

带有编码冗余控制的机会网络数据转发机制

吴大鹏, 楼芃雯, 刘乔寿, 熊余, 王汝言

(重庆邮电大学 宽带泛在接入技术研究所, 重庆 400065)

摘要: 网络编码方法能有效提高数据传输效率, 但采用该方法的机会网络数据转发机制均未考虑传输中的编码数据迭代冗余副本控制问题。为了充分降低译码复杂度并减少迭代冗余副本, 提出了相遇持续时间估计及编码机会感知方法, 进而设计了带有传输容量估计的编码冗余控制数据转发机制。节点以分布式的方式感知数据编码机会, 为不同扩散程度的数据确定相应的优先级, 并根据传输容量估计结果完成转发控制, 以充分利用节点间的相遇持续时间。结果表明, 所提出机制有效地减少了网络中的冗余副本, 使译码所需的数据量趋于理论上的最小值, 改善了网络资源利用率。

关键词: 机会网络; 网络编码; 冗余控制; 传输容量

中图分类号: TP393

文献标识码: A

Coding redundancy controlled data forwarding mechanism in opportunistic networks

WU Da-peng, LOU Peng-wen, LIU Qiao-shou, XIONG Yu, WANG Ru-yan

(Broadband Ubiquitous Network Research Laboratory, Chongqing University of Posts and Telecom., Chongqing 400065, China)

Abstract: The network coding can improve network transmission efficiency. But most of the data forwarding mechanism does not consider the iteration redundancy control during data transmission. In order to reduce the complexity of decoding and iteration redundancy copies sufficiently in coding nodes limited network, the encounter duration estimating method and coding opportunity-aware method are proposed. Further, a data forwarding mechanism for coding redundancy control with communication capacity estimating method is designed. It can make full use of the encounter duration between nodes to forward the data, which has more coding opportunity and less than the community capacity. Furthermore, the coding number of nodes is updated and the iteration redundancy in every generation is controlled. Experimental results show that, the mechanism could reduce the iteration redundancy in the network and minimize data amount required by decoding, which tends to the theoretical minimum and improves the utilization of network resources.

Key words: opportunistic networks; network coding; redundancy control; transmission capacity

1 引言

移动自组织网络 (MANET, mobile ad hoc network) 使通信终端能够在无需网络基础设施支持的情况下组建临时通信网络^[1], 节点传输数据之前需要建立完整的端到端路径。但是, 对于动态性较

强的 MANET 来说, 多种因素均可能导致传输链路断裂, 使节点间持续的端到端连接无法保障, 从而频繁地触发路径重建操作, 消耗了大量网络资源^[2]。为了充分利用节点运动过程中带来的相遇机会, 研究人员提出了一种以“存储—携带—转发”方式承载数据的机会网络, 即节点临时存储来自其他节点

收稿日期: 2013-10-27; 修回日期: 2013-12-29

基金项目: 国家自然科学基金资助项目(61371097); 重庆市自然科学基金重点基金资助项目(CSTC2013JJB40001, CSTC2013JJB40006); 重庆市青年科技人才培养计划基金资助项目(CSTC2014KJRC-QNRC40001); 重庆邮电大学青年自然科学基金资助项目(A2012-93)

Foundation Items: The National Natural Science Foundation of China (61371097); The Natural Science Foundation of Chongqing (CSTC2013JJB40001, CSTC2013JJB40006); The Youth Talents Training Project of Chongqing Science & Technology Commission (CSTC2014KJRC-QNRC40001); The Foundation of Chongqing University of Posts and Telecommunication (A2012-93)

的数据,并携带多个节点的数据副本进行移动,直至其遇到更合适的转发节点或目的节点。此种网络架构以多节点协作的方式,更加灵活地完成了数据传输。可见,合理、高效的数据转发机制能够提高机会网络整体性能。

机会网络具有拓扑动态性、节点缓存和能量有限性、节点连接持续时间随机性等特征,数据转发机制需要综合考虑上述各种因素,以更加合理地选择转发节点并确定转发优先级。对于链路呈现出极强间断特性的机会网络来说,数据转发过程需要向网络中注入多个数据副本,以提高端到端的可靠性。目前的数据转发机制普遍采用带有副本控制的泛洪方式^[3-6],力求以最小化网络资源开销实现网络承载能力最大化。

1998年,香港中文大学的 Robert Li 等提出网络编码方法之后,研究人员发现在机会网络数据转发机制中引入网络编码^[7,8]方法能够在很大程度上降低连接间断特性对数据转发过程的影响,提高网络传输效率,实现了网络吞吐量最大化。但考虑到实现复杂度,应用于机会网络的数据转发机制中广泛采用随机线性网络编码方式,其核心思想是编码中继节点不断对所接收到的数据副本进行线性加权融合,直至目的节点能够恢复原始数据。此种方式将产生大量的冗余迭代副本。一方面,在不断融合迭代中,编码中继节点对其接收的数据并不具有选择性,即当数据到达中继节点后,则由中继节点执行下一次编码转发操作,同时,数据在网络中以泛洪的方式继续传输,当数据携带节点具有较高的活跃度时,编码次数将急剧增加,其副本数量将迅速超过目的节点译码所需数量,出现大量冗余副本;另一方面,机会网络中的数据转发过程并不需要建立源节点到目的节点的连接,端到端路径在时间域和空间域上存在较强的不连续性,目的节点无法采用显式的反馈信息确认源节点数据成功传输。当数据已到达目的节点并译码恢复后,网络中依然存在该数据的冗余副本。可见,上述2种情况均会导致网络中出现较多的冗余副本,继续以“存储—携带—转发”的方式传输冗余副本将极大地消耗有限的网络资源。

为了解决迭代冗余副本导致的网络性能下降问题,文献[9]提出了一种带有冗余控制的数据转发机制,根据预先设定的编码融合数据和原始数据副本比例,在编码融合数据满足译码要求之后,节点

将不再继续对数据执行编码操作,达到了控制迭代冗余副本的目的。文献[10]提出了节点连通度感知的编码数据转发机制(HubCode),各个节点以分布式的方式选择具有高连通度的节点作为编码中继节点,进而完成数据转发,从而达到降低网络开销的目的。利用数据扩散程度估计方法,文献[11]提出了相应的数据编码及转发机制,以达到更为合理利用网络资源的目的。文献[12]提出了一种应用于无线网状网的网络编码机会转发方法,其利用数据在备选转发节点和邻居节点间的冗余性,动态选择有编码机会的节点进行数据转发,从一定程度上提升了网络吞吐量。文献[13]提出了一种网络编码机会感知的转发机制,网络中的各个节点以动态的方式选择编码节点,并赋予编码后数据更高的优先级进行转发,从一定程度上提高了网络吞吐量。但是,上述机制未能实时管理传输过程中参与迭代编码的数据,数据将多次参与中继节点的编码融合过程,不仅大大增加了节点的译码复杂度,而且导致网络中出现大量冗余副本,极大地消耗了有限的节点缓存及网络资源,且随着网络运行时间的逐渐增加,数据迭代冗余副本数量也将随之上升,导致网络性能进一步恶化。

可见,迭代冗余副本的处理是基于网络编码的数据转发机制中亟待解决的关键问题。虽然,限制编码节点数量能够达到控制网络中迭代冗余副本的目的,但是机会网络中的数据转发过程采用多副本方式,分布式运行的节点无法获知数据在网络中的传输状态及扩散程度。可见,数据转发机制中的迭代冗余副本控制较为复杂,需要充分地保证目的节点成功译码并最小化编码融合数据数量。

