

动态矩阵主动队列管理算法

王 萍, 陈 虹, 杨晓萍

(吉林大学 通信工程学院, 吉林 长春 130025)

摘要: 针对Internet网络这种大延迟, 变化复杂, 存在严重干扰的系统, 基于约束模型预测控制理论, 提出了一种新的主动队列管理算法(DMAQM算法). 首先根据预测模型和状态估计器, 预测瓶颈链路队列的未来动态特性, 它是确定丢包率的基础; 然后将拥塞控制的控制要求转化为优化目标函数, 在线求解优化问题得到丢包率. DMAQM算法通过滚动优化来适应网络环境的变化, 提高了算法鲁棒性; 同时在求取丢包率时显式地考虑了网络中存在的约束. 仿真结果显示DMAQM算法控制性能优于RED算法, 在满足约束的前提下, 具有较强的抗干扰能力, 同时具有较好的稳定性和鲁棒性.

关键词: 网络拥塞控制; 主动队列管理; 动态矩阵控制

中图分类号: TP273 **文献标识码:** A

Active queue management algorithm with dynamic matrix control

WANG Ping, CHEN Hong, YANG Xiao-ping

(School of Communication Engineering, Jilin University, Changchun Jilin 130025, China)

Abstract: We propose a novel active queue management algorithm based on the dynamic matrix control(DMAQM) for the Internet system with large time-delay, complex variations and detrimental disturbances. By the predictive model and state estimator, the future queue length in data buffer, which is the basis for determining the packet-loss rate, is predicted. Furthermore, the control requirements are converted to the optimal control objectives, and then, the packet-loss rate is obtained by solving the optimization problem online. The DMAQM algorithm adapts to the varying network environment and improves the robustness by moving the optimization horizon, and handles network constraints in the process for determining the packet-loss rate. Finally, the performances of DMAQM are evaluated through a series of simulations. The simulation results show that the DMAQM algorithm is superior to the random early detection(RED) algorithm in stability, disturbance rejection, and robustness.

Key words: network congestion control; active queue management; dynamic matrix control

1 引言(Introduction)

随着互联网规模的飞速发展和业务需求的不断增加, TCP/IP拥塞控制对保证网络的正常运行有着重要的作用. 端到端的TCP拥塞控制机制具有局限性, 难以提供良好的服务质量保证. 主动队列管理(active queue management, AQM)^[1]技术是在TCP窗口控制机制基础上发展的一种路由器拥塞控制算法, 能在拥塞发生之前就对TCP源端的发送量进行控制, 在保证较高吞吐量的同时, 有效的控制了缓存队列的长度. 基于路由器的拥塞控制与TCP端到端的拥塞控制相结合, 是目前解决Internet拥塞控制问题的一个主要途径.

早期的队列管理算法采用“丢弃算法”(drop-

tail), 当发生拥塞时, 路由器缓冲区溢出而丢弃队列尾部的数据包. 1998年关于“Internet中的队列管理和避免拥塞的建议”的RFC(request for comments)报告[RFC2309]中强烈建议在路由器队列管理算法中使用AQM策略, 并推荐RED(random early detection)^[1]为候选算法. 但RED性能敏感于网络参数, 在一定的网络负载下依然会导致多个TCP同步; RED的公平性和稳定性也存在问题.

由于TCP/AQM流动态模型的建立^[2], 应用控制理论设计AQM算法中的控制器或补偿器成为拥塞控制研究的热点. PI/PID控制器被比较早地用到拥塞控制中^[3,4], 并进一步与智能控制、遗传算法等结合使用^[5~7], 但不能对控制机制进行优化设计.

预测控制^[8~10]是一种可以显式处理系统约束的在线滚动优化控制方法,它已经被成功应用到了与网络拥塞控制相似的环境中^[11,12],针对它们存在的延迟,不可测干扰等问题具有很好的控制效果,同时表现出了很强的鲁棒性.预测控制可以预测网络的未来状态,进行优化设计,而且针对多变的网络环境,预测控制通过在线优化,反馈校正等提高系统的自适应能力.文献[13]运用广义预测控制(GPC)设计AQM算法,不考虑系统约束的情况下,进行了稳定性的分析.

本文将预测控制理论与网络拥塞控制相结合,将网络受到的限定转化为系统约束,将网络性能指标转化为控制目标,运用带约束的动态矩阵控制理论设计一种新的主动队列管理算法(DMAQM算法).DMAQM算法具有较强的抗干扰能力,同时具有较好的鲁棒性.

2 TCP/AQM 控制模型(TCP/AQM control model)

2.1 流模型(Fluid flow model)

文献[2]基于流体(fluid flow)理论建立了在AQM作用下动态TCP流量控制的非线性模型.在忽略TCP超时机制的情形下,考虑到往返时间以及路由器中队列长度的动态特性,可得TCP拥塞控制系统的微分方程为:

$$\frac{dw(t)}{dt} = \frac{w(t-R_0)}{w(t)R(t-R_0)}(1-p(t-R_0)) - \frac{w(t)w(t-R_0)}{2R(t-R_0)}p(t-R_0), \quad (1a)$$

$$\frac{dq(t)}{dt} = -C + n \frac{w(t)}{R(t)}, \quad (1b)$$

$$R(t) = \frac{q(t)}{C} + T_p. \quad (1c)$$

其中: w 为 TCP 窗口大小 (packets), C 为链路容量 (packets/s), n 为激活的TCP会话数, p 为分组丢弃概率, q 为队列长度 (packets), R 为往返时延(s), T_p 为固定的广播时延(s), R_0 为平衡状态下的往返时延(由公式(2c)计算).

