

# 基于能效的数据包长度自适应算法的研究

朱 静 严 军 杨其阔 王 臣

(上海大学通信与信息工程学院 上海 200072)

**摘要:** 随着无线网络 3G、4G 技术的不断发展和普及,利用无线数据传输终端进行数据交互已经成为信息化发展的重要环节。数据传输终端通信模块消耗的能量是终端各模块消耗的总能量的主要部分,而数据包长度作为通信的主要参数直接影响了终端能耗。在公网条件下,针对节省数据传输终端能耗,延长电池使用寿命的问题,提出了基于 SPSA 的数据包长度自适应算法 ASPSA(adapted simultaneous perturbation stochastic approximation)。该算法可根据信道状况,快速计算出该网络状况下最优数据包长,从而保证数据传输终端能耗最低。理论分析及仿真表明 ASPSA 算法在公网条件下可以有效提高无线数据传输终端的能效。

**关键词:** 公网;能耗;SPSA 算法;自适应;数据包长度

**中图分类号:** TN919 **文献标识码:** A **国家标准学科分类代码:** 510.5025

## Packetlength adaptation for improving the energy efficiency in public network

Zhu Jing Yan Jun Yang Qikuo Wang Chen

(Communication &amp; Information Department of Shanghai University, Shanghai 200072, China)

**Abstract:** With the development of wireless mobile network, 3G and 4G technologies are more and more mature nowadays. In the case of public network, it has been a very important part to use RTU (remote terminal unit) for data exchange in our daily lives. Data communication is the main energy consumption part during the whole transmission process. As the main the communications parameter, packet size directly affects the RTU energy consumption. In order to save energy and extend battery life, this paper proposes a packet length adaptive algorithm ASPSA(adapted simultaneous perturbation stochastic approximation) based on SPSA, which can calculate the most effective payload length and obtain the most optimal packet length. The results of theoretical analysis, simulation and spot tests show that the algorithm aroused in this article can improve the energy efficiency of wireless data transmission terminals greatly.

**Keywords:** public network; energy consumptions; SPSA algorithm; adaptive algorithms; packet length

## 0 引 言

随着全球信息化进程的发展,在公网条件下,利用无线数据传输终端进行数据交互已经成为信息化重要的一部分<sup>[1]</sup>。在实际应用中,公网网络的不稳定会造成数据丢包<sup>[2]</sup>。大部分无线数据传输终端能量由干电池提供<sup>[3]</sup>,降低数据传输终端的能耗一直是研究的重点<sup>[4]</sup>。数据传输终端通信模块消耗的能量是终端各模块消耗的总能量的主要部分,数据包长度作为通信的重要参数了直接影响终端能耗<sup>[5]</sup>。因此,在公网条件下,如何设计能自适应网络状态的数据包长度使得终端能耗最低是设计者迫切需要考虑的问题。目前,针对降低无线数据传输终端能耗的最优数据包长度的主流算法有 SA 算法<sup>[6]</sup>、SPSA 算法<sup>[7]</sup>、DPLC 算法<sup>[8]</sup>

等。考虑到网络状况、能效等多种因素,本文介绍了一种基于能效的公网数据包长度自适应算法。

## 1 数据传输系统及数据包传输机制

本文设计的数据传输系统分为数据传输终端和服务器端两部分。为降低数据传输终端的算法复杂度,本文在服务器端设计了基于能效的数据包长度自适应算法,通过一套数据传输软件加以实现,在公网传输的条件下,通过优化数据包长度,自适应网络状况,提高无线数据传输终端的能效,数据传输总体框图如图 1 所示。

