

(12) 发明专利申请

(10) 申请公布号 CN 103037391 A

(43) 申请公布日 2013. 04. 10

(21) 申请号 201310017700. 7

(22) 申请日 2013. 01. 17

(71) 申请人 上海交通大学

地址 200240 上海市闵行区东川路 800 号

(72) 发明人 薛广涛 余飞 朱弘恣 俞嘉地

(74) 专利代理机构 上海交达专利事务所 31201

代理人 王毓理 王锡麟

(51) Int. Cl.

H04W 16/22(2009. 01)

H04W 52/02(2009. 01)

H04W 76/04(2009. 01)

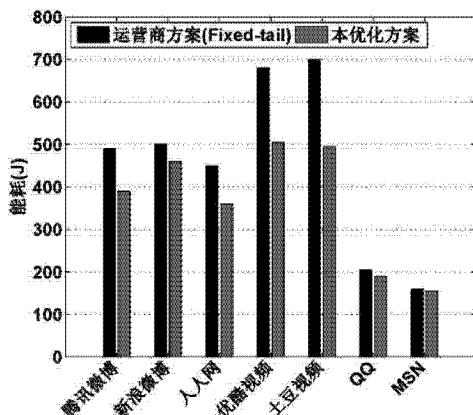
权利要求书 2 页 说明书 6 页 附图 7 页

(54) 发明名称

基于数据流预测的低功耗 RRC 协议优化控制方法

(57) 摘要

一种无线通信技术领域的基于数据流预测的低功耗 RRC 协议优化控制方法, 利用自回归滑动平均模型 (ARMA) 捕捉 3G 数据流中数据块块间间隔时间 (Inter-Block Time, IBT) 序列并得出 IBT 序列的时间相关性, 然后通过对时间相关性进行基于移动终端的能量转换的去尾处理以及预测误差修正处理, 最终实现降低功耗的 RRC 协议优化机制。本发明能够在保证用户体验度的前提下, 显著的较少移动终端的能量消耗。本发明通过时间序列分析模型捕捉 3G 数据流中的时间相关性, 并建立预测模型。利用未来数据流的预测值, 动态地调整 RRC 状态机的转换策略, 从而减少额外能量的消耗, 同时在用户体验度和能源效率之间做出了较好的权衡。



1. 一种基于数据流预测的低功耗 RRC 协议优化控制方法, 其特征在于, 利用自回归滑动平均模型捕捉 3G 数据流中 IBT 序列并得出 IBT 序列的时间相关性, 然后通过对时间相关性进行基于移动终端的能耗转换的去尾处理以及预测误差修正处理, 最终实现降低功耗的 RRC 协议优化机制。

2. 根据权利要求 1 所述的方法, 其特征是, 所述的 IBT 序列的时间相关性通过以下方式获得: 收集移动终端上收发的数据包, 并抽象成数据块序列, 然后从中提取出块间间隔 IBT 序列; 再利用自回归滑动平均模型捕捉 IBT 序列中的时间相关性建立用于对未来 IBT 的值进行预测的预测模型;

所述的 ARMA 预测模型针对不同的应用场景进行不同的对应训练;

所述的应用场景包括: 即时通讯、流媒体播放、网页浏览。

3. 根据权利要求 1 或 2 所述的方法, 其特征是, 所述的自回归滑动平均模型通过以下步骤得到:

1. 1) 采集终端收发的数据包, 记录下包头信息, 并通过深度包分析技术将数据包按照应用场景进行分类;

1. 2) 对每种应用场景下的数据包序列进行预处理, 将数据包序列转换成数据块序列, 间隔小于 500 毫秒的相邻数据包会被认为属于同一个数据块, 得到数据块序列后, 从中提取块间间隔时间, 形成 IBT 序列 $\{t_i, i = 1..n-1\}$;

1. 3) 采用自回归滑动平均模型对 IBT 序列进行建模, 设模型为 $t_i = \Phi_0 + \Phi_1 t_{i-1} + \Phi_2 t_{i-2} + \dots + \Phi_p t_{i-p} - \varepsilon_i - \theta_1 \varepsilon_{i-1} - \theta_2 \varepsilon_{i-2} - \dots - \theta_q \varepsilon_{i-q}$, 其中: $\Phi_0, \Phi_1, \dots, \Phi_p, \theta_1, \dots, \theta_q$ 为待定系数, $\varepsilon_i, \dots, \varepsilon_{i-q}$ 为误差分量, 模型的训练过程穷举参数 p 和 q 的取值空间, 对每一组取值, 拟合出待定系数 $\Phi_0, \Phi_1, \dots, \Phi_p, \theta_1, \dots, \theta_q$, 并计算在该系数下的拟合残差的方差 σ^2 , 再计算 $AIC = n \log \sigma^2 + (p+q+1) \log n$ 的值, 最终选取使最小信息准则 AIC 值最大的一组 p 和 q 作为 ARMA 模型参数;

1. 4) 采用自回归滑动平均模型对预测误差序列进行建模: 预测误差序列为 $\{e_i = t_i - t'_{i-1}, i = 1..n-1\}$, e_i 表示第 i 个 IBT 值的预测误差,

$e_i = \Phi_0 + \Phi_1 e_{i-1} + \Phi_2 e_{i-2} + \dots + \Phi_p e_{i-p} - \varepsilon_i - \theta_1 \varepsilon_{i-1} - \theta_2 \varepsilon_{i-2} - \dots - \theta_q \varepsilon_{i-q}$, 其中: $\Phi_0, \Phi_1, \dots, \Phi_p, \theta_1, \dots, \theta_q$ 为待定系数, $\varepsilon_i, \dots, \varepsilon_{i-q}$ 为误差分量, 自回归阶数 p 和滑动平均阶数 q 同样可根据最小信息准则测试确定。

4. 根据权利要求 1 所述的方法, 其特征是, 所述的去尾处理是指: 为了减少能耗并在后续数据传输发生时用户体验不到任何提升时延, 若后续 IBT 预测值大于提升时延, 则在数据传输完成之后, 立即将 3G 接口从 DCH 状态切换至 IDLE 状态; 同时根据后续数据块到达时间的估计值, 提前将 3G 接口提升至 FACH 状态, 以准备数据的传输。