把 (w, q) 作为状态, p 作为输入, q 作为输出, q_0 是期望队列长度,平衡点 (w_0, q_0, p_0) 则可以通过 $\frac{dw(t)}{dt} = 0$ 和 $\frac{dq(t)}{dt} = 0$ 得到,即满足:

$$w_0^2 p_0 = 2(1 - p_0), \quad (2a)$$

$$w_0 = \frac{R_0 C}{n}, \quad (2b)$$

$$R_0 = \frac{q_0}{C} + T_p. \quad (2c)$$

系统(1)在该平衡点的归一化非线性方程为

$$\delta \ddot{q}(t) = \frac{n(\delta \dot{q}(t-R_0) + C/q_0)}{q_0^3(R_0/q_0 + \delta q(t)/C)^2(\delta \dot{q}(t) + C/q_0)} - \frac{(\delta \dot{q}(t) + C/q_0)\delta \dot{q}(t)}{(R_0/q_0 + \delta q(t)/C)C} - \left\{ \frac{n(\delta \dot{q}(t-R_0) + C/q_0)}{q_0^3(R_0/q_0 + \delta q(t)/C)^2(\delta \dot{q}(t) + C/q_0)} + \frac{q_0(\delta \dot{q}(t-R_0) + C/q_0)(\delta \dot{q}(t) + C/q_0)}{2n} \right\} \cdot (p_0 \delta p(t-R_0) + p_0). \quad (3)$$

其中:

$$\delta \ddot{q}(t) = \frac{\ddot{q}(t)}{q_0}, \delta \dot{q}(t) = \frac{\dot{q}(t)}{q_0},$$

$$\delta q(t) = \frac{q(t) - q_0}{q_0}, \delta p(t) = \frac{p(t) - p_0}{p_0}.$$

为避免网络波动带来的性能下降,丢包率稳定时需要工作在平衡点附近的小范围内,使缓冲区能为新到来的数据流预留存储空间.

下面具体分析一下系统(3)在平衡点附近的特性.设定 $C = 3750$ packets/s, $n = 120$, $T_p = 0.15$ s, $q_0 = 200$ packets(T_p 的取值为大延迟),求得 $p_0 = 0.0472$.将系统(3)的控制输入 $\delta p(t)$ 分别取为幅值为0.1, 0.05的正,负阶跃信号,通过图1、图2所示的阶跃响应输出出来分析平衡点附近系统(3)的线性度.

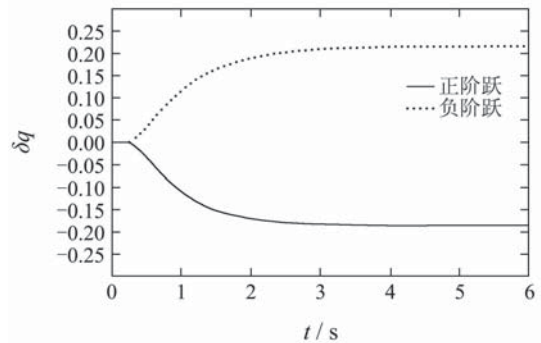


图1 幅值为0.1的正、负阶跃响应曲线
Fig. 1 The positive/negative step response curve with amplitude of 0.1

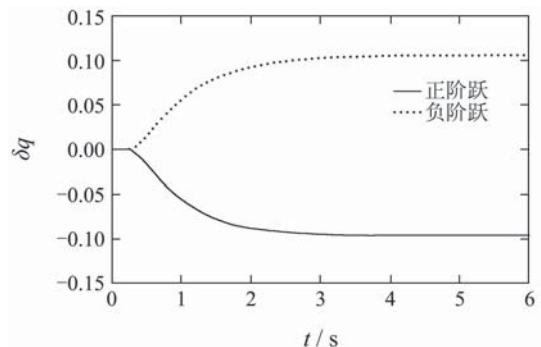


图2 幅值为0.05的正、负阶跃响应曲线
Fig. 2 The positive/negative step response curve with amplitude of 0.05

随着输入信号幅值的减小, 正负阶跃响应曲线的对称程度越来越好, 因此在平衡点附近, 可以认为整个网络是一个线性系统.

2.2 阶跃响应模型(Step response model)

给定 C, n, T_p, q_0 , 根据上一节的分析合理地假设系统(3)在平衡点附近是一个线性稳定系统, 则用阶跃响应模型*来描述非线性系统(3)在平衡点附近的输入输出关系. 定义控制量 $u(k) = \frac{p(k) - p_0}{q_0}$, 控制量变化 $\Delta u(k) = u(k) - u(k - 1)$, 系统输出 $y(k) = \frac{q(k) - q_0}{q_0}$, 采样时间为 T_s .

首先假设系统 $k - 1$ 时刻以后输入不变, 而且非零初始状态响应的过渡过程时间为 N 个采样间隔, 则定义 $k - 1$ 时刻的未来 N 步输出为

$$Y(k-1) \triangleq \begin{cases} [y(k-1) \ y(k) \ \cdots \ y(k+N-2)]^T, \\ \Delta u(k+i) = 0, i \geq -1, \end{cases}$$

显然, $Y(k-1)$ 满足状态变量在时间上的因果性. 因此, 称 $Y(k-1)$ 为系统在 $k-1$ 时刻的“状态”. 定义 k 时刻的状态为

$$Y(k) \triangleq \begin{cases} [y(k) \ y(k+1) \ \cdots \ y(k+N-1)]^T, \\ \Delta u(k+i) = 0, i \geq 0, \end{cases}$$

则当 $\Delta u(k-1) = 0$ 时, 有

$$Y(k) = M_{ss} Y(k-1), \quad (4)$$

其中

$$M_{ss} = \begin{bmatrix} 0 & 1 & 0 & \cdots & 0 & 0 \\ 0 & 0 & 1 & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & \cdots & 0 & 1 \\ 0 & 0 & 0 & \cdots & 0 & 1 \end{bmatrix}_{N \times N}$$