提出一种带有编码冗余控制的机会网络数据转发机制(CRC),节点采用混合估计方法预测相遇持续时间,进而,根据当前的网络状态信息准确地量化编码融合数据的冗余程度,以最大化资源利用率为目标完成编码融合数据转发。所提出的机制能够在保障数据可靠传输的同时,对编码融合数据进行实时监控,达到提升网络资源利用率及最小化迭代冗余副本的目的。

2 传输容量估计方法

对于具有间断连接特性的机会网络来说,多个节点运动到彼此的通信范围内才能够建立临时

连接, 且节点间的连接持续时间具有不确定性, 若正在传输数据的链路中断, 则所转发的数据将出现不完整性。因此, 为了充分利用节点移动带来的通信机会, 需准确地估计节点相遇之后产生链路的传输容量, 继而更加合理地选择冗余度较低的数据副本进行转发, 达到提高网络资源利用率的目的。提出一种混合估计方法预测节点之间的相遇持续时间。

在实际机会网络中, 节点自身的运动呈现随机性、无规律性, 即其可以表示为随机游走模型或随机路点模型。在此种随机模型下, 节点的相遇间隔时间及相遇持续时间均呈现一定的规律性。大量文献对机会网络中节点的运动规律进行了详细分析^[14], 结论表明机会网络中节点的相遇间隔时间服从指数分布, 且相遇持续时间 $\{X(t), t \geq 0\}$, 也服从指数分布。从上述结果可知, 相遇持续时间 $\{X(t), t \geq 0\}$ 为广义平稳随机过程, 具体证明过程如附录1所示。

因此, 可利用节点运动过程中所获知的历史信息估计节点之间的相遇持续时间, 采用适用于广义平稳随机过程的时间序列指数平滑预测方法估计相遇持续时间。对于节点相遇持续时间序列 $\{X_i, i=1, 2, \dots, n\}$ 来说, 集合中任意 X_i 独立不相关, 因此, 可采用加权平均方法预测相遇持续时间, 如式(1)所示。

$$\hat{X}_{i+1} = S_i = \alpha X_i + (1-\alpha)S_{i-1} \quad (1)$$

其中, \hat{X}_{i+1} 表示节点相遇持续时间的估计值; X_i 为当前时刻相遇时间的实际值; S_i 、 S_{i-1} 分别表示当前时刻及前一时刻的指数平滑值, 且初始值 S_0 为原时间序列第一项, 即首次相遇时间; α 为平滑系数。

根据式(1), 节点可估计与其他节点的相遇持续时间。为了进一步提升估计结果的准确程度, 采用历史信息对估计误差进行处理。当估计值小于实际值的时候, 两者之间的差值依然为广义平稳随机过程, 则可以利用历史误差信息的均值对连接持续时间误差 τ_i 进行估计, 如式(2)和式(3)所示。

$$\tau_i = \frac{\sum_{k=1}^{i-1} f_k(t)}{\left\{X_k \geq \hat{X}_k, 1 \leq k \leq i-1\right\}} \quad (2)$$

$$f_k(t) = \begin{cases} X_k - \hat{X}_k, & X_k \geq \hat{X}_k \\ 0, & \text{其他} \end{cases} \quad (3)$$

其中, $f_k(t)$, $1 \leq k \leq i-1$ 为相遇时间实际值与估计值的差值, $\sum_{k=1}^{i-1} f_k(t)$ 表示超过估计值的部分相遇时间累加, $\left\{X_k \geq \hat{X}_k, 1 \leq k \leq i-1\right\}$ 表示估计值小于实际值的次数。

当估计值大于实际值时, 数据按照既定规则传输, 若连接中断, 对于未传输成功的数据副本直接丢弃且不对其更新相应的控制信息。

按照上述2种情况, 节点可利用历史信息采用指数平滑时间序列模型估计相遇持续时间。当估计结果小于实际相遇持续时间时, 节点采用条件均值方法进一步估计误差, 否则按照既定规则进行数据传输, 传输失败的数据不再更新控制信息, 直接等待下次传输机会。节点相遇持续时间预测方法如式(4)所示。

$$\hat{X}_{i+1} = \begin{cases} S_i, & \hat{X}_{i+1} \geq X_{i+1} \\ S_i + \tau_i, & \hat{X}_{i+1} < X_{i+1} \end{cases} \quad (4)$$

在对节点相遇持续时间完成预测之后, 进一步对节点间通信的传输容量进行估计。当两节点彼此进入各自的通信范围内, 表示节点间建立连接, 那么节点的相遇持续时间如式(5)所示, 其中 t_0 、 t_1 分别为节点相遇时间和离开时间。

$$X_i = t_1 - t_0 \quad (5)$$

同时, 节点间传输容量定义如下。

定义1 假设发送节点为 s , 接收节点为 r , s 单位时间内发送的数据大小为 b_s , r 单位时间内发送的数据大小为 b_r , 则给定节点间的传输容量为 b_s 与 b_r 中的较小值, 即

$$B_{sr} = \min(b_s, b_r) \quad (6)$$

因此, 节点 s 和 r 间的通信传输容量 C_{sr} 可按照式(7)所示方式表示。

$$C_{sr} = B_{sr} X_i \quad (7)$$

假设所有节点同构, 那么式(7)可简化为 $C_{sr} = B X_i$ 。

机会网络中节点移动所产生的通信链路持续时间有限, 充分利用有限的传输容量将能够显著改善网络性能。根据上述相遇持续时间预测结果, 节点可获知临时链路的传输容量, 进而, 优先转发冗余度较小的数据。

3 编码机会感知

在非编码多副本数据转发机制中,节点可根据相遇概率预测结果、原始数据副本重要程度等参数来控制数据副本数量,以达到降低数据副本数量的目的。然而,基于随机线性网络编码方法的数据转发机制中,原始数据以融合信息的形式进行传输。随着原始数据迭代次数的逐渐增加,网络中该数据的编码融合数据数量也随之上升,可见,编码次数是衡量原始数据副本在网络中扩散程度的标准。编码次数越大,表示原始数据副本在网络中的扩散程度越高。因此,节点可通过原始数据的编码次数来衡量编码融合数据对最终译码的影响程度,从而定量地描述副本的冗余程度。在数据转发过程中,节点可充分利用有限的传输容量转发冗余度较小的编码融合数据,增加未投递原始数据副本的编码机会,确保成功译码的同时有效地控制迭代冗余副本的传输。

如前所述,节点相遇时间间隔服从指数分布,则数据 i 未投递至目的节点的概率为 $\exp[-\lambda n_i(\Gamma - t)]$,其中 n_i 表示 t 时刻数据 i 的副本数量, Γ 为该数据 i 的生存时间 (TTL, time to live), $\Gamma - t$ 为数据 i 的剩余生存时间。然而,网络中传输的数据为编码融合数据,而非原始数据本身,因此无法按照上述方式估计编码融合数据的投递状态。如前所述,数据的编码次数直观地反映了其投递状态,进而,可通过该参数获知编码融合数据的冗余程度。为了能够充分地利用有限的传输容量,节点需要优先转发冗余度较低的编码数据。因此,对于时刻 t 下节点的缓存数据 i 来说,按式(8)所示量化其编码机会 $\{\xi_{CM_i}, i=1,2,\dots,i\}$,其中 L_i 为数据 i 的编码次数。

$$\xi_{CM_i} = \exp[-\lambda L_i(\Gamma - t)], i=1,2,\dots,i \quad (8)$$

可见,编码次数感知结果将直接影响编码机会。对于具有间断连接特性的机会网络来说,数据转发操作以多个节点协作的方式完成,编码次数感知过程需要以分布式的方式进行;此外,不同编码次数的数据对目的节点译码贡献不同,编码次数感知过程还需要区分编码融合数据的具体组成。