5. 根据权利要求 1 或 4 所述的方法, 其特征是, 所述的去尾处理具体包括以下操作:

a) 启动移动终端后, 3G 接口保持 IDLE 状态;

b) 当 3G 接口在 DCH 状态传输完一组数据时, 如根据自回归滑动平均模型预测得到的下一个数据块的到达时间 \hat{t}_i 小于等于根据实际测量得到的 Promotion 时延 t_{delay} 则保留缓冲状态; 否则设置 3G 接口立即转入 IDLE 状态;

c) 当移动终端处于 IDLE 状态时, 如果一直未有数据包到达, 则设置 3G 接口在 $\hat{t}_i - t_{delay}$

时刻提升至 FACH 状态,为新数据包的到来做准备。

6. 根据权利要求 5 所述的方法,其特征是,所述的缓冲状态是指 :10 秒以内的 DCH 状态以及 20 秒以内的 FACH 状态。

7. 根据权利要求 5 或 6 所述的方法,其特征是,所述的缓冲状态是指 :5 秒的 DCH 状态和 11 秒的 FACH 状态。

8. 根据权利要求 1 所述的方法,其特征是,所述的预测误差修正处理是指 :当某个数据包的到达时间的估计值 \hat{t}_i 早于该数据包实际到达时间 t_i 时,3G 接口会过早地提升至 FACH 状态,从而增加了能量的消耗,则进行修正切换状态处理;当某个数据包到达时间的估计值 \hat{t}_i 晚于实际到达时间 t_i 时,即数据包到来时 3G 接口仍处于 IDLE 状态,则进行估计值修正处理。

9. 根据权利要求 8 所述的方法,其特征是,所述的修正切换状态为以下两种中的任意一种 :

i) 带误差估计的修正方法 :误差序列为 $\{e_i = t_i - \hat{t}_i, i = 1..n-1\}$,当提前提升至 FACH 状态之后,3G 接口通过 ARMA 预测模型得到误差的估计值 \hat{e}_i ;当 $\hat{e}_i < 0$,则将 \hat{e}_i 重新设置成 0,并从此时刻起比较 \hat{e}_i 和阈值时间 t_{th} 的关系 :当 $\hat{e}_i \leq t_{th}$,则设置 3G 接口维持在 FACH 状态并保持继续等待,否则设置 3G 接口从 FACH 到 IDLE 的切换;

ii) 不带误差估计的修正方法 :当提前提升至 FACH 状态之后,设置 3G 接口在 FACH 状态等待阈值时间 t_{th} ;当直到超时都未收到新的数据包,则设置 3G 接口切换回 IDLE 状态,且阈值时间 t_{th} 为固定值。

10. 根据权利要求 8 所述的方法,其特征是,所述的估计值修正处理是指 :当 3G 接口完成第 i 个数据块的传输之后,计算得到下一个数据块的到达时间 \hat{t}_i 和此次估计所产生的误差 \hat{e}_i ;当 $\hat{e}_i < 0$ 时,这表明此次对 IBT 的估计值很有可能偏大,则将下一个数据块到达时间的估计值 \hat{t}_i 修正为 $\hat{t}_i - |\hat{e}_i|$;否则保留原有的 \hat{t}_i 不变。

基于数据流预测的低功耗 RRC 协议优化控制方法

技术领域

[0001] 本发明涉及的是一种无线通信技术领域的方法，具体是一种 3G 无线通信网络环境下基于数据流预测的低功耗无线资源控制协议 (RRC) 优化方法。

背景技术

[0002] 3G 移动通信技术凭借着其高带宽、低时延等特性，在全世界范围内得到了广泛的应用。它的出现，使得更多的互联网应用（如流媒体播放、视频会议等）能够流畅地运行在各种移动终端上，极大地丰富了用户的体验。与此同时，移动终端上的电池技术却成为了瓶颈，终端能耗问题正吸引着越来越多的研究工作随之展开。

[0003] 在 3G 网络中，无线资源控制 (RRC) 协议负责 3G 网络中的移动终端 (UE) 和基站 (BS) 之间数据链路的建立和释放。RRC 协议在每一个终端设备上维护着一个状态机，该状态机负责管理物理链路的状态。通常来说，有三个主要状态：IDLE (UE 空闲状态)、FACH (UE 低速率传输状态) 和 DCH (UE 高速率传输状态)，其中：IDLE 状态表示如果一段时间内没有数据传输，则 3G 接口转入 IDLE 状态。该状态下，终端和基站之间没有数据链路，不能够传输数据，并且基本上不消耗终端能量；DCH 状态表示如果有数据需要收发，则 3G 接口转入 DCH 状态，一条专用的数据链路被分配给终端以供高速率的数据传输，该状态下，能耗最高，大约有 800mW 功率；FACH 状态表示当 3G 接口在 DCH 状态完成一组数据传输，其便转入 FACH 状态。该状态下，一条共享的数据链路被分配给终端，仅支持低速率的数据传输。此时功率约为 400mW。

[0004] RRC 各状态之间有两种转换类型：分别是 UE 由低能耗状态向高能耗状态转换 (Promotion) 和 UE 由高能耗状态向低能耗状态转换 (Demotion)，其中：Promotion 表示 3G 接口从低能耗状态转入高能耗状态，如从 IDLE 到 DCH，或从 FACH 到 DCH。这种转换涉及到专用链路的请求和分配，会带来较大的时延；Demotion 表示 3G 接口从高能耗状态转入低能耗状态，如从 DCH 到 FACH，或者从 FACH 到 IDLE。Demotion 由两个计时控制器 (T1 和 T2)。当 3G 接口在 DCH 状态传输完一组数据后，T1 启动。如果一直到 T1 结束都没有新的数据传输，则 3G 接口转入 FACH 状态，此时 T2 启动。同样，如果一直到 T2 结束都没有新的数据传输，则 3G 接口转入 IDLE 状态。T1 和 T2 合起来被称作尾时间。

[0005] 之所以在数据传输完成之后保留尾时间，是因为在尾时间内，3G 接口实际上仍是处于数据传输状态的 (DCH 或 FACH)。在这种状态下，3G 接口能够更快地响应新的数据包的到来。一方面，较长的尾时间可以减少 Promotion 所带来的时延，提升用户体验度。另一方面，尾时间过长会导致 3G 接口长时间地处于较高能耗状态，从而浪费终端电量。因此尾巴时间的选择，需要在用户体验度和能源效率之间进行权衡。一个好的选择策略应该能够适应不同应用场景下不同的数据传输需求。

[0006] 针对 3G 网络中尾效应能耗问题，现有的解决方案可以归为以下两类：

[0007] 基于应用层的解决方案：不同的应用有着不一样的数据流，如在流媒体应用中，数据包连续不断地到达，且对时延的要求很好。而在电子邮件应用中，数据包的收发时刻相对

比较灵活,而且多以突发形式传输。可将时间不敏感应用(如电子邮件、RSS 订阅等)的数据包捆绑并选择合适时机发送,以减少形成尾巴的数目,从而降低能耗。这种解决方案缺点在于不适用诸如流媒体播放、网页浏览等实时性要求较高的应用。

[0008] 基于链路层的解决方案:一些工作根据网络负载和传输需求动态地调整尾时间,但这会导致 RRC 状态机状态转换次数的增加,而一般而言,从低能耗状态转移到高能耗状态会带来能够被用户感知的时延,因此,这类方法会明显降低用户体验度。现有的方案均没有较深入地分析和利用 3G 网络数据流的特点,在用户体验度和能源效率的权衡上无法满足现有需求。

