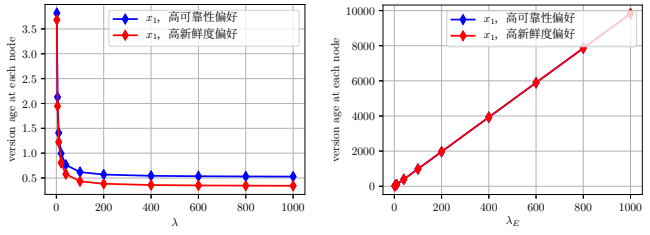



(a) (b)

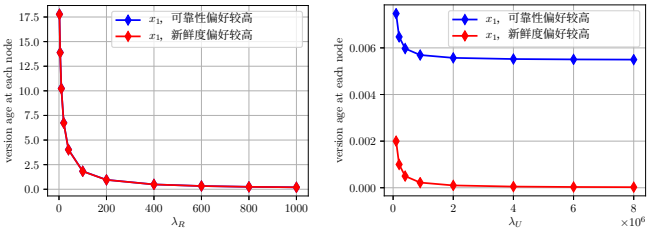



(c) (d)

图3. $F$ 作为网络参数 $\lambda$, $\lambda_E$, $\lambda_R$ 和 $\lambda_U$的函数。




(a) (b)




(c) (d)

图4. $x_1$ 作为网络参数 $\lambda$, $\lambda_E$, $\lambda_R$ 和 $\lambda_U$的函数。

接下来，我们分别在图3和图4中绘制节点数为 $n=100$ 时的 $F$ 和 $x_1$ 随网络参数 $\lambda_E$、$\lambda_U$、$\lambda_R$ 和 $\lambda$ 变化的情况。图3(a)显示，正如式（16）所预测的，$F$ 收敛到非零值 $\left(\frac{\lambda_E}{\lambda_E+\lambda_R}\right)^2\frac{\lambda_U}{\lambda_U+\lambda_R} = 0.37$。当 $\lambda_R$ 较大时，正如第III节所述，图3(c)和图4(c)中的 $F\to 0$ 和 $x_1\to 0$ 分别如此；而 $\lambda_U$ 较大并不会产生相同的效果，图3(d)中 $F$ 收敛到严格小于1的值。此外，从图4(d)可以看出，正如第III节所述，在高可靠性偏好情况下，版本年龄不会收敛到零；但在高新鲜度情况下，它会降至零，因为这种情况下优先获取任何可能降低其年龄的数据包。最后，随着 $\lambda_E$ 增大，我们看到高可靠性偏好和高新鲜度偏好两种情况下的 $x_1$ 重合，这与第IV节的预测一致；并且它随 $\lambda_E$ 线性增长，因为迭代（29）中的所有项都含有公因子 $\lambda_E$。

# 参考文献


[1] A. Kosta、N. Pappas 和 V. Angelakis。《信息年龄：一种新概念、指标和工具》。《网络基础与趋势》，12(3):162–259，2017年11月。[2] Y. Sun、I. Kadota、R. Talak 和 E. Modiano。《信息年龄：衡量信息新鲜度的新指标》。《通信网络综合讲座》，12(2):1–224，2019年12月。[3] R. D. Yates、Y. Sun、R. Brown、S. K. Kaul、E. Modiano 和 S. Ulukus。《信息年龄：介绍与综述》。《IEEE通信领域选刊》，39(5):1183–1210，2021年5月。[4] R. D. Yates。《及时谣言传播》。载于 IEEE SPAWC，2021年9月。[5] R. D. Yates。《网络中的谣言传播年龄》。载于 IEEE ISIT，2021年7月。[6] P. Kaswan 和 S. Ulukus。《存在干扰攻击时环形网络中的谣言传播年龄》。载于阿西洛马会议，2022年10月。[7] M. Bastopcu 和 S. Ulukus。《缓存更新系统中的信息新鲜度》。《IEEE无线通信汇刊》，20(3):1861–1874，2021年3月。[8] P. Kaswan、M. Bastopcu 和 S. Ulukus。《基于新鲜度的并行中继网络缓存更新》。载于 IEEE ISIT，2021年7月。[9] M. Bastopcu、B. Buyukates 和 S. Ulukus。《采用二元新鲜度指标的八卦传播》。载于 IEEE Globecom，2021年12月。[10] A. Maatouk、S. Kriouile、M. Assaad 和 A. Ephremides。《错误信息年龄：一种状态更新的新性能指标》。《IEEE/ACM网络汇刊》，28(5):2215–2228，2020年10月。[11] A. J. Demers、D. H. Greene、C. H. Hauser 等。《用于复制数据库维护的流行病算法》。载于 ACM PODC，1987年8月。[12] Y. Minsky。《廉价、快速且可靠地传播谣言》。博士论文，康奈尔大学，2002年3月。[13] R. Karp、C. Schindelhauer、S. Shenker 和 B. Vocking。《随机化谣言传播》。载于 FOCS，2000年11月。[14] B. G. Pittel。《论谣言传播》。《SIAM应用数学杂志》，47(1):213–223，1987年2月。[15] S. Deb、M. Medard 和 C. Choute。《代数谣言传播：一种实现多谣言最优传播的网络编码方法》。《IEEE信息论汇刊》，52(6):2486–2507，2006年6月。[16] D. Mosk-Aoyama 和 D. Shah。《通过网络编码进行信息传播》。载于 IEEE ISIT，2006年7月。[17] S. Sanghavi、B. Hajek 和 L. Massoulie。《传播多条消息的八卦传播》。《IEEE信息论汇刊》，53(12):4640–4654，2007年12月。[18] G. DeCandia、D. Hastorun 等。《Dynamo：亚马逊的高可用性键值存储》。《ACM SIGOPS操作系统评论》，41(6):205–220，2007年。[19] A. Lakshman 和 P. Malik。《Cassandra：一种去中心化结构化存储系统》。《ACM SIGOPS操作系统评论》，44(2):35–40，2010年。[20] B. Buyukates、M. Bastopcu 和 S. Ulukus。《具有社区结构的网络中的谣言传播年龄》。载于 IEEE SPAWC，2021年9月。[21] P. Kaswan 和 S. Ulukus。《结合文件切片与网络编码的及时八卦传播》。载于 IEEE ISIT，2022年6月。[22] P. Mitra 和 S. Ulukus。《ASUMAN：网络中的年龄感知更新多址接入》。载于阿勒顿会议，2022年9月。[23] M. Bastopcu、S. R. Etesami 和 T. Başar。《通过八卦传播网络化智能体上的时变信息》。载于 ISIT，2022年6月。[24] P. Kaswan 和 S. Ulukus。《谣言传播年龄对篡改时间戳的易受影响性》。载于 IEEE ITW，2022年11月。[25] J. Hespanha。《随机混合系统的建模与分析》。《IEE控制理论与应用论文集》，《混合系统》特刊，153:520–535，2007年1月。