随机线性网络编码机制采用迭代的方式运行,若编码融合数据未直接投递到目的节点,则其将被重新注入网络中继续进行编码转发,可见,编码中继节点处参与编码的数据包含 2 类,分别为原始数

据副本及编码融合数据。因此,编码次数的感知过程也分以下 2 种情况:原始数据编码次数,编码中继节点接收到的数据为原始数据,则记为原始编码次数 l ;融合数据编码次数,编码中继节点接收到的数据为融合数据,记为融合编码次数 l' 。

根据机会网络数据转发的基本原理可知,数据在网络传输中不断被融合,当融合数据在编码中继节点处再次执行编码操作时,其与历史融合数据的线性相关性变大,其证明过程如附录 2 所示。由此可知,影响目的节点译码的因素主要包括数据的原始编码次数和融合编码次数,且原始编码次数对目的节点译码的影响程度较高。

针对多源多目的机会网络,为了进一步降低译码的计算复杂度,采用“代”的方式对数据进行编码,定义同一“代”数据为目的节点相同的原始数据,“代”的大小为 G ,表示所生成的原始数据数量。根据网络编码基本原理可知,对该“代”内的所有数据成功译码的充分必要条件是接收到至少 G 个线性无关的编码融合数据,因此,为了能够有效地控制冗余编码数据,定义了最大编码次数 $L_{\max} = G$,即数据的编码次数达到最大编码次数的时候,该数据副本不再参加编码中继节点的编码操作。

编码中继节点数据处理方法如式(9)所示。

$$F_{mj} = \sum_{m=n=1}^{m=k,n=i} \alpha_{mn} M_{mn}, \alpha_{mn} \in F_q \quad (9)$$

其中, F_{mn} 是目的节点均为 m 的编码融合数据, M_{mn} 是目的节点均为 m 的原始数据及编码融合数据, F_q 为有限伽罗华域, $q = 2^n$, α_{mn} 为伽罗华域中随机选取的编码系数; m 为目的节点 Id , $m=1,2,\dots,k$; n 为参与编码原始数据的 Id , $n=1,2,\dots,i$; j 为编码融合数据 Id , $j=1,2,\dots,j$ 。

由式(8)可知,当给定数据剩余生存时间时,数据的编码次数越多,其编码机会越小。如前所述,原始数据直接编码对译码成功率的影响较大,因此,采用线性累加方式感知原始数据编码次数 l ,即 $l=l+1$;此外,融合数据多次参与编码过程之后所得到的数据具有较强的线性相关性,且融合数据的成分直接影响目的节点译码成功率,因此,考虑到编码融合数据所融合的任何原始数据对译码增益的影响相同,采用均值方法感知融合数据编码机会,即 $l'=l_0/n$,其中 n 为融合的原始数据数量。

对于基于网络编码的数据转发机制来说,目的

节点根据成功接收的原始数据和全部编码融合数据完成数据恢复，其中编码融合数据经过多个编码中继节点多次融合操作，其线性相关性较大，导致该编码融合数据中所包含的原始数据副本对译码的影响也随之减小，因此对这类再编码次数做均值处理。综上，分类统计所有原始数据的编码次数 L ，如式(10)所示，若数据的编码次数已超过最大编码次数，表明网络中该数据的编码融合数据已足够译码，即当 $L \geq L_{\max} = G$ 时，编码机会置 0。

$$L = \lceil l + l' \rceil \quad (10)$$

图 1 所示为编码机会感知的数据转发机制在多源单目的机会网络中的具体执行过程，其中节点 a 、 b 、 c 为编码中继节点， S 、 R 、 P 、 T 为源节点， D 为目的节点； S 、 R 、 P 、 T 分别携带原始数据 $\{x_1, x_2, x_3, x_4\}$ 。 t_1 时刻， a 接收到 S 和 R 转发的原始数据副本 x_1, x_2 ，若 a 尚有缓存，将其直接存储，并发送数据概要向量； t_2 时刻， a 与 b 相遇， a 将其自身携带数据编码后转发 F_1 ，同时转发编码系数并记录 x_1, x_2 的编码次数； $t_3 \sim t_7$ 时刻节点操作均同上，但在 t_8 时刻，节点选择编码机会较大的数据优先编码融合并转发，而不是直接将缓存上的所有数据编码； t_9 时刻， D 接收到 4 个线性无关的编码融合数据时成功译码。各转发时刻数据的编码次数如图 2 所示，图 2 中各时刻数据的编码次数按升序排列，由于按照前述方法，数据的编码次数与该数据在下一时刻的编码机会会有直接关系，即编码次数越多，说明该数据在网络中的扩散程度越高，则数据的编码机会越小。因此，可以根据图 2 中数据编码次数的多少，确定下一时刻与该数据对应的转发优先级。

按照上述方法，在多源多目的机会网络中，通过编码机会感知方法，编码中继节点将网络中的原始数据按照当前网络状态及数据副本的扩散状态动态地确定数据的编码机会，使迭代冗余副本得到合理控制，更加充分地利用了节点运动过程所产生的临时链路资源。

4 编码冗余控制数据转发机制

按照所提出的相遇持续时间及编码机会估计方法，节点可根据当前网络状态优先转发冗余度较低的数据，以达到保障目的节点成功译码的前提下最大程度减少传输数据冗余的目的。显然，设计合

理、有效的数据转发机制至关重要。

根据网络中的节点类型，数据转发过程主要包含以下 3 种情况，分别为普通中继节点之间、普通中继节点与编码中继节点之间及编码中继节点之间。具体的转发步骤如下。

1) 初始化过程：为不同类型节点设定相应的标志位，其中 $Flag_i=0$ 为普通中继节点； $Flag_i=1$ 为编码中继节点；网络中各个节点以分布式的方式记录与其他节点的首次相遇时间 t_0 和离开时间 t_1 ，并获知首次相遇持续时间 X_0 。

2) 相遇持续时间估计：根据首次相遇持续时间 X_1 ，节点建立相遇持续时间序列，按照所提出的方法估计下一时刻相遇持续时间 X_{i+1} ，并在此基础上估计传输容量。

3) 数据转发过程：数据到达编码中继节点之后，节点根据与其相遇的节点类型确定相应的转发操作。

① 若 2 个普通中继节点相遇，即 $Flag_i=Flag_j=0$ ，且节点 i 缓存中数据的目的节点为 j ，即 $M_{i-id}=j_{id}$ ，则节点直接进行数据转发，同时将 M_{i-id} 的编码次数 L 置为 $L=G$ ；反之，则携带该原始数据继续移动，直到遇到编码中继节点。

② 若普通中继节点与编码中继节点相遇 $Flag_i=0$ 且 $Flag_j=1$ ，则直接转发未编码数据 M_i ；若编码中继节点缓存仍有剩余，直接存储该原始数据；否则，利用式(9)将该数据 M_i 与自身缓存中目的节点一致的数据编码后存储，并更新相应编码次数 L 等控制信息。