发明内容

[0009] 本发明针对现有技术存在的上述不足,提出一种基于数据流预测的低功耗 RRC 协议优化控制方法,利用 3G 数据流的时间相关性对其进行预测,并根据预测值调整 RRC 状态机的状态转换策略,进而减少尾效应所带来的能量消耗;同时本发明在减少能耗的同时,并未对用户体验度产生较大影响。

[0010] 本发明是通过以下技术方案实现的,本发明利用自回归滑动平均模型(ARMA)捕捉 3G 数据流中数据块块间间隔时间(Inter-Block Time, IBT)序列并得出 IBT 序列的时间相关性,然后通过对时间相关性进行基于移动终端的能耗转换的去尾处理以及预测误差修正处理,最终实现降低功耗的 RRC 协议优化机制。

[0011] 所述的 IBT 序列为 $\{t_i, i = 1..n-1\}$,其中: t_i 表示第 i 个数据块和第 $i+1$ 个数据块之间的时间间隔, i 为自然常数。

[0012] 所述的 IBT 序列的时间相关性通过以下方式获得:收集移动终端上收发的数据包,并抽象成数据块序列,然后从中提取出块间间隔 IBT 序列;再利用 ARMA 模型捕捉 IBT 序列中的时间相关性建立用于对未来 IBT 的值进行预测的预测模型。

[0013] 由于不同的应用场景(如即时通讯、流媒体播放、网页浏览等等)IBT 序列分布不一,且生成最优的 ARMA 模型所需要的训练数据量也不一样,本发明所述的 ARMA 预测模型针对不同的应用场景进行不同的对应训练。

[0014] 所述的数据块是指:一串连续收发的数据包序列,相邻数据包之间的间隔不超过 500 毫秒。

[0015] 所述的自回归滑动平均模型通过以下步骤得到:

[0016] 1. 1) 采集终端收发的数据包,记录下包头信息,并通过深度包分析技术将数据包按照应用场景进行分类;

[0017] 1. 2) 对每种应用场景下的数据包序列进行预处理,将数据包序列转换成数据块序列,间隔小于 500 毫秒的相邻数据包会被认为属于同一个数据块。得到数据块序列后,从中提取块间间隔时间(IBM),形成 IBT 序列 $\{t_i, i = 1..n-1\}$;

[0018] 1. 3) 采用自回归滑动平均(ARMA)模型对 IBM 序列进行建模,设模型为 $t_i = \phi_0 + \phi_1 t_{i-1} + \phi_2 t_{i-2} + \dots + \phi_p t_{i-p} - \theta_1 \varepsilon_{i-1} - \theta_2 \varepsilon_{i-2} - \dots - \theta_q \varepsilon_{i-q}$, 其中: $\phi_0, \phi_1, \dots, \phi_p, \theta_1, \dots, \theta_q$ 为待定系数, $\varepsilon_1, \dots, \varepsilon_{i-q}$ 为误差分量,模型的训练过程穷举参数 p 和 q 的取值空间,对每一组取值,拟合出待定系数 $\phi_0, \phi_1, \dots, \phi_p, \theta_1, \dots, \theta_q$, 并计算在该系数下的拟合残差的方差 σ^2 , 再计算 $AIC = n \log \sigma^2 + (p+q+1) \log n$ 的值,最终选取

使最小信息准则(AIC)值最大的一组 p 和 q 作为 ARMA 模型参数。

[0019] 1.4) 采用自回归滑动平均模型对预测误差序列进行建模：预测误差序列为 $\{e_i = t_i - \hat{t}_i, i=1..n-1\}$, \hat{t}_i 表示 t_i 的预测值, e_i 表示第 i 个 IBT 值的预测误差,

[0020] $e_i = \Phi_0 e_{i-1} + \Phi_1 e_{i-2} + \dots + \Phi_p e_{i-p} - \varepsilon_i - \theta_1 \varepsilon_{i-1} - \theta_2 \varepsilon_{i-2} - \dots - \theta_q \varepsilon_{i-q}$, 其中：
 $\Phi_0, \Phi_1, \dots, \Phi_p, \theta_1, \dots, \theta_q$ 为待定系数, $\varepsilon_i, \dots, \varepsilon_{i-q}$ 为误差分量。自回归阶数 p 和滑动平均阶数 q 同样可根据最小信息准则测试确定。

[0021] 所述的去尾处理是指：为了减少能耗并在后续数据传输发生时用户体验不到任何提升时延，若后续 IBT 预测值大于提升时延，则在数据传输完成之后，立即将 3G 接口从 DCH 状态切换至 IDLE 状态；同时根据后续数据块到达时间的估计值，提前将 3G 接口提升至 FACH 状态，以准备数据的传输。

[0022] 所述的去尾处理具体包括以下操作：

[0023] a) 启动移动终端后,3G 接口保持 IDLE 状态；

[0024] b) 当 3G 接口在 DCH 状态传输完一组数据时,如根据自回归滑动平均模型预测得到的下一个数据块的到达时间 \hat{t}_i 小于等于根据实际测量得到的 Promotion 时延 t_{delay} 则保留缓冲状态；否则设置 3G 接口立即转入 IDLE 状态；

[0025] 所述的缓冲状态是指：10 秒以内的 DCH 状态以及 20 秒以内的 FACH 状态；优选为：5 秒的 DCH 状态和 11 秒的 FACH 状态。

[0026] c) 当移动终端处于 IDLE 状态时,如果一直未有数据包到达,则设置 3G 接口在 $\hat{t}_i - t_{delay}$ 时刻提升至 FACH 状态,为新数据包的到来做准备。

[0027] 所述的预测误差修正处理是指：当某个数据包的到达时间的估计值 \hat{t}_i 早于该数据包实际到达时间 t_i 时,3G 接口会过早地提升至 FACH 状态,从而增加了能量的消耗,则进行修正切换状态处理；当某个数据包到达时间的估计值 \hat{t}_i 晚于实际到达时间 t_i 时,即数据包到来时 3G 接口仍处于 IDLE 状态,则进行估计值修正处理。

[0028] 所述的修正切换状态为以下两种中的任意一种：

[0029] i) 带误差估计的修正方法：误差序列为 $\{e_i = t_i - \hat{t}_i, i=1..n-1\}$,当提前提升至 FACH 状态之后,3G 接口通过 ARMA 预测模型得到误差的估计值 \hat{e}_i ;当 $\hat{e}_i < 0$,则将 \hat{e}_i 重新设置成 0,并从此时刻起比较 \hat{e}_i 和阈值时间 t_{th} 的关系:当 $\hat{e}_i \leq t_{th}$,则设置 3G 接口维持在 FACH 状态并保持继续等待,否则设置 3G 接口从 FACH 到 IDLE 的切换。