在实际数据传输过程中,服务器端利用基于能效的数据包长度自适应算法,计算出当前网络状况下的最优数据包长度  $L$ ,并应答给数据传输终端。数据传输终端收到服

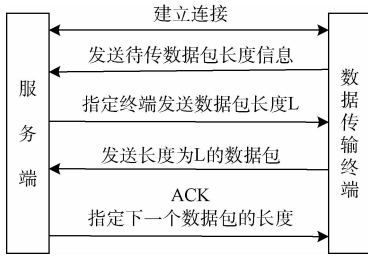


图1 数据传输系统

务器端发来的应答消息后,截取长度为  $L$  的数据包发送至服务器端并等待应答。

## 2 数据包能耗模型

在本文所述的系统中,数据传输终端每次传输的数据包一般由3个部分组成: $\alpha$ bit的包头、 $L$ bit的有效负载和 $\tau$ bit的包尾<sup>[9]</sup>。因此,数据传输终端发送一个数据包并接收由服务器端发送的ACK数据包所需要的能耗  $E$  为:

$$E = E_T + E_R + E_{idle} \quad (1)$$

式中: $E_T$ 表示数据传输终端发送一个数据包所消耗的能量, $E_R$ 表示数据传输终端接收一个ACK数据包所消耗的能量, $E_{idle}$ 表示数据传输终端发送完一个数据包到接收到ACK确认包这段时间产生的空闲能耗。 $E_T$ 和 $E_R$ 计算公式如下:

$$E_T = P_{T\text{-elec}} \times \frac{L_{\text{Data}}}{R} + P_T \times \frac{L_{\text{Data}}}{R} + P_{T\text{-start}} \times t_{T\text{-start}} \quad (2)$$

$$E_R = P_{R\text{-elec}} \times \frac{L_{\text{ACK}}}{R} + P_T \times \frac{L_{\text{ACK}}}{R} + P_{R\text{-start}} \times t_{R\text{-start}} \quad (3)$$

式中: $L_{\text{Data}}$ 表示数据包的长度且满足 $L_{\text{Data}} = \alpha + L + \tau$ , $L_{\text{ACK}}$ 表示ACK确认包的长度; $P_{T\text{-elec}}/P_{R\text{-elec}}$ 是数据传输终端本身的固有功率, $P_T$ 表示发射功率; $P_{T\text{-start}}/P_{R\text{-start}}$ 表示发送和接收部件启动功率, $t_{T\text{-start}}/t_{R\text{-start}}$ 表示数据传输终端启动时间; $R$ 表示公网条件下数据传输速率。

与数据传输能耗相比,空闲能耗 $E_{idle}$ 可忽略不计<sup>[10]</sup>。因此,数据传输终端发送一个数据包并接收由服务器端发送的ACK数据包所需要的能耗  $E$  可表示为:

$$E = E_T + E_R = k_1(L_{\text{Data}} + L_{\text{ACK}}) + 2k_2 \quad (4)$$

$$k_1 = \frac{P_{T\text{-elec}} + P_T + P_{R\text{-elec}}}{R} \quad (5)$$

$$k_2 = P_{T\text{-start}} \times t_{T\text{-start}} + P_{R\text{-start}} \times t_{R\text{-start}} \quad (6)$$

式中: $k_1$ 表示传输1bit信息的能量消耗, $k_2$ 表示传输一个数据包数据传输终端收发部件启动能耗。

假设误比特率为 $BER$ ,已知一个数据包长度为 $L_{\text{data}}$ bit,则一个数据包的丢包率为:

$$P_{\text{drop}} = 1 - (1 - BER)^{L_{\text{max}}} \quad (7)$$

传输一个数据包的能效<sup>[11]</sup>就是成功传输一个包中的有用信息能耗占总能耗的百分比:

$$\eta = \frac{k_1 L (1 - P_{\text{drop}})}{E} \quad (8)$$

## 3 ASPSA 算法

### 3.1 SPSA 算法

同步微扰随机逼近(simultaneous perturbation stochastic approximation,SPSA)算法由随机逼近算法改进而来,它通过估计目标函数的梯度信息来逐渐逼近最优解。