③ 若两节点均为编码中继节点 $Flag_i=Flag_j=1$ ，则根据概要向量的比较结果，确定需要传输的数据集合 θ_i ，并与传输容量 C 的估计结果进行比较，若 $C \geq \theta_i$ ，处于发送端的编码中继节点依次按式(9)对缓存中数据进行编码转发，并分别记录更新数据的编码次数 L 等控制信息，否则，分别计算缓存数据的编码机会 ξ_{CM_n} ，按照传输容量及编码机会依次按照式(9)编码转发数据。

4) 相遇持续时间估计结果修正：若 t 时刻估计的相遇持续时间小于 $t+1$ 时刻实际测量值，即存在剩余链路持续时间 τ_i ，则利用式(2)对 τ_i 进行预测，并继续转发缓存中编码机会较大的数据并更新相应控制信息。

5) 重复以上 4 步，直至全部数据完成投递。

所提出的数据转发过程如图 3 所示。

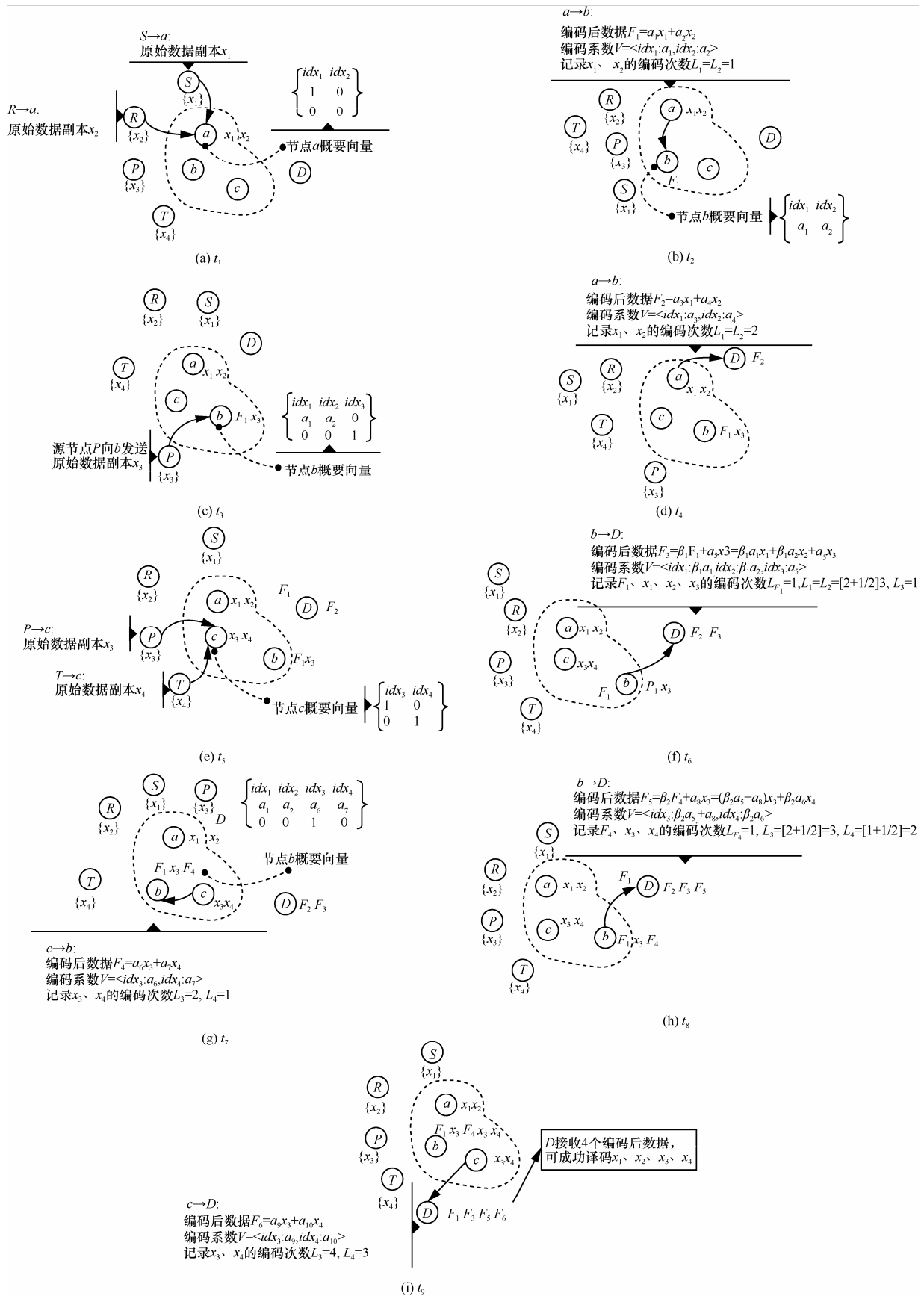


图 1 编码机会感知方法

t_1	数据	x_4	x_3	x_2	x_1					
	编码次数	0	0	0	0					
t_2	数据	F_1	x_4	x_3	x_2	x_1				
	编码次数	0	0	0	1	1				
t_3	数据	F_1	x_4	x_3	x_2	x_1				
	编码次数	0	0	0	1	1				
t_4	数据	F_2	F_1	x_4	x_3	x_2	x_1			
	编码次数	0	0	0	0	2	2			
t_5	数据	F_2	F_1	x_4	x_3	x_2	x_1			
	编码次数	0	0	0	0	2	2			
t_6	数据	F_3	F_2	x_4	F_1	x_3	x_2	x_1		
	编码次数	0	0	0	1	1	3	3		
t_7	数据	F_4	F_3	F_2	F_1	x_4	x_3	x_2	x_1	
	编码次数	0	0	0	1	1	2	3	3	
t_8	数据	F_5	F_2	F_3	F_4	F_1	x_4	x_3	x_2	x_1
	编码次数	0	0	0	1	1	2	3	3	3