[0030] ii) 不带误差估计的修正方法：当提前提升至 FACH 状态之后,设置 3G 接口在 FACH 状态等待阈值时间 t_{th} ;当直到超时都未收到新的数据包,则设置 3G 接口切换回 IDLE 状态,且阈值时间 t_{th} 为固定值。

[0031] 所述的估计值修正处理是指：当 3G 接口完成第 i 个数据块的传输之后,计算得到下一个数据块的到达时间 \hat{t}_i 和此次估计所产生的误差 \hat{e}_i ;当 $\hat{e}_i < 0$ 时,这表明此次对 IBT 的估计值很有可能偏大,则将下一个数据块到达时间的估计值 \hat{t}_i 修正为 $\hat{t}_i - |\hat{e}_i|$,否则保留原有的 \hat{t}_i 不变。

技术效果

[0032] 本发明主要应用于 3G 无线通信网络环境下移动终端节能,优势在于能够在保证用户体验度的前提下,显著的较少移动终端的能量消耗。本发明通过时间序列分析模型捕捉 3G 数据流中的时间相关性,并建立预测模型。利用未来数据流的预测值,动态地调整 RRC 状态机的转换策略,从而减少额外能量的消耗,同时在用户体验度和能源效率之间做出了较好的权衡。

附图说明

- [0033] 图 1 为 3G 数据流抽象模型 : 数据块和块间间隔 (IBT) 示意图。
- [0034] 图 2 为用 ARMA 模型预测 IBT 序列的结果图。
- [0035] 图 3 为用 ARMA 模型预测误差序列的结果图。
- [0036] 图 4 为低功耗 RRC 协议优化方法对应的 RRC 状态转换图。
- [0037] 图 5 为实施例原型系统的架构图。
- [0038] 图 6 为实施例能源效率指标 R_e 比较图。
- [0039] 图 7 为实施例用户体验度指标 R_p 比较图。
- [0040] 图 8 为实施例能源效率和用户体验权衡度指标 η_e 比较图。
- [0041] 图 9 为实施例和 Fixed-tail 方案在不同应用场景下的能耗比较图。

具体实施方式

[0042] 下面对本发明的实施例作详细说明,本实施例在以本发明技术方案为前提下进行实施,给出了详细的实施方式和具体的操作过程,但本发明的保护范围不限于下述的实施例。

实施例 1

[0043] 如图 1 所示,本实施例包括 :

[0044] 应用场景 :选取 HTC G6 智能手机作为实现低功耗 RRC 协议优化方法的原型机。主要考察网页浏览、流媒体和即时通讯这三类主要的移动应用。采集的数据量为 1000 个用户,14 天的使用时间。

[0045] 原型系统和对比系统 :本实施例选取 HTC G6 智能手机作为原型机。我们通过调用快速休眠实现优化方法中的去尾操作。具体来说,当手机在 DCH 状态完成一次数据传输之后,可以通过控制信道发送一个 SCRI (Signaling Connection Release Indication) 消息给基站。在接收到 SCRI 消息后,基站立即释放分配给相应手机的数据链路,并通知手机将 3G 接口切换至 IDLE 状态,从而可以节约链路资源和能量消耗,实现去尾操作。优化方法还可通过发送短 UDP 包的方法实现提前 Promotion。具体来说,每当需要执行提前 Promotion 操作时,优化方法向某个特定的服务器发送一个短 UDP 包,这样就确保了实际数据到达时,接口已经处于数据传输状态。综上所述的,低功耗 RRC 优化方法的总体架构如图 5 所示。系统分为应用层和内核层。其中 :应用层组件包括 ARMA 模型训练器和预测器,内核层组件包括数据包采集器和调度器。调度器根据 IBT 序列的预测值,调用快速休眠模块和提前 Promotion 模块,进而动态地改变 RRC 状态,减少额外能量的消耗。

[0046] 所述的快速休眠是指 :新的 3G 标准提出的节能方案。当手机在 DCH 状态完成

一次数据传输之后,可以通过控制信道发送一个 SCRI (Signaling Connection Release Indication) 消息给基站。在接收到 SCRI 消息后,基站立即释放分配给相应手机的数据链路,并通知手机将 3G 接口切换至 IDLE 状态,从而节约链路资源和能量消耗。

[0047] 低功耗 RRC 协议优化方法将与以下三个对比方案进行比较:

[0048] Always-on 方案:不管有没有实际的数据传输,3G 接口总是保持在数据传输状态。也即是说,当在 DCH 状态完成一次数据传输之后,接口只是简单地切换至 FACH 状态以等待下一个数据包的到达。

[0049] Always-off 方案:与 Always-on 方案相反,每当完成一次数据传输之后,3G 接口总是立即切换至 IDLE 状态。因此,该方案不会存在尾效应。

[0050] Fixed-tail 方案:每当完成一次数据传输之后,3G 接口维持一段固定长度的尾时间(5 秒的 DCH 状态和 11 秒的 FACH 状态)。该方案普遍地被现有网络运营商所使用,并不能很好地适应不同的应用场景,从而在尾效应上浪费了过多的能量。

[0051] 效果比较:共有三项指标,分别是 R_e 、 R_p 和 η_e ,定义如下: $R_e = E_{extra}/E_{total}$

[0052] $R_p = \#Packet_{undelayed}/\#Packet_{total}$

[0053] $\eta_e = \#Packet_{undelayed}/E_{extra}$

[0054] 其中:

[0055] E_{extra} 为 UE 额外能耗,定义为尾效应与 Promotion 过程引入的能耗之和; E_{total} 为 UE 总能耗,包括尾效应和 Promotion 操作所带来的能耗;总能耗定义为 $E_{transmission} + E_{extra}$, $E_{transmission}$ 为 UE 实际用于数据传输的能耗; $R_e = E_{extra}/E_{total}$:用于反映能源效率的指标,定义为额外能耗与总能耗之比。显然, R_e 越高表明能源效率越低;

[0056] $R_p = \#Packet_{undelayed}/\#Packet_{total}$:用于反映用户体验度的指标,定义为不被用户体验到时延的数据包数与总数据包数之比。显然, R_p 越高表明用户体验度越好;

[0057] $\eta_e = \#Packet_{undelayed}/E_{extra}$:用于反映用户体验度和能源效率之间权衡优劣的指标,定义为不被用户体验到时延的数据包数与额外能耗之比。直观上来说, η_e 表示每消耗一单位的额外能量所带来的用户体验度提升的程度;

[0058] $\#Packet_{total}$ 表示总数据包个数,而 $\#Packet_{undelayed}$ 表示不被用户体验到时延的数据包个数。

[0059] 由于单独考虑能源效率或者用户体验度是不能全面地评价方案的优劣:如,可以利用快速休眠方法去除掉所有的尾巴,即能在能源效率上达到最优。但是这样的设计必然会引起大量的时延,严重地影响用户体验。因此,本发明通过指标 η_e 来刻画方案在能源效率和用户体验度之间权衡的优劣。直观上来说, η_e 表示每消耗一单位的额外能量所带来的用户体验度提升的程度。

