

文章编号: 1001-0920(2003)05-0545-05

网络控制系统在线时延估计控制

魏震, 李长虹, 谢剑英
(上海交通大学自动化系, 上海 200030)

摘要: 从网络控制系统的实际应用出发, 在不附加网络同步时钟和对时延特征的离线假设下, 运用通讯技术中的网络协议对网络时延进行在线估计; 在此基础上, 运用最优控制为网络控制系统设计控制器。通过基于 CAN 总线的实验系统所得结果验证了上述设计的有效性。

关键词: 网络控制系统; 在线时延估计; ADW 方法; LQ 控制; CAN 总线

中图分类号: TP273.5 文献标识码: A

Online delay-evaluation control for networked control systems

WEI Zhen, LI Chang-hong, XIE Jian-ying

(Department of Automation, Shanghai Jiaotong University, Shanghai 200030, China)

Abstract: An online delay-evaluation approach, namely average delay window (ADW) is introduced for networked control systems, which can characterize the time delays without any synchronized clock in the network and prior assumptions about time delays. Integrated with ADW approach, an optimal control scheme is designed to achieve desired performance despite the uncertain delays in the system. The experimental results based on CAN bus illustrate the effectiveness of the proposed control design and satisfactory performance of the closed-loop system.

Key words: Networked control systems; Online delay evaluation; ADW approach; LQ control; CAN bus

1 引言

网络控制系统(NCS)是指一类分布式控制系统。其中:控制回路中的传感器、执行器和控制器之间通过串行网络连接,控制系统中的状态变量和控制变量以数字量的形式在网络中传递。

网络时延是NCS研究的主要方面,它不仅涉及对控制理论的讨论,而且与网络通讯技术密切相关。本文的基本出发点是将这两方面结合起来,用于NCS的研究。

在对NCS进行具体讨论之前,需要对其的工作模式进行必要的假定。在此采用Nillsen^[1]假设,即控制回路中的传感器节点是时间驱动的,其周期设为 h ;控制器节点和执行器节点是事件驱动的,即由

相关的数据到达来激发节点的相应行为。

一般NCS的结构及其时间延迟的产生如图1所示。其中: τ^e 为传感器节点向控制器节点传输的数据所产生的时延, τ^{sc} 为控制器节点向执行器节点传输的数据所产生的时延, τ 为控制器节点算法执

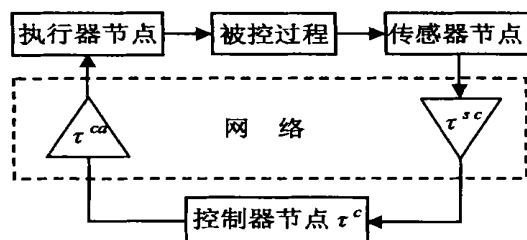


图1 一般NCS结构

收稿日期: 2002-03-25; 修回日期: 2002-05-27。

作者简介: 魏震(1976—),男,上海人,博士,从事分布式实时控制的研究; 谢剑英(1940—),男,上海人,教授,博士生导师,

行所产生的时延。一般情况下, τ 可以忽略不计; 但在某些情况下, τ 会对控制系统的品质产生不可忽略的影响^[2]。这里遵从前一种认识, 即 τ 不计。由此, 回路中由于网络所产生的总时延为 $\tau = \tau^c + \tau^a$ 。NCS 一般以采样控制为基础, 以 k 表示采样时刻。在第 k 个采样周期内, 回路时延描述为 $\tau_k = \tau_k^c + \tau_k^a$, 且假设满足条件 $\tau_k < h$ 。

一般文献是在假设网络中存在同步时钟的情况下, 事先离线设定网络时延的特性, 如文献[1, 3]。这种假设在特定情况下成立, 并不具有一般性。首先, 一般假设中的网络时钟同步, 并不是所有的网络技术都能支持, 如 Device Net 就不存在同步时钟; 其次, 网络中参与数据传输的节点个数和数据任务都是变化的, 不可能事先准确地对某一类数据的时延特性作出离线假设。因此在实际的网络应用中, 有必要对数据的网络时延进行准确的在线估计。