图 2 各时刻待转发数据编码次数统计

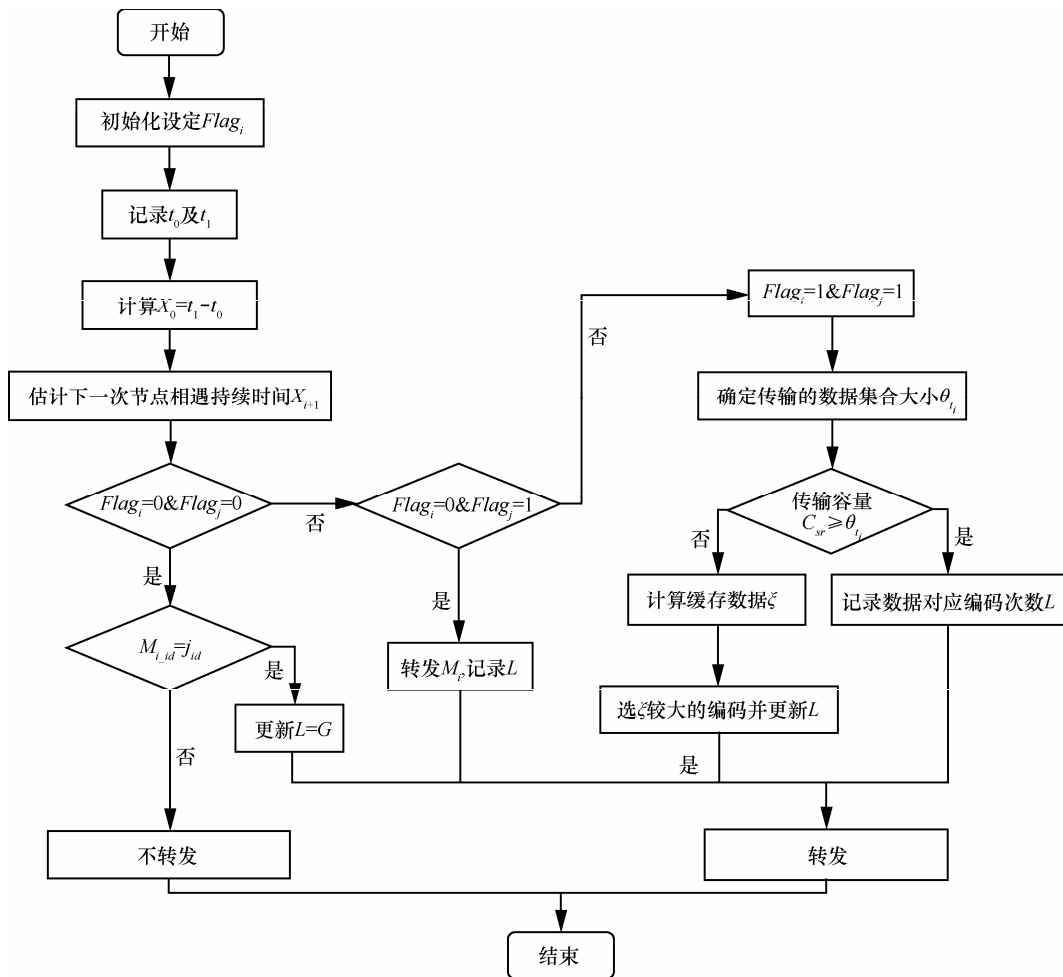


图 3 编码冗余控制数据转发流程

5 数值结果分析

采用机会网络(ONE, opportunistic network environment)仿真平台^[15]对所提出机制的性能进行验证,并分别与其他2种典型数据转发机制进行比较,其中包括文献[16]中所提出的BubbleRap机制和文献[9]中所提出的HubCode机制。BubbleRap机制充分地考虑了节点社会属性,利用本地等级及全局等级较高的节点分别完成社区内和社区间数据转发,最终以分布式的方式实现数据投递;在编码节点受限的基础上,HubCode机制选取高连通度节点作为编码节点,数据转发操作限制在编码节点之间执行。

在所提机制中,由于采用随机线性网络编码方法,在译码时,节点需要根据编码后数据及相应的编码系数矩阵完成译码,译码复杂度主要与副本的冗余程度直接相关,也就是说,若节点接收到较多的迭代冗余副本,那么编码系数矩阵的维数也将随之增加,译码的过程将会复杂化,即译码复杂度增加。提出的机制根据数据的编码次数及传输状态合理地设定编码机会,则冗余度较高的数据编码机会较小,同时由于节点之间的连接时间有限,且数据生存时间有限,那么,随着转发数据生存时间逐渐减少,则冗余度较高的数据将无法得到投递机会,充分降低了译码复杂度的同时有效地减少了迭代冗余副本。

为了验证所提出机制的适应性,在不同网络负载状态及不同节点缓存能力下对上述3种机制进行了分析比较,主要性能指标包括数据成功投递率、平均时延和网络负载率3方面。网络负载率定义为数据的冗余投递次数与成功投递的消息数之间的比值关系,如式(11)所示,其中 OR 为网络负载率, N_r 为数据转发的总次数, n_s 为成功投递到目标节点的数据数量。该参数反映了数据转发机制的运行效率,即转发效率。可见,数据投递过程中的转发次数越多,则网络负载率越大,表明数据转发机制的运行效率越低,即数据转发机制的转发效率越低。

$$OR = (N_r - n_s) / n_s \quad (11)$$

按照如前所述的编码冗余控制数据转发机制,在数据转发过程的初始阶段,对不同类型的节点设定相应的标志位,记录首次相遇时间,随着数据转发过程的不断持续,节点根据记录运动过程中所获知的历史信息估计节点之间的相遇持续时间,获知的历史信息越多,预测将会逐渐准确。进而,可以根据节点间相

遇持续时间的预测结果完成传输容量的估计,并按照所提出机制的3种情况进行分类转发。在传输容量有限的情况下,当两节点均为编码节点时,为避免出现大量的迭代冗余副本,根据前述的编码机会感知方法,优先转发编码机会较大的数据,并及时更新数据编码次数,进而实现了对编码迭代冗余副本的控制。

具体仿真场景的参数设置如表1所示。

表1 仿真参数设置

参数设定	参数数值
网络区域	4 500 m × 3 400 m
网络仿真时间/h	12
节点通信方式	Bluetooth
数据分组间隔时间/s	25~35
节点缓存大小/MB	5
节点带宽/(kbit·s ⁻¹)	250
节点总数	128
社区数	4
社区编码节点数	9
平滑系数	0.15
数据生存时间/m	300

5.1 相遇持续时间预测准确度

显然,节点相遇持续时间估计方法的准确性直接影响数据转发机制的性能。给出了节点在仿真运行时间内相遇持续时间的估计值及对应该时刻下实测值,如图4所示,并统计了机会网络随机移动模型下各个节点连接持续时间的实际值及估计值,同时对节点相遇持续时间的误差给出了累积概率分布,如图5所示。结果表明,82.3%的相遇持续时间估计误差小于7%,最大相遇持续时间估计误差小于15%,整体平均误差为10%。可见,所提出的相遇持续时间估计方法较为准确。