[0060] 如图 6 所示,为不同的方案下,数据集中 1000 个用户 R_e 的平均值。从图中可以看出,在所有的应用场景中,Always-on 方案有着最低的能源效率,而 Always-off 方案能源效率最高。我们还可看到,尽管在 Always-off 方案中不存在尾巴效应,但是额外能耗仍然高达总能耗的 50%。其原因在于该方案引入了过多的 Promotion 操作,这些操作同样需要消耗大量的额外能量。值得注意的是,本方法总是比 Fixed-tail 方案有着更高的能源效率。以下为各个应用场景下,本方法相对于 Fixed-tail 方案节省的能耗比较表:

[0061]

应用场景	节省能耗 (%)
网页浏览	49.19
流媒体	56.57
即时通讯	30.16
混合应用	49.56

[0062] 如图 7 所示,为不同的方案下,数据集中 1000 个用户 R_p 的平均值。图中未标出 Always-on 和 Always-off 相应的 R_p 值,是因为前者 R_p 总为 1(所有数据包均未被时延),而后者 R_p 总为 0(所有数据包均被时延)。从图中可以看出,在大部分情况下,本方法比 Fixed-tail 方案引入了更多能被用户体验到时延的数据包,这是为了节省能耗所必须付出的代价。同时,在流媒体应用场景下,本方法比 Fixed-tail 方案有着更高的 R_p 值。其原因可能在于流媒体应用的数据流有着更加显著的模式,本方法可以达到较高的预测精度。

[0063] 如图 8 所示,为在不同的方案下,数据集中 1000 个用户 η_e 的平均值。图中未标出 Always-off 相应的 η_e 值,是因为其 η_e 总为 0(所有数据包均被时延)。从图中可以看出,在所考察的四种方案中,本方法有着最高的 η_e 值。尤其在流媒体应用场景下,本方法的 η_e 值是 Fixed-tail 方案的五倍。这表明本方法更加有效地利用了额外能耗去提高用户体验度。