NCS 中的网络技术可为在线时延估计提供支持, 这一点在以往的研究中没有得到足够的重视, 而本文则利用网络技术对如何获得时延信息进行讨论。首先阐述一种在线时延分析方法, 在具体实现时, 它可被设计为网络会话层协议功能的一部分; 然后介绍建立在获得时延信息基础上的最优控制方法; 最后利用基于 CAN 总线的实验系统来验证上述设计。

2 在线时延估计方法

NCS 中的网络可分为两大类: 令牌网络和 CAN 网络, 它们都称为实时网络, 即网络中数据的最大时延是有界的。令牌网络中数据传输的网络时延可看作常值, 它们的值可由离线分析得到, 因此可将这类系统模型抽象为定常时延系统, 这类系统的控制器设计较为容易^[4]。CAN 网络中的数据传输时延是时变的, 本文主要对这种时延时变系统展开讨论。

下面描述一种时延估计方法, 即平均时延窗口 (ADW)。这种方法不需要网络同步时钟的存在, 而是借助于独立于具体某种网络且易于实现的网络技术。

2.1 连接和数据优先级

NCS 中的传感器节点与控制器节点以及控制器节点与执行器节点, 在进行数据传输之前都需要建立网络连接。在此过程中, 节点之间的重要信息进行相互交流。在下面的时延估计中, 需要控制器节点能通过网络的连接过程获知传感器的采样周期 h 。

一般网络中的每个数据帧都有相应的优先级。CAN 中数据帧的标识即代表其优先级, 优先级高的

数据帧在同样情况下将先于优先级低的数据帧发送。如果把数据的网络时延看作随机过程, 则有关系 $E\{\tau(\text{高优先级数据})\} < E\{\tau(\text{低优先级数据})\}$, 其中 E 代表取期望值。如果两类数据帧 D_1 和 D_2 的优先级相同或足够接近, 则可推断

$$E\{\tau(D_1)\} = E\{\tau(D_2)\} \quad (1)$$

因此可以认为, 数据帧的优先级决定了它的时延特征。

2.2 一种在线时延估计方法 —— 平均时延窗口

这种方法的基本思路是: 通过建立网络连接, 使得控制器能获得传感器的采样周期 h , 把它用作对 τ_k^c 的估计; 通过预先在连接过程中设定的数据帧的优先级, 使得 τ_k^a 的特征与 τ_k^c 一致, 由此可以得到 τ_k 的特征。

τ_k^c 的特征可通过设置在控制器中的一个时延窗口获得, 此窗口存储对时延的当前和历史估计值 $\{\tau_{k-M+1}^c, \dots, \tau_k^c\}$ 共 M 项。时延估计方法如下所述:

设控制器接受第 k 个传感器数据帧的时钟时刻为 T_k^c , 此时钟值是控制器自身的逻辑时钟。由于控制器已获知传感器的采样周期 h , 则可预计第 $(k + 1)$ 个采样数据的达到时刻为

$$T_{k+1}^c = T_k^c + h$$

事实上, 第 $(k + 1)$ 个采样数据的达到时刻为 T_{k+1}^c , 它与 T_{k+1}^c 之间可能存在差异。在 τ_k^c 已知的情况下, τ_{k+1}^c 可由以下算法得出。

$$\begin{aligned} &\text{If } (\tau_k^c + (T_{k+1}^c - T_k^c)) > 0 \\ &\text{Then } \tau_{k+1}^c = \tau_k^c + (T_{k+1}^c - T_k^c) \quad (*) \\ &\text{Else} \quad \quad \quad (**) \end{aligned}$$

$$\tau_{k+1}^c = 0$$

$$\text{If } k = M - 1$$

$$\begin{aligned} &\text{Then } \tau_k^c = \tau_k^c + (T_{k+1}^c - T_k^c) \\ &\quad i = k - M, \dots, k \end{aligned}$$

Else

$$\begin{aligned} &\tau_k^c = \tau_k^c + (T_{k+1}^c - T_k^c) \\ &\quad i = 1, 2, \dots, k \end{aligned}$$

End

初始化时, 时延窗口中的所有值均为零。在没有历史数据信息的条件下, 设 $\tau^c = 0$ (这一点将在下面讨论)。以此为基础, 时延估计过程便可由上述算法进行下去。