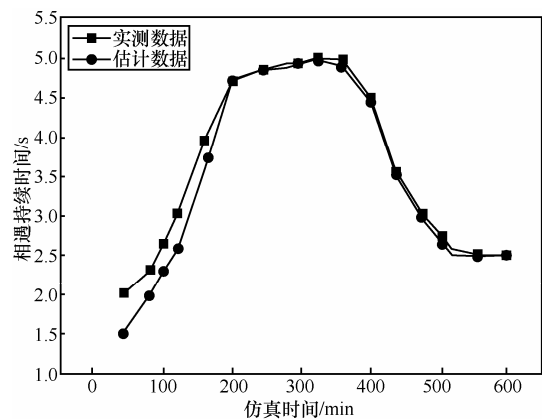


图4 节点相遇持续时间实测数据与估计数据

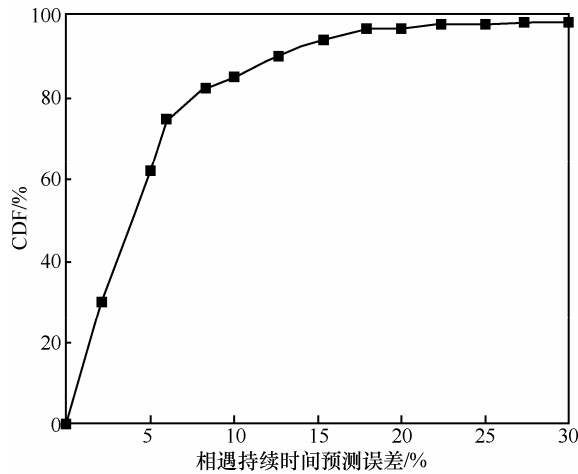


图 5 节点相遇持续时间准确性验证

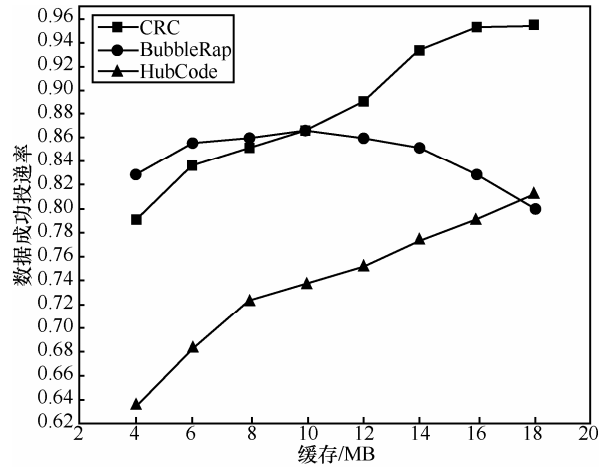


图 6 不同缓存下的数据成功投递率

5.2 不同缓存容量下的网络性能

不同缓存容量下的网络性能如图 5~图 7 所示，主要包括数据成功投递率、网络负载率和平均时延。

数据成功投递率直观地反映了数据传输的可靠性及网络承载能力，图 6 表明了不同缓存容量下，3 种机制的数据成功投递率随缓存增加呈上升趋势，其主要原因在于节点除了需要存储待传输数据外，还需要存储大量的控制信息，如编码系数矩阵等，当缓存较为充足的时候，控制信息与数据的资源竞争较小。与所提出的机制类似，BubbleRap 中的节点需要感知并存储相应的本地等级、全局等级及其他相关信息，这些信息的更新和维护均需要节点提供额外的空间，因此，3 种机制的数据成功投递率均随缓存的增加而提高。但是在缓存达到 10 MB 时，BubbleRap 机制的性能有所下降，其主要原因在于随着缓存容量的逐渐增加，节点存储的待转发数据数量也随之上升，但由于节点资源有限，等级较高的节点承担了大部分数据的转发，导致数据拥塞，成功投递率下降。虽然等级较低的节点缓存容量也逐渐增加，但其大部分时间处于闲置状态，使节点缓存无法得到充分利用。所提出的 CRC 机制预先估计节点间的传输容量，根据传输容量的大小选择编码机会较大的数据进行转发，确保冗余度较低的编码融合数据成功接收并译码，能够根据当前网络状态及数据传输状态更加合理地利用有限的网络资源。而 HubCode 机制仅通过降低编码节点数量来达到控制冗余传输的目的，并未考虑编码数据与目的节点译码的关系。数值结果表明，CRC 的成功投递率比 BubbleRap 和 HubCode 2 种机制分别提升了 11.5% 和 18.8%。

图 7 给出了各机制在不同缓存容量下的网络负载率。从结果可知，HubCode 与 CRC 机制的网络负载率均随着缓存的增加而不断降低，而 BubbleRap 机制的负载率随着缓存的增加呈现出不断上升的趋势。CRC 的平均负载率始终保持在 115~130 之间，与 BubbleRap 和 HubCode 2 种机制相比，其负载率分别降低了 43.8% 和 18.7%。其主要原因在于，随着缓存容量的不断上升，BubbleRap 中等级较高的节点将接收并存储更多的数据，即数据转发次数也大幅增加，但是由于大部分数据转发都依托等级较高的节点来完成，使其以较大概率产生拥塞，导致数据的投递机会和成功投递数量下降，因此，BubbleRap 机制的负载率随着缓存容量的上升而逐渐升高。HubCode 在编码节点选取完毕后，通过受限的编码节点对接收到的数据执行编码转发，随着缓存容量的不断上升，相应的控制信息就获得了较充足的存储空间，利于数据的成功投递。

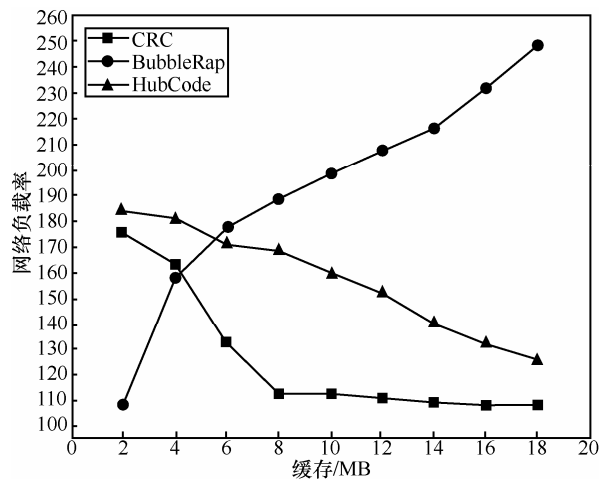


图 7 不同缓存下的网络负载率

同样, CRC 机制也获得了足够的信息存储空间, 并且所提出的编码机会感知方法及传输容量估计方法能够对数据的转发状态进行区分, 进一步控制迭代冗余副本, 限制了不必要的数据转发次数, 因此, 网络负载率较低。

各机制在不同缓存容量下的平均时延如图 8 所示。从图中可以看出, CRC 的平均延时较低, HubCode 平均延时最高, BubbleRap 居中。其主要原因在于, BubbleRap 机制利用社区内和社区外中心度等级对数据进行选择性投递, 使其转发更具有针对性, 降低了数据传输时延。而 HubCode 选择连通度较高的节点作为编码节点, 虽然能够一定程度上降低部分数据的转发次数, 但是数据需要在编码节点处经过较长时间的等待, 使延时上升。由于在实际机会网络中, 节点通过移动带来的相遇机会进行数据转发, 节点之间的连接时间有限, 通过传输容量估计及编码机会感知方法, CRC 完成待编码数据的选择, 进一步限制了编码后数据冗余副本的增加, 选择了更需要转发的数据, 即编码机会较大的数据进行转发, 同时利用编码融合, 缩短了数据的传输时延, 带来了性能增益。结果表明, CRC 机制的平均传输延时比 BubbleRap 降低了 38.1%, 比 HubCode 降低了 67.5%。

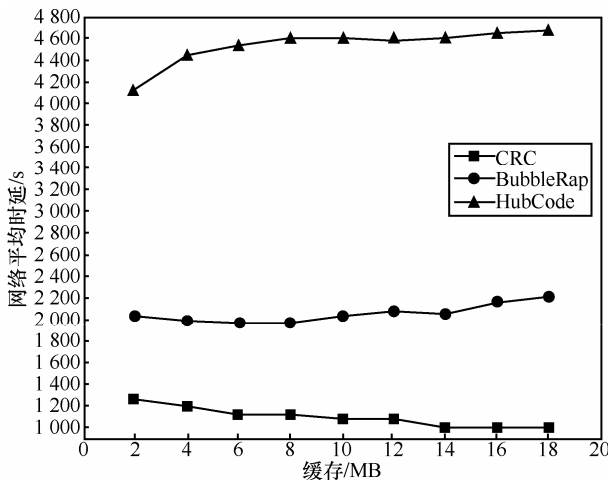


图 8 不同缓存下的网络平均时延

5.3 不同网络负载下的网络性能

图 9~图 11 给出了不同网络负载下的网络性能。数据生成时间间隔能够直观地反映出网络负载状态。数据生成时间间隔越短, 表明给定时间内网络内的数据总量越高, 在节点缓存有限及传输资源受限的情况下, 极易造成网络拥塞, 最终导致数据转