接着在 $k - 1$ 时刻, 对系统零初始平衡状态下的幅值为 F 的阶跃响应进行采样, 得到的响应序列为 $\{0, s_1, s_2, \dots, s_N, s_N, \dots\}$, 由线性系统的可乘性得, 对任意的输入变化 $\Delta u(k - 1)$, 系统的响应序列为 $\{0, s_1, s_2, \dots, s_N, s_N, \dots\} \frac{1}{F} \Delta u(k - 1)$, 记阶跃响应矩阵为 $S = [s_1 \ s_2 \ \cdots \ s_N]^T$, 则零初始状态下, 系统对任意变化输入的响应可以描述为

$$Y(k) = \frac{S}{F} \Delta u(k - 1). \quad (5)$$

由式(4)和(5)得到, 在非零初始状态下系统对任意输入变化的响应为

$$Y(k) = M_{ss} Y(k - 1) + \frac{S}{F} \Delta u(k - 1), \quad (6)$$

其中初始条件为 $Y(0) = [y(0) \ \cdots \ y(0)]_{1 \times N}^T$.

系统在 k 时刻的输出为

$$y(k) = C_y Y(k), \quad (7)$$

其中 $C_y = [1 \ 0 \ \cdots \ 0]_{1 \times N}$. 综上, 式(6)和(7)就是系统(3)在平衡点附近的阶跃响应模型.

注 1 从阶跃响应模型(6)可以推导出文献[14]中给出的阶跃响应模型的常规形式(8)*:

$$y(k) = \sum_{i=1}^N \frac{s_i}{F} \Delta u(k - i) + \frac{s_N}{F} u(k - N - 1), \quad (8)$$

式(6)和(8)是等价的, 但前者具有状态空间的形式, 它便于用状态空间法设计观测器.

注 2 对具有输入时延的网络系统(3)建立阶跃响应模型(6)时, 时延 R_0 的影响只体现在阶跃响应系数矩阵 S 中, 并没有给模型(6)的表达式带来任何变化, 即时延为 $\theta = \lfloor \frac{R_0}{T_s} \rfloor$ ($\lfloor \cdot \rfloor$ 表示向下取整)时, 则 $s_1 = s_2 = \cdots = s_\theta = 0$.

在 2.1 节实例中, 根据幅值为 0.05 的正阶跃响应输出建立阶跃响应模型(即 $F = 0.05$), 调节时间为 2.475 s, 参考文献[13]中的采样时间, 取 $T_s = 0.075$ s, 则 $N = 34, \theta = 3$, 对图 2 中的正阶跃响应曲线进行定时采样得到模型(6)的阶跃响应系数 S (见附录).

3 主动队列管理的动态矩阵算法(Dynamic matrix algorithm of AQM)

图 3 给出了基于动态矩阵控制(DMC)^[15]的网络拥塞控制系统结构, 系统输出为队列长度, 控制量为丢包率, 运用 DMC 设计了一种新的 AQM 算法——DMAQM.

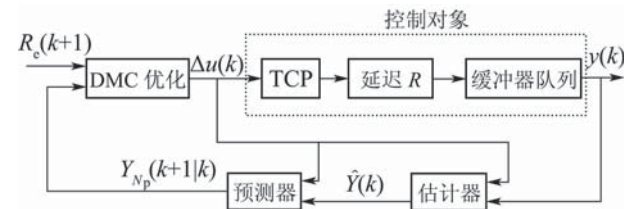


图 3 基于动态矩阵控制的网络拥塞控制系统
Fig. 3 Congestion control system based on DMC

DMAQM 算法的核心思想是根据测量的队列长度估计阶跃响应模型(6)的状态量, 以估计状态为起点, 基于模型(6)预测系统未来队列长度; 通过优化丢包率序列使得预测的队列长度与期望值之间的误差最小, 同时满足系统约束条件. 优化丢包率序列的第 1 步将作为反馈控制量作用于系统. 这个预测-优化-反馈的过程在每个采样时刻重复进行, 使得队列长度较快到达期望值.

* 陈虹. 预测控制讲义, 2008.

3.1 状态估计器的设计(Design of state estimator)

由于阶跃响应模型(6)的状态中只有第1个分量是可以测量的, 所以需要对状态进行估计. 基于 $k-1$ 时刻的估计状态 $\hat{Y}(k-1)$ 和输入变量 $\Delta u(k-1)$ 由式(6)来计算 k 时刻的状态 $\hat{Y}(k)$, 即用状态空间法设计估计器

$$\hat{Y}(k) = M_{ss}\hat{Y}(k-1) + \frac{S}{F}\Delta u(k-1) + K_f(y(k-1) - C_y\hat{Y}(k-1)), \quad (9)$$

其中 K_f 是估计器增益矩阵, 它既要保证 $M_{ss} - K_fC_y$ 渐近稳定, 又要保证估计误差快速收敛. 本文采用区域极点配置, 将 $M_{ss} - K_fC_y$ 的极点配置在以原点为圆心, 半径为 r_0 的圆盘区域内($r_0 < 1$), 基于LMI的方法得到 K_f , 下面给出区域极点配置条件.

引理 1^[16] 设 A 是任一方阵, $\sigma(A) \subset D(r, d)$ 当且仅当存在矩阵 $X > 0$, 使得

$$\begin{bmatrix} -rX & AX - dX \\ XA^T - dX & -rX \end{bmatrix} < 0, \quad (10)$$

其中: $\sigma(A)$ 表示矩阵的谱集, $D(r, d)$ 表示以 $(d, 0)$ 为圆心, r 为半径的圆盘.

由于方阵 $M_{ss} - K_fC_y$ 满足 $\sigma(M_{ss} - K_fC_y) = \sigma(M_{ss}^T - C_y^TK_f^T)$, 所以有下面的推论.