近几年,SPSA算法被成功地应用于多个领域的理论研究和实际应用<sup>[12-14]</sup>,均被证明了实现方便、收敛速度快<sup>[15]</sup>。应用SPSA算法求解最优包长,首先在 $[0, N]$ 中任选一个数值作为初始迭代点,即初始有效负载长度,然后开始迭代。通过对称伯努利±1分布确定微扰方向,该值为 $-1 \sim 1$ 之间的随机数。首先,计算微扰幅度 $c_k$ :

$$c_k = \frac{c}{(k+1)^\gamma} \quad (9)$$

式中: $\gamma=0.101$ , $c$ 为大于0的常数, $k$ 为当前迭代次数。

经过扰动后的两个扰动点所对应的目标函数为 $Y(l_k + c_k \Delta_k)$ 和 $Y(l_k - c_k \Delta_k)$ ,计算这两扰动点所确定的梯度作为迭代处的估计梯度:

$$g_k = \frac{Y(l_k + c_k \Delta_k) - Y(l_k - c_k \Delta_k)}{2c_k \Delta_k} \quad (10)$$

初始步长因子 $a_k$ ,其计算公式如下,其后就可以对迭代点进行更新:

$$A_k = \frac{a}{(k+A+1)^\alpha} \quad k = 0, 1, 2, \dots \quad (11)$$

式中: $\alpha = 0.602$ , $a = \delta_0 \min \left( \frac{A+1}{|g_0(l_0)|} \right)$ , $A \leq 10\% i_{\text{max}}$ , $I_{\text{max}}$ 表示预期迭代总次数, $k$ 表示迭代次数。

由式(10)、(11)可得到下次迭代的有效负载长度 $L_{k+1}$ :

$$l_{k+1} = l_k - A_k g_k \quad (12)$$

### 3.2 ASPSA 算法

基于SPSA算法的最优数据包长度算法是搜索出使能效最高的最优数据包长度,并在整个数据包传输过程中始终以该数据包长度进行传输。其缺点是始终以定长数据包进行传输,无法适应网络变化。当网络状态良好时,造成带宽的浪费;而当网络状态差时,则会造成丢包。此外,该算法始终以固定步长增加或减少数据包长,很容易导致搜索不到最佳包长。本文提出的基于SPSA的公网数据包长度自适应算法即ASPSA算法成功解决了以上问题。

ASPSA算法分为数据包长度变化趋势自适应算法和动态改变步长算法。服务器端与数据传输终端建立连接后,对于每一次数据包的传输,服务器端利用数据长度自适应算法得到初始步长因子 $a_0$ ,及初始迭代数据包长度 $L_0$ 。通过动态改变步长算法计算每次迭代后的步长因子 $a$ 和数据包长度 $L$ ,并记录每次迭代后的最优步长,从而得到最终使传输终端能效最高的最优步长 $L_{\text{op}}$ 。ASPSA算法流程如图2所示。

1)数据包长度变化趋势自适应算法

服务器端记录数据传输终端上次发送数据包所消耗的

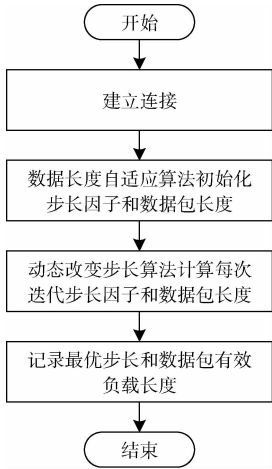


图 2 ASPSA 算法流程

能耗  $E_{last}$ , 并计算当前能耗值  $E_{current}$ 。若当前能耗值  $E_{current}$  低于上一状态能耗值  $E_{last}$ , 则继续保持上一状态包长变化趋势, 且每次迭代的步长初始值绝对值  $|a_0|$  被设为逐步减小。若当前能耗值  $E_{current}$  高于上一状态能耗值  $E_{last}$ , 则继续保持上一状态数据包长度, 并改变步长变化的符号, 不改变步长初始值大小。若当前能耗值  $E_{current}$  较上一状态能耗值  $E_{last}$  没有变化, 则继续保持上一状态包长变化趋势, 且每次迭代的步长初始值绝对值  $|A_0|$  增加。