发机制出现较差的性能表现。从图 8 中可知, 利用编码机会感知方法, CRC 机制限制了编码次数较多的数据副本在网络中继续扩散, 同时增加了冗余度较小的数据转发机会, 更加有效地利用了有限的网络资源, 因此, 其在数据总量较大时依然能保持较高的数据成功投递率。BubbleRap 的数据投递率随着数据生成时间间隔的增加而提高, 由于其属于非编码机制, 无需存储除数据副本之外的控制信息, 同时其转发时充分地考虑了节点的社会属性, 中继节点的选择更加有针对性, 因此, 其性能优于 HubCode 机制。HubCode 机制中仅采用部分节点作为编码节点, 在数据量快速上升的时候, 大量的数据及控制信息集中在编码节点上, 将会造成编码节点无法接收其他节点的数据并进行编码转发, 因此, 其数据成功投递率较低。结果表明, CRC 机制的数据成功投递率较 BubbleRap 机制提高了 16.2%, 比 HubCode 机制提高了 42.4%。

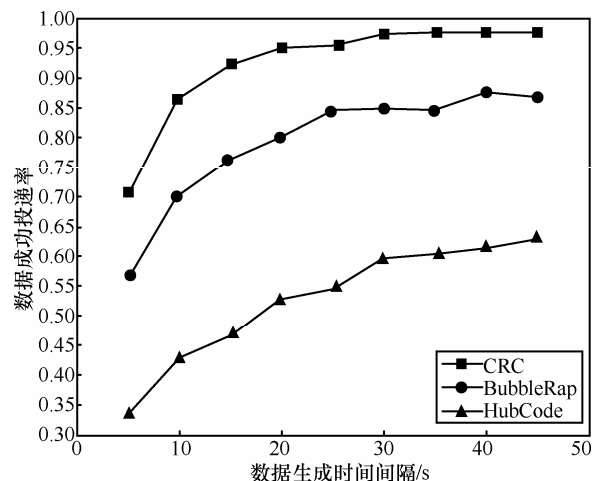


图 9 不同数据生成时间间隔下的成功投递率

不同数据生成时间间隔下的网络负载率如图 10 所示。随着数据生成时间间隔的增加, 网络中的数据总量逐渐减少, 因此数据的转发次数与成功投递的数据数量均呈下降趋势, 但由于转发次数下降的趋势较小, 导致了网络负载率随数据生成时间间隔的增加而上升。在数据生成时间间隔大于 15 s 后, CRC 机制的负载率逐渐低于 HubCode 机制。原因在于 CRC 机制较好地利用了每次节点间的相遇持续时间, 并依据相遇持续时间估计结果, 同时结合编码机会感知, 降低了全网迭代冗余副本, 合理地分配了网络资源, 降低了译码复杂度, 从而保证了较低的网络负载率。

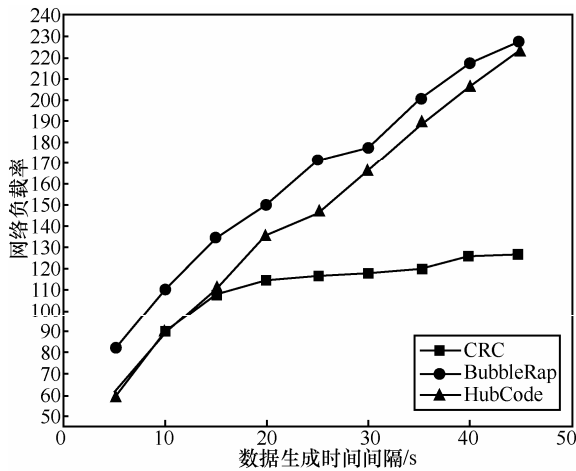


图 10 不同数据生成时间间隔下的网络负载率

从图 11 所示的结果可知，随着数据生成时间间隔的增加，各机制网络平均时延均迅速降低。在数据生成时间间隔较短时，网络数据总量较大，同时由于节点缓存有限，造成其缓存溢出，导致所携带数据的转发机会降低，进而，使数据的平均时延较高。随着全网数据量的减少，各节点的缓存得到释放，数据的转发机会得到了增加，因此网络平均时延逐渐降低。CRC 机制从需转发数据的优先级以及编码机会量化 2 方面出发，其既增加了数据转发的针对性，减少数据丢弃，同时又使迭代冗余副本降低，网络资源得到合理利用，进一步缩短数据传输时延。

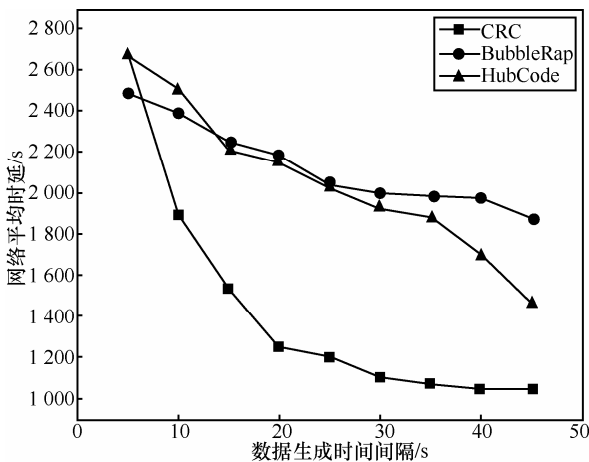


图 11 不同数据生成时间间隔下的网络平均时延

6 结束语

为了充分降低译码计算复杂度及编码数据冗余，提出了相遇持续时间估计及编码机会量化方法，进而设计了带有传输容量估计的编码冗余控制

数据转发机制。充分利用节点间的相遇持续时间，根据当前的网络状态，优先转发编码机会较大的数据，并及时更新数据的编码次数，进而分批对编码迭代冗余副本进行控制。与传统的机制相比，所提出的机制在确保成功译码的同时，最大程度地减少了数据转发过程中的编码融合数据冗余，使译码所需的数据量趋于理论上的最小值。同时，所提出的机制有效地提高了数据传输过程中的可靠性，改善了网络资源利用率。

附录 1 相遇持续时间证明

相遇持续时间 $\{X(t), t \geq 0\}$ 如式(12)所示。

$$X(t) = \begin{cases} 1 - e^{-\lambda t}, & t \geq 0 \\ 0, & t < 0 \end{cases} \quad (12)$$

可获知节点相遇持续时间的数学期望，如式(13)所示。

$$E[X(t)] = 1/\lambda \quad (13)$$

继而，可得相遇持续时间的自相关函数如式(14)和式(15)所示。

当 $\tau = 0$ 时， τ 为任意时间的取整常数，则

$$R_X(t, t - \tau) = E[X(t)X(t - \tau)] = 1/\lambda^2 \quad (14)$$

当 $\tau \neq 0$ 时，

$$R_X(t, t - \tau) = E[X(t)X(t - \tau)] = \frac{1}{\lambda e^{\lambda \tau}} - \tau e^{\lambda \tau} \quad (15)$$