推论 1 对于方阵 $M_{ss}^T - C_y^TK_f^T$, $\sigma(M_{ss}^T - C_y^TK_f^T) \subset D(r_0, 0)$ 当且仅当存在矩阵 $X > 0$, 使得

$$\begin{bmatrix} -r_0X & M_{ss}^TX - C_y^TK_f^TX \\ XM_{ss} - XK_fC_y & -r_0X \end{bmatrix} < 0. \quad (11)$$

推论 2 对于方阵 $M_{ss}^T - C_y^TK_f^T$, $\sigma(M_{ss}^T - C_y^TK_f^T) \subset D(r_0, 0)$ 当且仅当存在矩阵 $X > 0$ 和 Q , 使得

$$\begin{bmatrix} -r_0X & M_{ss}^TX - C_y^TQ^T \\ XM_{ss} - QC_y & -r_0X \end{bmatrix} < 0. \quad (12)$$

当式(12)有解 Q 和 $X > 0$ 时, $K_f = X^{-1}Q$ 是估计器(9)的一个满足要求的增益矩阵.

3.2 未来队列长度预测(Predictive future queue length)

在 k 时刻, 以估计的状态 $\hat{Y}(k)$ 为初始条件, 基于模型(6)预测系统未来的动态. 设定控制时域为 N_c , 预测时域为 N_p , 定义向量

$$Y_{N_p}(k+1|k) \triangleq \begin{bmatrix} y(k+1|k) \\ y(k+2|k) \\ \vdots \\ y(k+N_p|k) \end{bmatrix},$$

$$\Delta U(k) \triangleq \begin{bmatrix} \Delta u(k) \\ \Delta u(k+1) \\ \vdots \\ \Delta u(k+N_c-1) \end{bmatrix},$$

则对系统未来 N_p 步输出的预测为

$$Y_{N_p}(k+1|k) = \mathcal{M}\hat{Y}(k) + \mathcal{S}\Delta U(k), \quad (13)$$

其中:

$$\mathcal{M} = \begin{bmatrix} 0 & 1 & 0 & \cdots & \cdots & \cdots & \cdots & 0 \\ 0 & 0 & 1 & 0 & \cdots & \cdots & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & \cdots & 0 & 1 & 0 & \cdots & 0 \end{bmatrix}_{N_p \times N}$$

$$\mathcal{S} = \frac{1}{F} \begin{bmatrix} s_1 & 0 & 0 & \cdots & 0 \\ s_2 & s_1 & 0 & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ s_{N_c} & s_{N_c-1} & \cdots & s_1 & \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ s_{N_p} & s_{N_p-1} & \cdots & \cdots & s_{N_p-N_c+1} \end{bmatrix}_{N_p \times N_c}$$

3.3 优化及反馈控制(Optimization and feedback control)

本节首先用优化目标函数描述网络拥塞控制的要求, 然后将网络中存在的约束和性能指标转化为优化问题的约束条件, 最后通过求解QP问题给出最优的丢包率控制序列.

AQM算法主要的控制要求就是主动地通过控制量丢包率 p 来通知源端调节发送速度, 使缓冲区的队列长度 q 能尽可能的接近期望值 q_0 . 本文用预测输出 $Y_{N_p}(k+1|k)$ 描述系统的未来状态, 定义控制输出的参考序列为

$$R_e(k+1) = [r(k+1) \quad r(k+2) \quad \cdots \quad r(k+N_p)]^T.$$

由于方程(3)的平衡为原点, 所以 $R_e(k+1)$ 中 $r(k+i) = 0, i = 1, 2, \cdots, N_p$. 通过求 $\|Y_{N_p}(k+1|k) - R_e(k+1)\|^2$ 的最小值使得队列长度 q 尽快接近期望值 q_0 .

另外, 为保证网络的服务质量, 还要尽量避免丢包率变化带来的网络波动, 因此DMAQM算法最小化 $\|\Delta U(k)\|^2$ 来避免网络波动.

综上, 本文中设计的AQM算法要满足两个控制要求: 队列长度接近期望值和避免丢包率变化太大, 而且这两个控制要求是互相矛盾的, RED,PI/PID等算法往往无法同时兼顾这两个要求. 本文中通过加权阵 Γ_y 和 Γ_u 来调节两者之间的权重, 即用下面的优

化方程(14)来描述:

$$\min_{\Delta U(k)} J(y(k), \Delta U(k), N_c, N_p), \quad (14a)$$

$$\begin{aligned} & J(y(k), \Delta U(k), N_c, N_p) = \\ & \|\Gamma_u \Delta U(k)\|^2 + \|\Gamma_y (Y_{N_p}(k+1|k) - R_e(k+1))\|^2. \end{aligned} \quad (14b)$$

丢包率 p 和队列长度 q 是网络中具有实际物理意义的量, 它们的物理约束为: $p \in [0, 1], q \in [0, q_{\max}]$, 其中 q_{\max} 为缓冲区的最大容量, 本文显式地考虑系统实际存在的这些约束:

1) 控制约束: 丢包率能否稳定地在一个固定范围内变化是评价网络性能的一个重要指标. 丢包率的理想值0是控制约束的下限, 丢包率的上限保证网络运行的良好性能, 考虑到系统(1)的平衡点 p_0 , 同时参考RED, PI/PID等算法的控制效果, 将丢包率的上限设为 $2.5p_0$.

2) 输出约束: 缓冲器是一个实际物理设备, q_{\max} 做为输出约束的上限, 输出约束的下限为0.