2) 动态改变步长算法

动态改变步长算法流程如图 3 所示。记录使得  $Y$  最小的数据包长, 并设为初始包长, 此处为上一状态的数据包长度。若  $\min\{y(l_k + c_k \Delta_k), y(l_k - c_k \Delta_k)\} - y(l_0) \geq 0$ , 则  $l_{k+1} = l_{min}, a_{k+1} = 0.5a_k$ , 从而得到使得能耗最低的最优数据包长度  $L_{op}$ 。

4 仿真与测试

本文仿真环境为 MATLAB, 参数设置如表 1 所示。

表 1 能效公式参数

参数	$k_1$	$k_2$	包头 $\alpha$	包尾 $\tau$	$\alpha$	$\gamma$
值	1.85	24.86	16	16	0.602	0.101

在实际应用环境中, 3G、4G 网络误比特率随时间不断变化, 本文设定平均 5 s 发生一次变化, 如图 4 所示。

本文所述算法的目的即在这 5 s 里按照算法的机制调节数据包的负载长度, 使网络能效最大。本文对基于 SPSA 的固定步长算法与 ASPSA 算法进行了对比分析。

图 5 展示了这两种不同算法中数据包长度随时间的变化曲线。从仿真图可看到固定包长优化算法的最优数据包负载为 238 bit。ASPSA 算法的数据包负载长度可随着误比特率的变化而变化, 且数据包负载长度变化相对平稳。

图 6 显示了能效值随时间的变化曲线。从图中可看

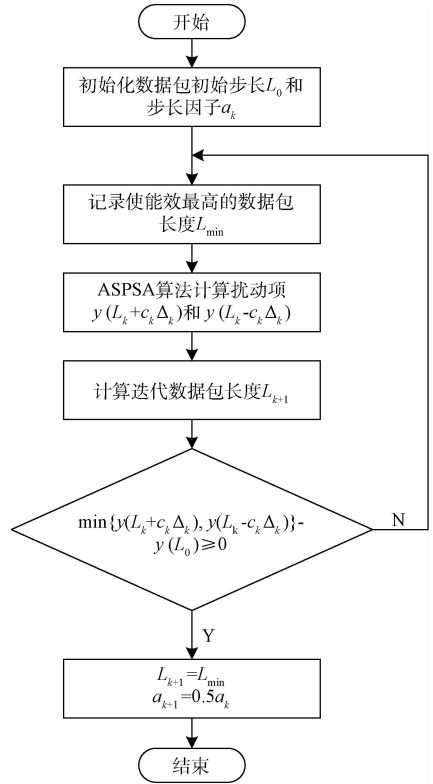


图 3 动态改变步长算法流程

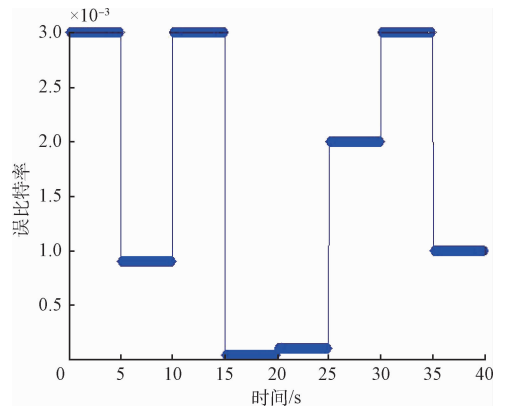


图 4 误比特率变化情况

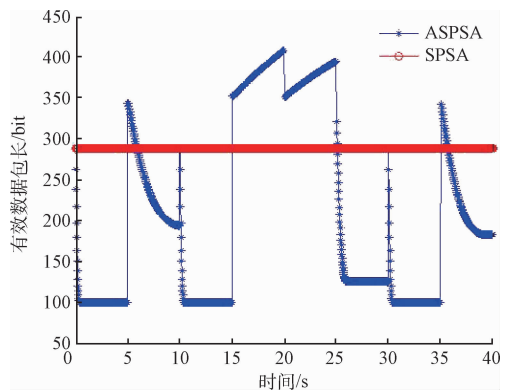


图 5 两种算法包负载长度的变化情况

出,采用固定数据包长度优化算法得到的能效基本与误比特率成相反变化趋势,误比特率越高,能效越低,即固定数据包长度优化算法并不能减弱误码率对能效带来的负面影响。ASPSA 算法能效有所提高,整体处于优势地位。