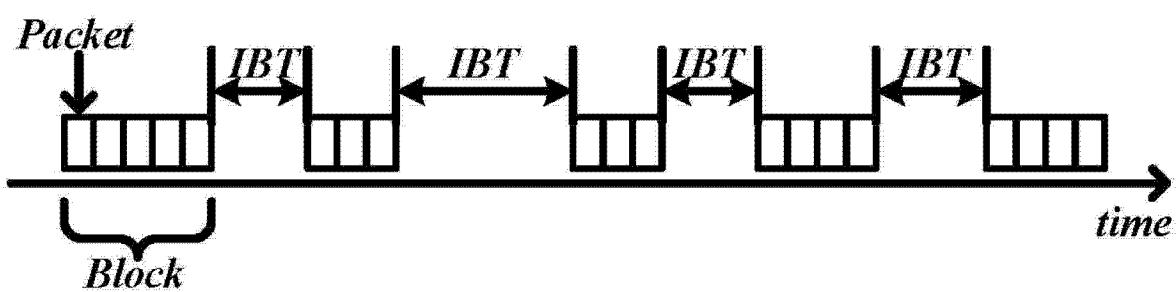


图 1

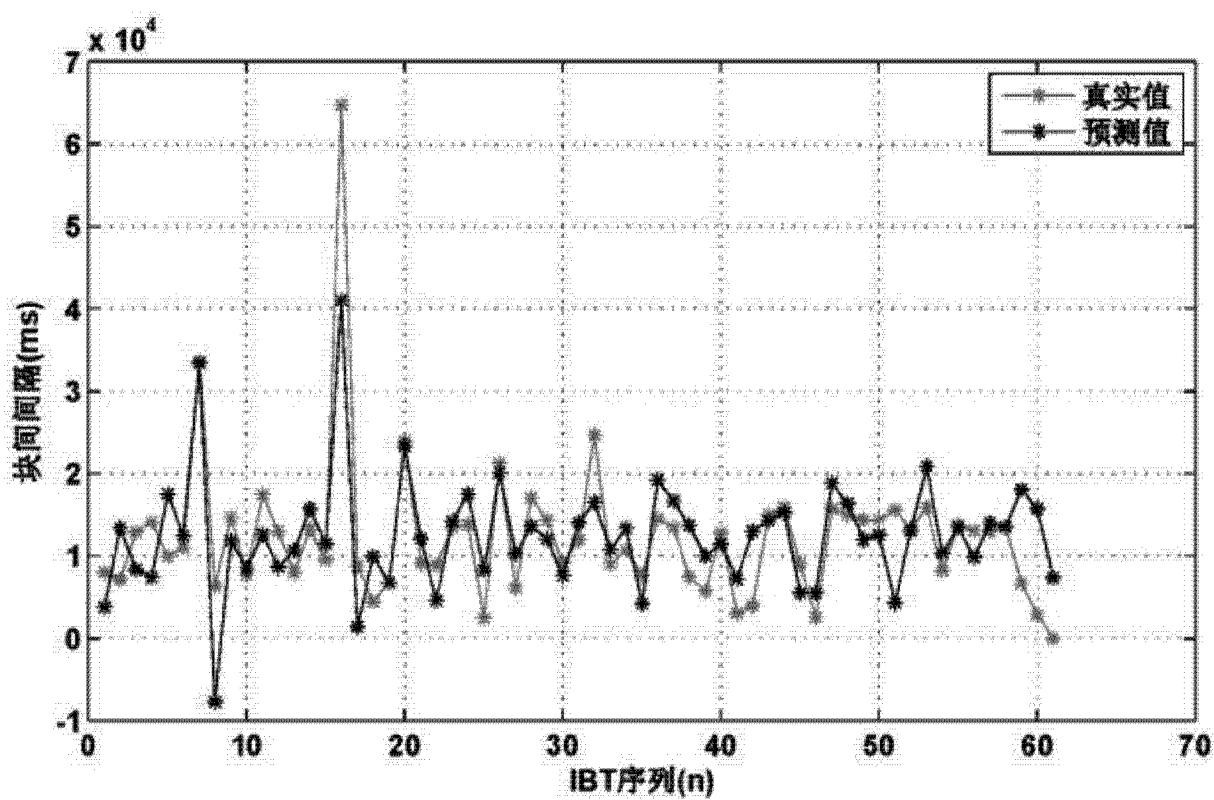


图 2

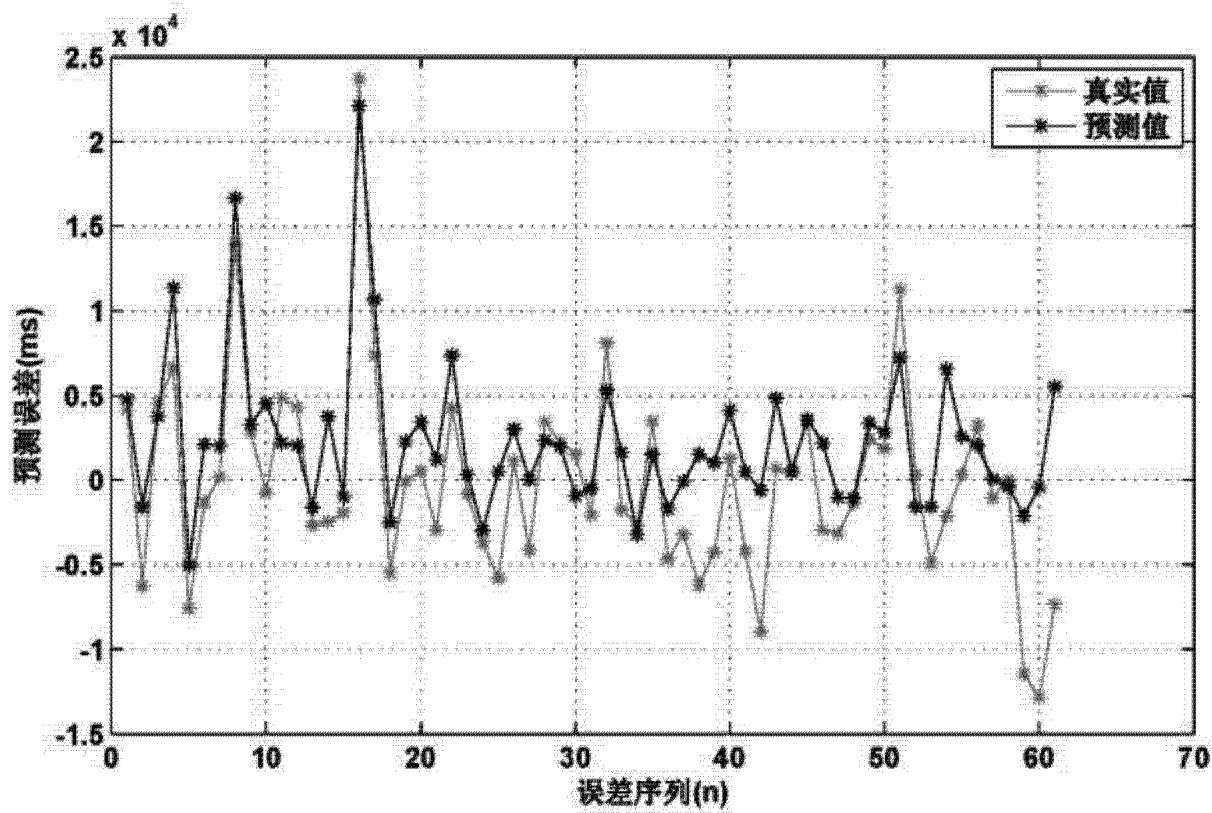


图 3