则整理得到系统(3)的约束为

$$\begin{cases} -1 \leq u(k+i) \leq 1.5, i=0, 1, \dots, N_c-1, \\ -1 \leq y(k+j|k) \leq \frac{q_{\max}}{q_0} - 1, j=1, 2, \dots, N_p. \end{cases} \quad (15)$$

满足约束(15)的DMC优化问题(14)可改写为如下的QP问题描述:

$$\min_{\Delta U(k)} \Delta U(k)^T H \Delta U(k) - G(k+1|k)^T \Delta U(k), \quad (16a)$$

$$\text{s.t. } C_u \Delta U(k) \geq b(k+1|k), \quad (16b)$$

其中:

$$H = S^T \Gamma_y^T \Gamma_y S + \Gamma_u^T \Gamma_u,$$

$$G(k+1|k) = 2S^T \Gamma_y^T \Gamma_y (R_e(k+1) - \mathcal{M}\hat{Y}(k)),$$

$$b(k+1|k) = \begin{bmatrix} u(k-1) - 1.5 \\ \vdots \\ u(k-1) - 1.5 \\ -u(k-1) - 1 \\ \vdots \\ -u(k-1) - 1 \\ \mathcal{M}\hat{Y}(k) - Y_{\max}(k+1) \\ -\mathcal{M}\hat{Y}(k) + Y_{\min}(k+1) \end{bmatrix},$$

$$C_u = \begin{bmatrix} -L \\ L \\ -S \\ S \end{bmatrix}, \quad L = \begin{bmatrix} 10 \dots 0 \\ 11 \dots 0 \\ \vdots \dots \vdots \\ 11 \dots 1 \end{bmatrix}_{N_c \times N_c},$$

$$Y_{\min}(k+1) = \begin{bmatrix} -1 \\ \vdots \\ -1 \end{bmatrix}, Y_{\max}(k+1) = \begin{bmatrix} \frac{q_{\max}}{q_0} - 1 \\ q_0 \\ \vdots \\ \frac{q_{\max}}{q_0} - 1 \end{bmatrix}.$$

根据预测控制的基本原理, 开环控制序列的第一步将作用于被控系统, 因此, 闭环控制律定义为

$$\Delta u(k) = [1 \ 0 \ \dots \ 0] \Delta U^*(k). \quad (17)$$

其中 $\Delta U^*(k)$ 是约束优化问题(14)的解. 将 $\Delta u(k)$ 代回 $\Delta u(k) = u(k) - u(k-1)$, 得到DMAQM算法在 k 时刻的控制量为 $u(k) = \Delta u(k) + u(k-1)$.

4 仿真验证(Simulation verification)

4.1 DMAQM算法参数设置(DMAQM algorithm parameter setting)

网络参数 C, n, T_p, q_0 沿用2.1小节例子中的值, 那么 T_s, N, S, F 相应的取2.2小节中分析得到的值, 另外取 $q_{\max} = 500$ packets. 由于阶跃响应模型(6)的阶次 $N = 34$, 根据经验取预测时域 $N_p = \lfloor N/2 \rfloor = 17$, 为降低计算量则取控制时域 $N_c = 1$.

设定LMI(12)中 $r_0 = 0.65$, 解LMI(12)得估计器增益矩阵 K_f (见附录), 计算得矩阵 $M_{ss} - K_f C_y$ 特征值模的最大值为0.4144, 可见估计器(9)是稳定的, 而且 $0.4144 < r_0 = 0.65$ 满足极点配置条件. 在避免丢包率剧烈震荡的前提下, 笔者希望队列长度尽快到达期望值, 因此优化方程(14)的加权阵取为: $\Gamma_u = 1, \Gamma_y = 1000 \times I_{17 \times 17}$, 其中 I 表示单位矩阵.

注 3 DMAQM算法的在线计算时间主要由 N, N_p 和求解QP问题的速度等决定. 为增加算法的实用可能性, 进一步研究中可以适当增大采样周期来降低系统的阶次(同时增大 Γ_u 限制控制动作的幅度, 避免振荡), 另外将QP求解过程用硬件(例如FPGA)来完成, 提高计算速度.

为验证DMAQM算法的性能, 选择RED算法作为对比算法, 给出了当前广泛使用的参数配置方案^[17]为: $w_q = 0.002, p_{\max} = 0.1, T_{\min} = 80$ packets, $T_{\max} = 320$ packets, $B = 500$ packets.

4.2 无干扰仿真(Simulation without disturbance)

理想的网络环境就是所有的源端发送的数据流都遵循标准的TCP协议, 此时的网络可以认为是无干扰的系统. 在这种理想环境下, 本文进行了DMAQM算法和RED算法的对比实验, 仿真结果由图4给出, 其中水平的点划线分别标出了期望队列长度以及平衡状态下的丢包率.

从图中可以看出, 在满足系统约束的前提下, DMAQM算法能够使系统输出快速稳定地到达期望值, 调节时间比较短, 同时丢包率的超调量明显小

于RED算法,而且丢包率的变化过程比较平稳,有效地避免了网络振荡,而RED算法无法使队列长度稳定到期望值.虽然DMAQM算法的队列长度的超调要明显大于RED算法,但从丢包率曲线中可以看出,队列长度短间接接近上限并没有造成丢包率的剧烈变化而影响网络性能.