由此可知，相遇持续时间 $X(t)$ 满足广义平稳随机过程的定义，即其数学期望为常数，自相关函数仅与时间间隔 τ 有关。

附录 2 数据相关性证明

假设当前时刻下，网络中存在 2 种编码融合数据，分别为原始数据的编码融合数据和迭代冗余副本，如式(16)和式(17)所示，其中 F_i 及 F_i^1 为编码融合数据， β_i 及 α_{ij} 为所选取的编码系数， x_i 为原始数据副本。

$$\begin{bmatrix} F_1 \\ F_2 \\ \vdots \\ F_i \end{bmatrix} = \begin{bmatrix} \alpha_{11} & \alpha_{12} & \cdots & \alpha_{1i} \\ \alpha_{21} & \alpha_{22} & \cdots & \alpha_{2i} \\ \vdots & \vdots & \vdots & \vdots \\ \alpha_{i1} & \alpha_{i2} & \cdots & \alpha_{ii} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_i \end{bmatrix} \quad (16)$$

$$\begin{bmatrix} F_1^1 \\ F_2^1 \\ \vdots \\ F_i^1 \end{bmatrix} = \begin{bmatrix} \beta_1 F_1 \\ \beta_2 F_2 \\ \vdots \\ \beta_i F_i \end{bmatrix} = \begin{bmatrix} \beta_1 \alpha_{11} & \beta_1 \alpha_{12} & \cdots & \beta_1 \alpha_{1i} \\ \beta_2 \alpha_{21} & \beta_2 \alpha_{22} & \cdots & \beta_2 \alpha_{2i} \\ \vdots & \vdots & \vdots & \vdots \\ \beta_i \alpha_{i1} & \beta_i \alpha_{i2} & \cdots & \beta_i \alpha_{ii} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_i \end{bmatrix} \quad (17)$$

对于目的节点来说，若其接收到 i 组线性无关的编码融

合数据, 则其编码系数矩阵满秩, 则有且只有唯一一组解恢复原始数据, 目的节点能够成功译码; 但是, 在实际网络中, 译码时通常会收到原始数据的编码融合数据和迭代冗余副本 2 种编码数据。当接收到的 i 组数据如式 (18) 时, 编码系数矩阵出现了线性相关组, 矩阵不满秩, 根据非齐次线性方程组的性质可知, 该方程无解或有无穷解, 即式(18)中增广矩阵的秩小于 i , 无法恢复原始数据。因此, 可得结论, 编码融合数据再融合时与前一时刻的融合数据线性相关性变大。

$$\begin{bmatrix} F_1^1 \\ F_1 \\ \vdots \\ F_{i-1} \end{bmatrix} = \begin{bmatrix} \beta_1 \alpha_{11} & \beta_1 \alpha_{12} & \cdots & \beta_1 \alpha_{1j} \\ \alpha_{11} & \alpha_{12} & \cdots & \alpha_{1j} \\ \vdots & \vdots & \vdots & \vdots \\ \alpha_{(i-1)1} & \alpha_{(i-1)2} & \cdots & \alpha_{(i-1)j} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_i \end{bmatrix} \quad (18)$$

参考文献:

- [1] KHABBAZ M J, ASSI C M, FAWAZ W F. Disruption-tolerant networking: a comprehensive survey on recent developments and persisting challenges[J]. IEEE Communications Surveys and Tutorials, 2012, 14(2): 607-640.
- [2] BULUT E, GEYIK S, SZYMANSKI B. Efficient routing in delay tolerant networks with correlated node mobility[A]. The 7th IEEE International Conference on Mobile Ad-Hoc and Sensor Systems[C]. San Francisco, USA, 2010.79-88.
- [3] DALY E M, HAAHR M. The challenges of disconnected delay-tolerant MANET[J]. Ad Hoc Networks, 2010, 8(2):241-250.
- [4] GAO W, LI Q H, ZHAO B, *et al.* Social-aware multicast in disruption-tolerant networks[J]. IEEE/ACM Transactions on Networking, 2012, 20(5): 1553-1566.
- [5] CONAN V, COMMUN T, CLARKSBURG M D, *et al.* Fixed point opportunistic routing in delay tolerant networks[J]. IEEE Journal on Selected Areas in Communications, 2008, 26(5):773-782.
- [6] PAN D R, LIN M, CHEN L J, *et al.* An improved spray and wait with probability choice routing for opportunistic networks[J]. Journal of Networks, 2012, 7(9): 1486-1492.
- [7] HAN X, YANG Y W, WANG L. Performance of network coding protocol based epidemic routing[J]. Journal of Computer Applications, 2012, 32(3): 791-794,799.
- [8] 樊凯, 李令雄, 龙冬阳. 无线 Mesh 网中网络编码感知的按需无线路由协议的研究[J]. 通信学报, 2009, 30(1): 128-134.
FAN K, LI L X, LONG D Y. Study of on-demand cope-aware routing protocol in wireless Mesh networks[J]. Journal on Communications, 2009, 30(1): 128-134.
- [9] SONG Z M, SU J S, PENG W, *et al.* NTC: towards efficient network coding in delay tolerant networks[A]. The Fifth International Conference Innovative Mobile and Internet Services in Ubiquitous Computing[C]. Seoul, Korea, 2011. 57-64.
- [10] SHABBIR A, SALIL S K. HUBCODE: hub-based forwarding using network coding in delay tolerant networks[J]. Wireless Communications and Mobile Computing, 2013, 13(9): 828-846.
- [11] NARMAWALA Z, SRIVASTAVA S. MIDTONE: multicast in delay tolerant networks[A]. The Fourth International Conference on Communications and Networking[C]. Xian, China, 2009. 1-8.
- [12] ZHANG J, CHEN Y P, MARSIC I. Network coding via opportunistic forwarding in wireless mesh networks[A]. Wireless Communications and Networking Conference[C]. 2008. 1775-1780.
- [13] 袁永琼, 张军, 王赞. 无线多跳网络中网络编码感知的机会转发机制[J]. 北京航空航天大学学报, 2012, 38(5): 648-653.
YUAN Y Q, ZHANG J, WANG Y. Network coding-aware opportunistic forwarding mechanism in wireless multi-hop networks[J]. Journal of Beijing University of Aeronautics and Astronautics, 2012, 38(5): 648-653.
- [14] HAN B, PAN H, KUMAR V S, *et al.* Mobile data offloading through opportunistic communications and social participation[J]. IEEE Transactions on Mobile Computing, 2012, 11(5): 821-834.
- [15] KERÄNEN A, OTT J, KÄRKKÄINEN T. The ONE simulator for DTN protocol evaluation[A]. Proceedings of 2nd International Conference on Simulation Tools and Techniques[C]. Rome, Italy, 2009. 1-10.
- [16] PAN H, CROWCROFT J, YONEKI E. BUBBLE rap: social-based forwarding in delay-tolerant networks[J]. IEEE Transactions on Mobile Computing, 2011, 10(11): 1578-1589.

作者简介:



吴大鹏 (1979-), 男, 黑龙江大庆人, 重庆邮电大学教授、硕士生导师, 主要研究方向为泛在无线网络、无线网络服务质量管理。



楼芃雯 (1988-), 女, 河南南阳人, 重庆邮电大学硕士生, 主要研究方向为容迟网络、网络编码。



刘乔寿 (1979-), 男, 云南曲靖人, 重庆邮电大学副教授, 主要研究方向为泛在无线网络。

熊余 (1982-), 男, 四川内江人, 重庆邮电大学副研究员, 主要研究方向为互联网服务质量控制。

王汝言 (1969-), 男, 湖北浠水人, 重庆邮电大学教授、博士生导师, 主要研究方向为泛在网络、多媒体信息处理等。