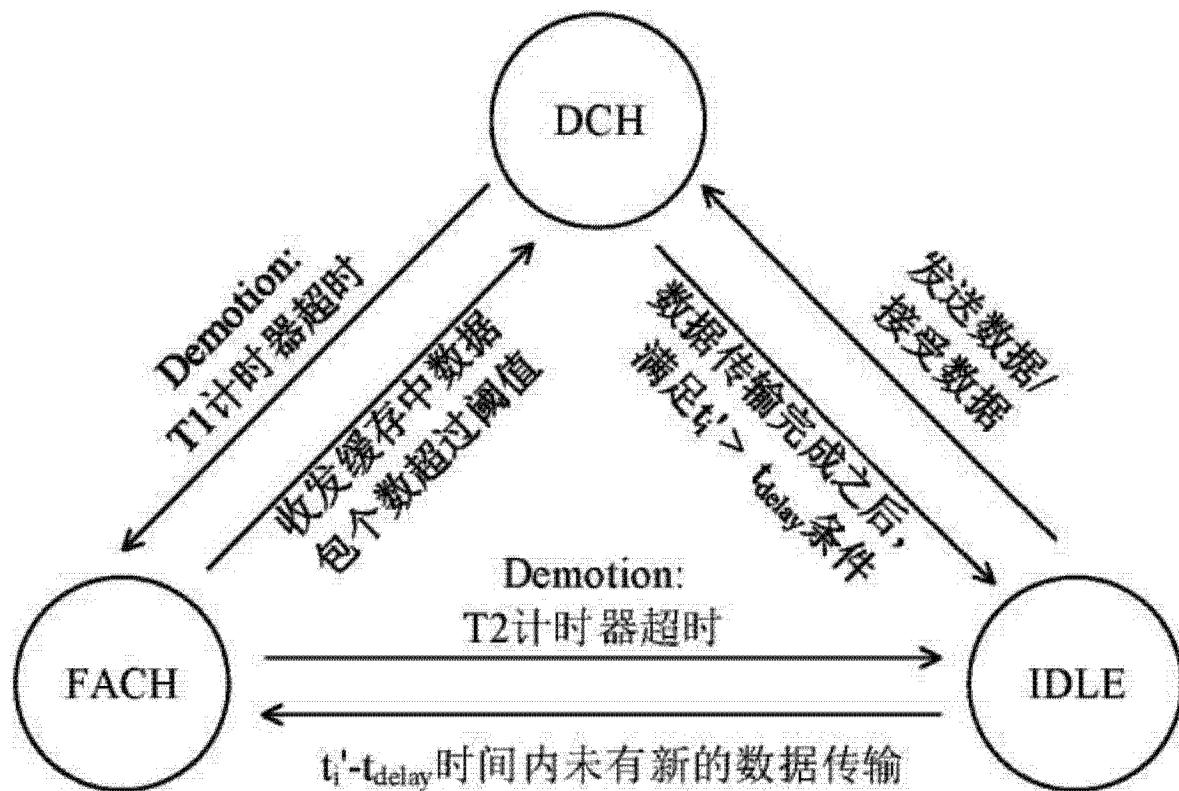


图4

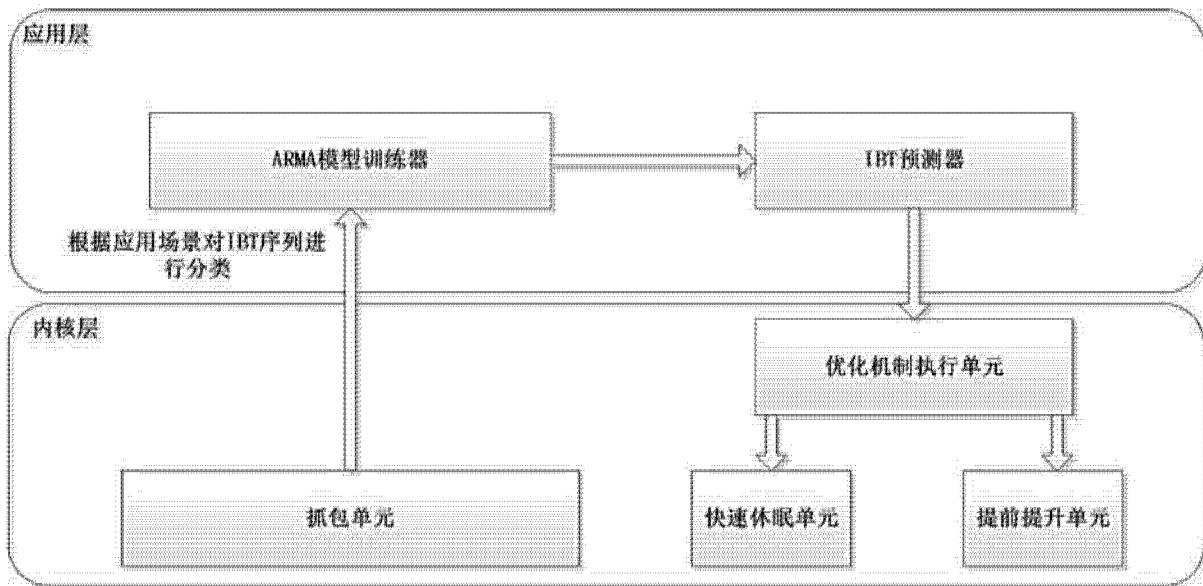


图5

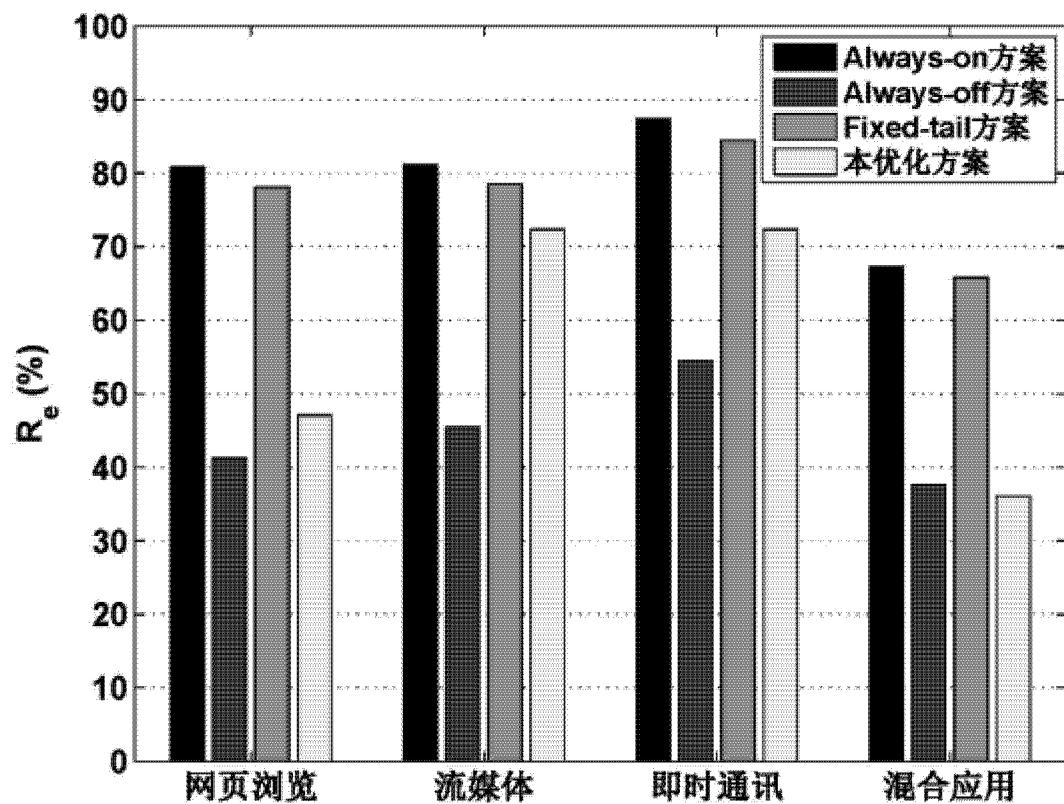


图 6