此方法的基本过程是: 当 (*) 成立时, 把 τ_k^c 作为推断 τ_{k+1}^c 的基础是合理的。当 $T_{k+1}^c < T_{k+1}^c$ 时, 说明第 $(k + 1)$ 个采样数据经历了不少于第 k 个采样

数据的时延,因此需要把这部分时延记入 τ_{k+1}^c ; 当 $T_{k+1}^c < T_{k+1}^c$ 且 $(*)$ 成立时,说明第 $(k+1)$ 个采样数据经历了少于第 k 个采样数据的时延,计算 τ_{k+1}^c 时,需从 τ_k^c 中减去相应的差值。当 $(*)$ 不成立,即 $(\tau_k^c + (T_{k+1}^c - T_{k+1}^c)) < 0$ 时,算法将执行 $(**)$ 以下的部分。这种情况说明过去的时延估计忽略了部分时延,因此需要把这些忽略的时延加到历史时延数据中去,同时设定 $\tau_{k+1}^c = 0$, 把它作为下一次估计的基础。

通过在时延窗口中的 M 个时延历史数据,可以在线描述出 τ^c 的随机分布规律。假设 τ^a 与 τ^c 无关,则通过前述设置数据优先级的方法,使得 τ^a 的随机分布规律与 τ^c 相近。这样,在没有网络同步时钟和时延离线假设的情况下,可获得相关时延信息。根据这些信息,已有文献中的控制方法便可加以运用,如文献[1]中的最优 LQG 设计等。

最简单的处理方法是取时延窗口中数据的平均值用于计算控制输出,同时考虑 τ^a 与 τ^c 的关系, τ_k 可定义为 $\tau_k = 2\tau_k^c$, 其中

$$\tau_k^c = \begin{cases} \tau_i^c / k, & k < M \\ \tau_i^c / M, & k = M \end{cases} \quad (2)$$

2.3 ADW 方法讨论

本节对 ADW 中的一些要点进行讨论。

1) 时延窗口的参数 M 是一个重要参数,它指明所包含时延历史数据的多少。为了更好地描述时延的随机特征, M 的取值不宜太小。另一方面,由于时延的时变性,控制输出的计算往往只需要最近的时延信息,因此 M 的取值也不能过大。极端情况下,设置 M 值为 1,即时延窗口只有最近采样数据的时延值。

2) 在前述 ADW 方法中,假设 $\tau^c = 0$, 它往往与实际情况不相符合。为了得到准确的 τ^c 值,可借助于网络技术来实现。在控制回路工作之前,传感器节点向控制器节点发送一个特别的数据帧,此数据帧具有较高的优先级,因此它所经历的网络时延较小。假设这个帧到达控制器的时刻为 T_0^c , 则第 1 个传感器数据帧的期望到达时刻为 $T_1^c = T_0^c + h_d$, h_d 可以不等于 h , 它也是控制器节点在网络连接阶段获得的信息。这样, τ^c 便可由上述 ADW 算法推断得出。

3) 在实际中, τ^a 的信息可以通过相关硬件设备的支持获得,如 CAN 控制器 SJA 1000 提供一个中断信号,用于指示数据发送完成。因此,可设置另一

个时延窗口用于描述 τ^a 的特征。需要特别指出的是,这个时延窗口不包括当前的时延值,即在第 k 个采样周期,窗口中最近的时延值为 τ_{k-1}^a , 这也是因果系统的必然结果。所以,此时 τ^a 的值必须通过窗口中的记录加以推断,把它记为 $\tilde{\tau}_k^a$ 。

设 M_1 代表这个时延窗口的大小,对 $\tilde{\tau}_k^a$ 的推断如下

$$\tilde{\tau}_k^a = \begin{cases} \tau_i^a / M_1, & k = M_1 + 1 \\ \tau_i^a / (k - 1), & 1 < k < M_1 \\ \tau_k^c, & k = 1 \end{cases} \quad (3)$$