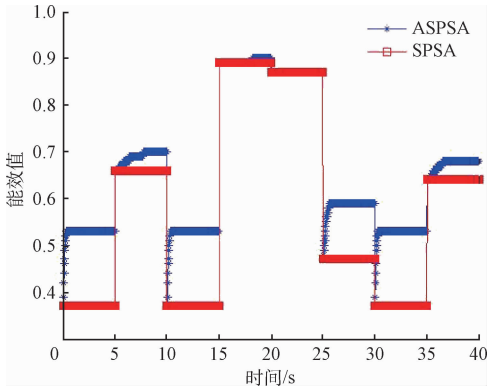


图 6 两种算法中能效的变化情况

## 5 结 论

本文讨论公网条件下基于能效的数据包长度自适应算法。针对公网网络不稳定,数据传输终端功耗高等问题,本文提出了基于 SPSA 的公网数据包长度自适应算法 ASPSA 算法,通过数据包长度变化趋势自适应和动态改变步长,根据信道变化动态改变数据包长,保证数据传输终端能效最高。仿真实验证明,本文提出的 ASPSA 算法相对于原有算法在公网条件下可以有效节省数据传输终端能耗,延长电池使用寿命。

## 参考文献

- [1] 于思江. 基于 LBS 的 Android 校园服务 APP 客户端的设计[J]. 国外电子测量技术, 2015, 34(4): 73-77.
- [2] 魏振春, 吴亚伟, 张本宏. 一种用于多信道无线 Mesh 网络的信道分配方法[J]. 电子测量与仪器学报, 2015, 29(8): 1144-1150.
- [3] 瞿佳俊, 严军, 朱渊婧. 无线 s 传感网低开销型数据

可靠传输方法的研究[J]. 电子测量技术, 2014, 37(1): 92-95, 108.

- [4] 连彬. 基于能效分析的无线传感器网络包长度自适应算法研究[D]. 邯郸:河北工程大学, 2014.
- [5] 杨晨. 面向能耗均衡的无线传感器网络路由协议研究[D]. 长沙:中南大学, 2013.
- [6] 卫星, 张芳荣, 段章领, 等. 基于模拟退火的无线传感节点重编程策略[J]. 仪器仪表学报, 2015, 36(3): 537-544.
- [7] 袁微, 朱学峰. 基于单边 SPSA 的动态偏差数据驱动控制算法[J]. 华南理工大学学报, 2012, 40(9): 81-86, 92.
- [8] WEI D. Dynamic packet length control in wireless sensor networks[J]. IEEE Transactions on Wireless Communications, 2014, 13(3):1172-1181.
- [9] 王亚钢. 水下传感器网络中数据包长优化算法研究[D]. 南昌:南昌航空大学, 2014.
- [10] 蒋文贤, 程光. 无线传感器网络能效模型的量化评价与优化[J]. 哈尔滨工业大学学报, 2014, 46(5): 87-94.
- [11] 宋杰, 李甜甜, 闫振兴. 一种云计算环境下的能效模型和度量方法[J]. 软件学报, 2012, 23(2): 200-214.
- [12] 李双江. 传感器网络 SPSA 声源定位算法研究[D]. 合肥:合肥工业大学, 2012.
- [13] 万琦. 基于最优控制与 SPSA 算法的水驱油藏优化方法研究[D]. 成都:西南石油大学, 2015.
- [14] 丁胜, 蒋建国, 夏娜, 等. 无线监测网络中多信道优化选择算法[J]. 合肥工业大学学报:自然科学版, 2016, 39(2): 177-183.
- [15] 袁微. 基于随机逼近的数据驱动控制方法研究[D]. 广州:华南理工大学, 2011.

## 作者简介

朱静, 1992 年出生, 工学硕士, 主要研究方向为物联网等。

E-mail: zhujing1024@163.com