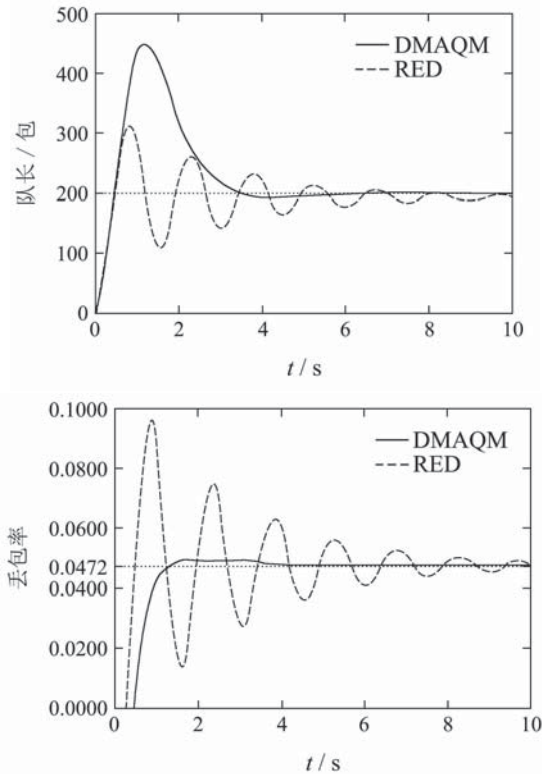


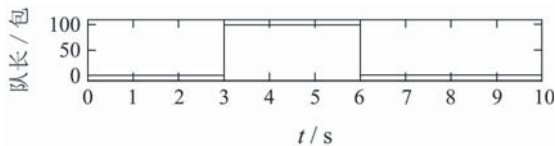
图4 无干扰环境下DMAQM算法与RED算法的队长和丢包率比较

Fig. 4 The comparison of the queue length and packet loss rate in DMAQM and RED without disturbance

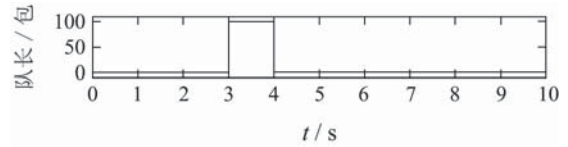
4.3 有干扰仿真(Simulation with disturbance)

实际的网络中存在的多媒体数据流,短流应该视为不可测干扰.这些干扰影响网络的性能,带来不公平性,因此主动队列管理算法应该具有抗干扰,保持公平性等性能.

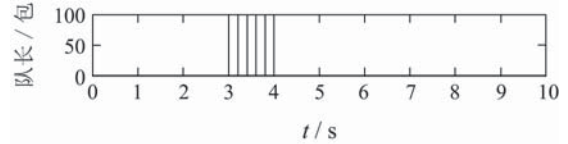
本节分别将图5给出的3种持续时间不同的干扰视为不可测干扰,它们分别代表网络中连接时间长短不同的非响应数据流,通过仿真实验来验证DMAQM算法的抗干扰性能,仿真结果由图6至图8给出.



(a) 长干扰



(b) 短干扰



(c) 瞬时干扰

图5 系统干扰

Fig. 5 System disturbance

图6表明系统存在“长干扰”流时,DMAQM算法抗干扰能力比较弱.但是DMAQM算法能在干扰消失后较快的调节到期望状态,而且队列和丢包率的调节过程都比较平稳. RED算法中的队列长度和丢包率振荡得更加剧烈,在3.6 s~4.2 s时间段内,TCP流的队列长度与期望值的差值大于干扰流的队长,这造成网络资源的更大浪费.

图7表明系统存在“短干扰”流时,DMAQM算法具有较好的抗干扰能力,队列长度比“长干扰”时更加接近期望值.而RED算法的控制效果没有明显改善,而且丢包率的变化范围进一步扩大.

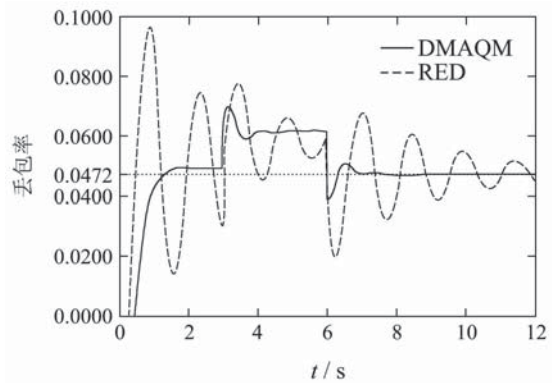
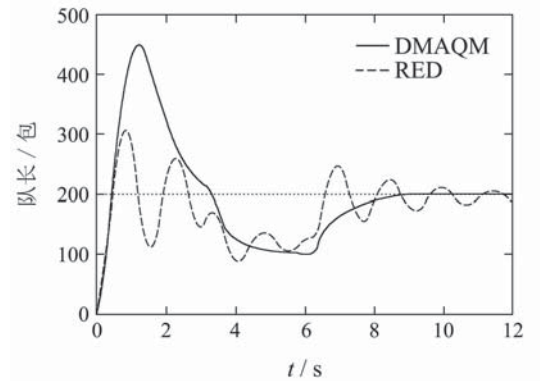


图6 长流干扰环境下DMAQM算法与RED算法的队长和丢包率比较

Fig. 6 The comparison of the queue length and packet loss rate in DMAQM and RED with long disturbance

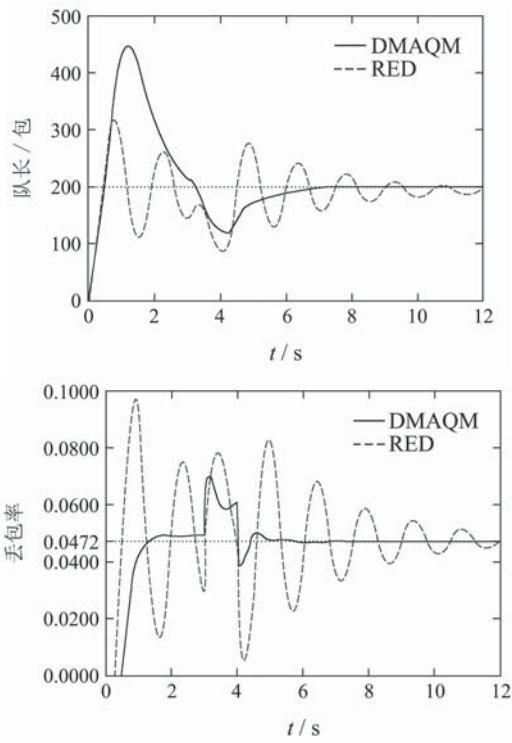


图 7 短流干扰环境下DMAQM算法与RED算法的队长和丢包率比较
 Fig. 7 The comparison of the queue length and packet loss rate in DMAQM and RED with short disturbance