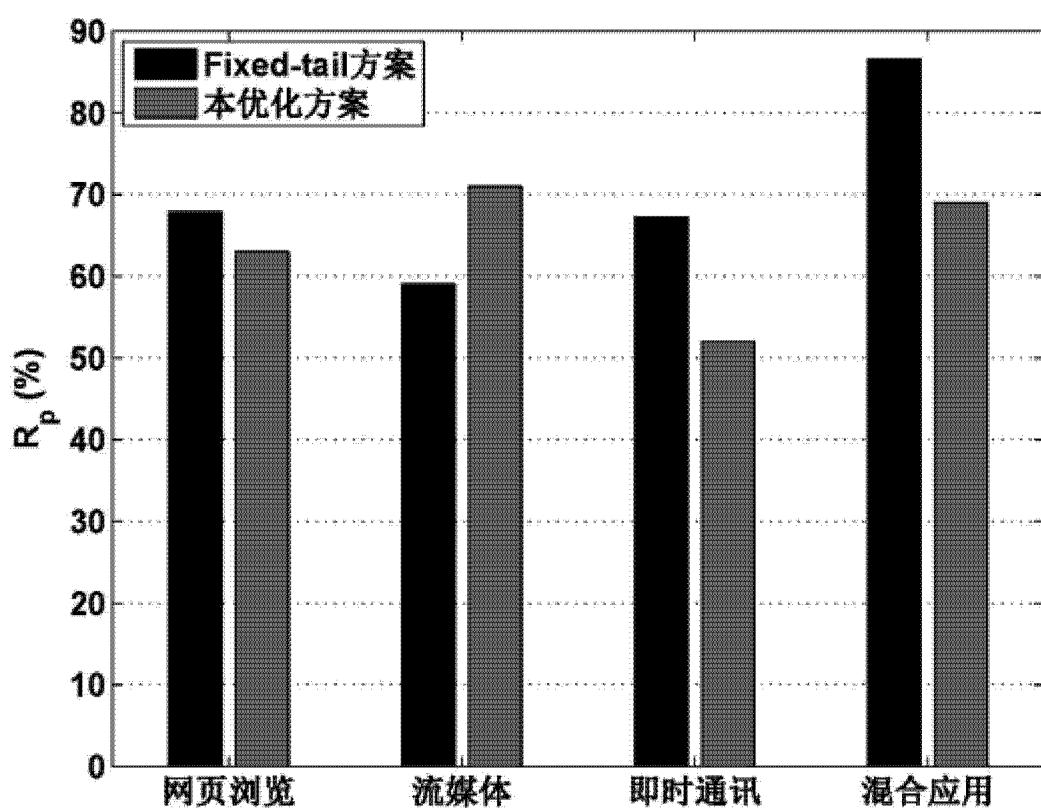


图 7

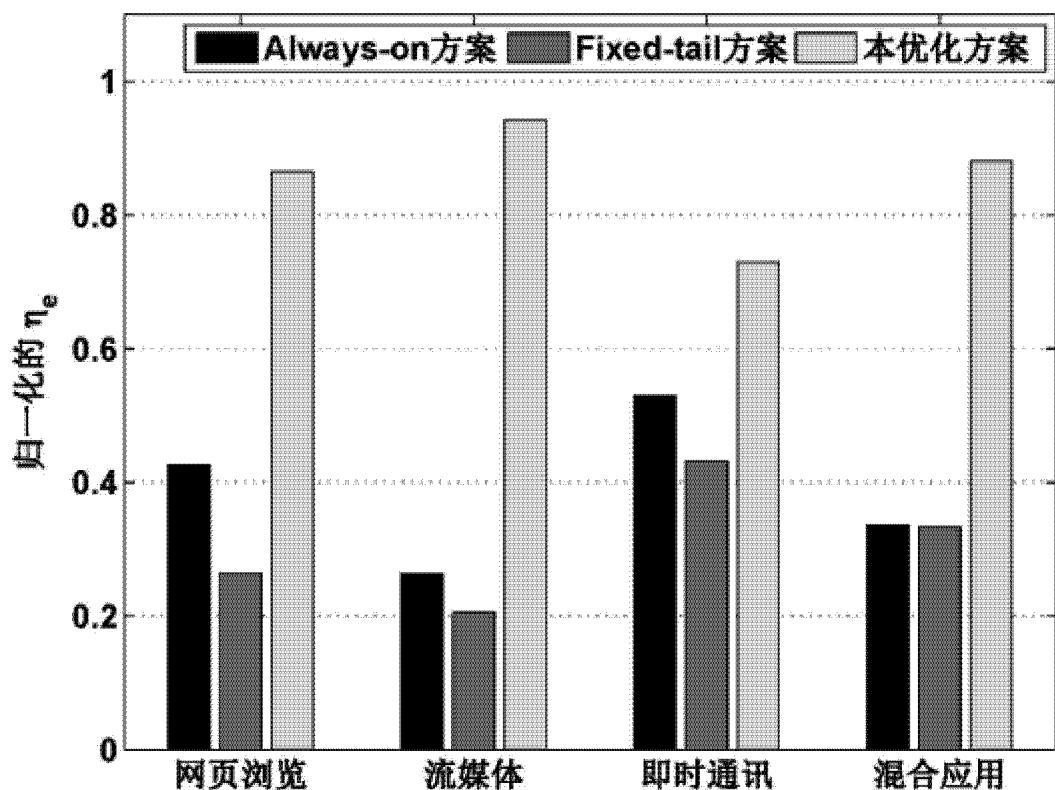


图 8

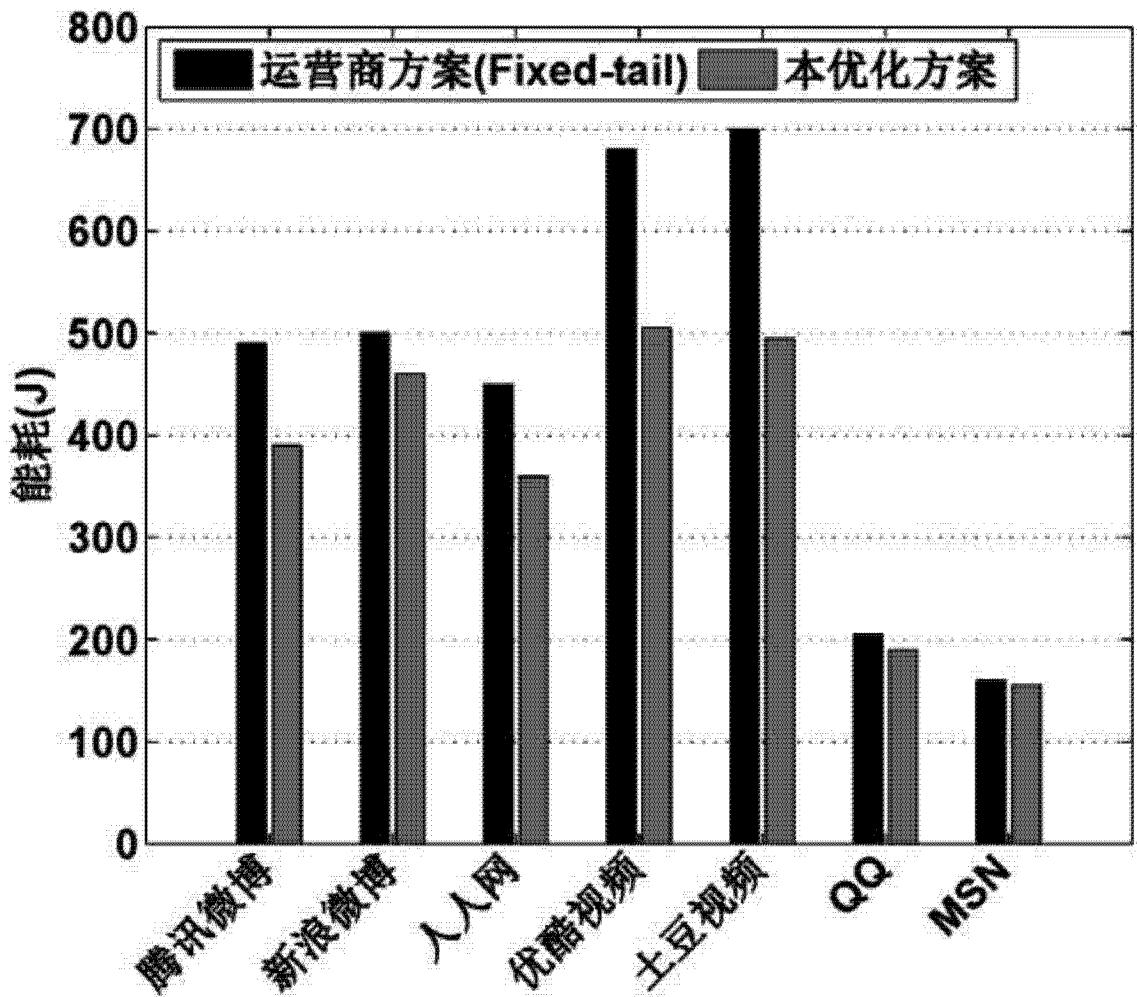


图 9