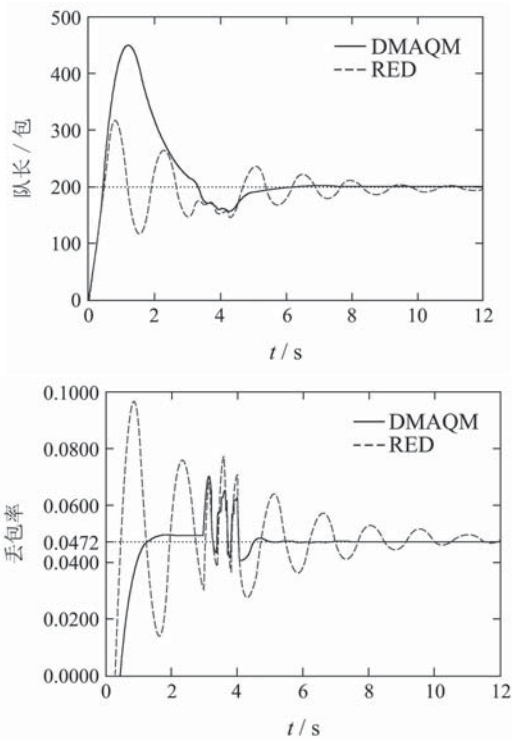


图 8 瞬时流干扰环境下DMAQM算法与RED算法的队长和丢包率比较
 Fig. 8 The comparison of the queue length and packet loss rate in DMAQM and RED with instantaneous disturbance

图8表明系统存在“瞬时干扰”流时, DMAQM算法具有良好的抗干扰能力, 队列长度与期望值的偏差明显减小, 尽管队列有微弱波动. RED算法整体控制效果仍明显劣于DMAQM算法.

综合以上仿真结果, DMAQM算法对持续时间较短的干扰有很好的抗干扰能力, 系统的输出能够较快地达到期望值, 避免了网络的振荡.

4.4 鲁棒性分析(Robustness analysis)

实际网络是一个动态变化的复杂网络, 网络参数 n 和 $R(t)$ 具有很强的不确定性. 本节中 n 和 $R(t)$ 分别在名义值上下25%(点线)和50%(虚线)范围内变化, 并与它们取名义值时(实线)作对比, 检验DMAQM算法的鲁棒性, 仿真中其它网络参数和控制器参数固定不变, 仿真结果如图9所示.

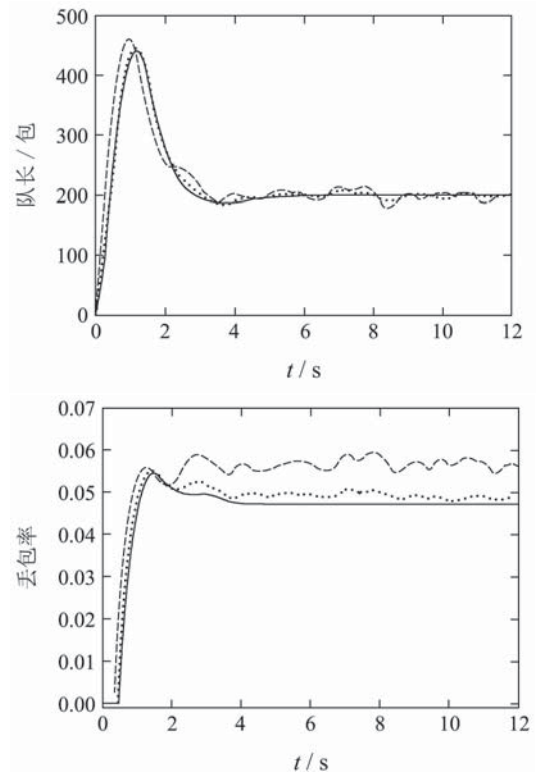


图 9 通过队列长度和丢包率分析DMAQM算法的鲁棒性
 Fig. 9 Robustness analysis of DMAQM by the queue length and packet loss rate

仿真表明, 随着网络参数不确定性的增加, 调节时间增加, 队列长度振荡的幅度在增大, 丢包率增大, 但DMAQM算法仍能使队列长度保持在期望值附近, 而且丢包率的振荡过程也比较平缓.

5 结论(Conclusion)

在网络的中间节点引入AQM机制, 可以更好地解决网络拥塞, 保证服务质量. 但是目前大多数AQM策略不能对网络性能进行优化设计, 而且控制参数缺乏自适应性. 本文基于约束模型预测

控制理论,提出了一种新的主动队列管理算法——DMAQM算法.它利用预测控制显式地处理约束,通过滚动优化提高了系统的自适应性和鲁棒性.仿真表明DMAQM算法使瞬时队列保持稳定,且较快速地到达期望值,算法的调节时间短,丢包率变化较小,另外它有较好的抗瞬时干扰的能力,对负载和延迟存在不确定性的网络具有较好的鲁棒性.

参考文献(References):

- [1] BRADEN B, CLARK D, CROWCROFT, et al. Recommendations on queue management and congestion avoidance in the Internet[DB/OL]. 1998. <http://www.faqs.org/rfcs/rfc2309>, 2005-09-11.
- [2] MISRA V, GONG W B, TOWSLEY D. Fluid-based analysis of a network of AQM routers supporting TCP flows with an application to RED[C] // *Proceedings of the Conference on Applications, Technologies, Architectures, and Protocols for Computer Communication*. Stockholm, Sweden: [s.n.], 2000: 151-160.
- [3] HOLLOT C, MISRA V, TOWSLEY D. On design improved controllers for AQM routers supporting TCP flows[C] // *Proceedings of the IEEE Infocon*. Anchorage, Alaska: [s.n.], 2001: 1726-1734.
- [4] KIM K B. Design of feedback controls supporting TCP based on the state-space approach[J]. *IEEE Transactions on Automatic Control*, 2006, 51(7): 1086-1098.
- [5] 钱艳平, 李奇, 刁翔. 预测PI时滞网络拥塞控制算法设计及性能分析[J]. *控制理论与应用*, 2006, 23(2): 161-168.
(QIAN Yanping, LI Qi, DIAO Xiang. Design and analysis of predictive PI algorithm for congestion control in time-delay network[J]. *Control Theory & Applications*, 2006, 23(2): 161-168.)
- [6] CHEN C K, KUO H H, YAN J J, et al. GA-based PID active queue management control design for a class of TCP communication networks[J]. *Expert Systems with Applications*, 2009, 36(2): 1903-1913.
- [7] MANFREDI S, BERNARDO M D, GAROFALO F. Reduction-based robust active queue management control[J]. *Control Engineering Practice*, 2007, 15(2): 177-186.
- [8] LEE J H, MORARI M. Model predictive control: past, present and future[J]. *Computers and Chemical Engineering*, 1999, 23: 667-682.
- [9] 席裕庚. 预测控制[M]. 北京: 国防出版社, 1993.
(XI Yugeng. *Predictive Control*[M]. Beijing: National Defence Industry Press, 1993.)
- [10] 陈虹, 刘志远, 谢小华. 非线性模型预测控制的现状与问题[J]. *控制与决策*, 2001, 16(4): 385-391.
(CHEN Hong, LIU Zhiyuan, XIE Xiaohua. Nonlinear model predictive control: the state and open problems[J]. *Control and Decision*, 2001, 16(4): 385-391.)
- [11] LIU G P, MU J X, REES D, et al. Design and stability analysis of networked control systems with random communication time delay using the modified MPC[J]. *International Journal of Control*, 2006, 79(4): 288-297.
- [12] 朱瑞军, 索东海, 马吉荣. ATM网络预测拥塞控制器设计[J]. *控制与决策*, 2004, 19(1): 61-64.
(ZHU Ruijun, SUO Donghai, MA Jirong. Predictive congestion controller in ATM networks[J]. *Control and Decision*, 2004, 19(1): 61-64.)
- [13] ZHANG X H, ZOU K S, CHEN Z Q, et al. Stability analysis of AQM algorithm based on generalized predictive control[C] // *Proceedings of the 4th International Conference on Intelligent Computing*. Shanghai, China: [s.n.], 2008: 1242-1249.
- [14] AUFDERHEIDE B, BEQUETTE B W. Extension of dynamic matrix control to multiple models[J]. *Computers and Chemical Engineering*, 2003, 27(11): 1079-1092.
- [15] CUTLER C R, RAMAKER B L. Dynamic matrix control—a computer control algorithm[C] // *Proceedings of the Joint Automatic Control Conference*. San Francisco, CA: [s.n.], 1980.
- [16] CHILALI J H, GAHINET P. H_∞ design with pole placement constraints: an LMI approach[J]. *IEEE Transactions on Automatic Control*, 1996, 41(3): 358-367.
- [17] 翟双. 网络中基于路由器的拥塞控制策略研究[D]. 长春: 吉林大学, 2002.
(ZHAI Shuang. *A study of congestion control based on the router in the internet*[D]. Changchun: Jilin University, 2002.)

附录(Appendix)

阶跃响应系数:

$S =$

$$\begin{bmatrix} 0 & 0 & 0 & -0.0004 & -0.0034 & -0.0085 & -0.0145 \\ & -0.0210 & -0.0275 & -0.0339 & -0.0399 & -0.0455 \\ & & -0.0507 & -0.0554 & -0.0597 & -0.0636 & -0.0671 \\ & & & -0.0703 & -0.0731 & -0.0756 & -0.0779 & -0.0799 \\ & & & & -0.0817 & -0.0833 & -0.0848 & -0.0861 & -0.0872 \\ & & & & & -0.0882 & -0.0891 & -0.0899 & -0.0906 & -0.0913 \\ & & & & & & -0.0919 & -0.0924 \end{bmatrix}.$$

估计器增益矩阵:

$K_f =$

$$\begin{bmatrix} 0.8668 & 0.8690 & 0.8711 & 0.8719 & 0.8723 & 0.8725 & 0.8725 \\ & 0.8726 & 0.8726 & 0.8726 & 0.8726 & 0.8726 & 0.8726 & 0.8726 \\ & & 0.8726 & 0.8726 & 0.8726 & 0.8726 & 0.8726 & 0.8726 \\ & & & 0.8726 & 0.8726 & 0.8726 & 0.8726 & 0.8726 \\ & & & & 0.8726 & 0.8726 & 0.8726 & 0.8726 \\ & & & & & 0.8726 & 0.8726 & 0.8726 \end{bmatrix}.$$

作者简介:

王萍 (1982—), 女, 博士研究生, 研究方向为预测控制、通信网数据流的优化控制, E-mail: wangping08@mails.jlu.edu.cn;

陈虹 (1963—), 女, 教授, 博士生导师, 主要从事预测控制、鲁棒控制及非线性控制的理论与应用研究, E-mail: chen@jlu.edu.cn;

杨晓萍 (1963—), 女, 博士, 副教授, 硕士生导师, 主要研究方向为通信网数据流的优化控制、网络服务质量, E-mail: yxp@jlu.edu.cn.