由此,整个控制回路的时延为

$$\tau_k = \tilde{\tau}_k^a + \tau_k^c \quad (4)$$

4) 在实现 ADW 方法对网络时延进行估计时,可将它所完成的功能集成在网络协议层次结构中的会话层加以实现。

3 基于网络时延的最优控制

在获得在线的时延估计 τ_k 后,便可将它结合到 NCS 的控制器中。在此采用时延系统的最优控制器设计方法来实现。

不考虑噪声,假设被控对象模型为

$$\begin{cases} dx/dt = Ax(t) + Bu(t) \\ y(t) = Cx(t) \end{cases} \quad (5)$$

其中: $x(t) \in R^n$, $u(t) \in R^m$, $y(t) \in R^p$; A , B 和 C 为具备相应维数的矩阵。考虑 NCS 中的网络时延 τ_k , 在条件 $\tau_k < h$ 满足的情况下,离散系统模型为

$$\begin{cases} x_{k+1} = \Phi x_k + \Gamma_0(\tau_k) u_k + \Gamma_1(\tau_k) u_{k-1} \\ y_k = Cx_k \end{cases} \quad (6)$$

对某一特定的 τ_k , 在全状态反馈条件下, LQ 最优控制器为

$$u_k = -L(\tau_k) \begin{bmatrix} x_k \\ u_{k-1} \end{bmatrix} \quad (7)$$

由于 τ_k 是时变的,理想情况下,控制器需要根据每次“采样-控制”过程得到的 τ_k , 在线计算出对应的 $L(\tau_k)$ 。这样的过程要求控制器要有较高的运算速度,这对于广泛应用的以单片机作为控制器处理器的系统是不适合的。实际应用中,在满足一定的系统控制品质的条件下,可用如下分段函数来替代实际所得的 τ_k , 即

$$f(\tau_k) = \begin{cases} \delta_1, & 0 < \tau_k < \alpha \\ \vdots & \vdots \\ \delta_N, & \alpha_{N-1} < \tau_k < h \end{cases} \quad (8)$$

由此,只需在控制器中保留一张存有 N 组与 $f(\tau_k)$ 相对应的参数 $L(\tau_k)$ 的表格,控制器所要完成的工作就是根据每次获得的 τ_k , 查找这张表得到 $L(\tau_k)$, 从而由式(3) 算出控制输出 u_k 。

4 实验研究

本节介绍运用 ADW 在线时延估计并结合基于时延的最优控制的实验及其结果。实验中的网络为 CAN 总线,整个实验体系结构在文献[5] 中已有详细描述。实验系统中没有真实的被控对象,因此通过其中一个节点的运行仿真程序来模拟被控对象。CAN 总线的数据传输率设为 10 kbps。

考虑被控模型

$$\frac{dx}{dt} = \begin{bmatrix} 0 & 4 \\ 1 & 0 \end{bmatrix} x + \begin{bmatrix} 1 \\ 0 \end{bmatrix} u$$

$$y(t) = [0 \ 1]x$$

传感器的采样周期为 500 ms,即 $h = 500$ ms。设计 LQ 控制器时,性能指标函数取作

$$J = \sum_{k=0}^{\infty} (y_k^2 + u_k^2)$$

在 NCS 网络中,传输的数据任务大都为周期性释放的,它们的特征可用二元组 (T, C_r) 来表示。其中: T 代表数据任务的释放周期; C_r 代表在每个数据释放周期中,为完成此次数据发送任务所需占用网络资源的时间。传感器发送的模型状态数据的特征可表示为 $(500, 10)$, 记为 Q_1 ; 控制量输出数据特征可表示为 $(500, 7.8)$, 记为 Q_2 。除了与控制回路有关的 Q_1 和 Q_2 外,实验中还涉及另外两个数据发送任务 $(100, 30)$ 和 $(150, 80)$, 分别记为 Q_3 和 Q_4 。它们用于模拟网络中其他数据任务,其数据帧优先级都高于 Q_1 和 Q_2 。可以验证,即使网络中存在 Q_3 和 Q_4 , Q_1 数据帧的网络时延仍然满足 $\tau_k < h^{[6]}$ 。

为了满足系统输入-输出无静差,实验中在最优控制器的前端设置了一个积分器,用于消除静差。

实验 1 不考虑网络时延设计 LQ 控制器。在 Q_3 和 Q_4 任务没有参与网络活动的情况下,系统的阶跃响应如图 2 所示,可以看出,此时的控制品质是令人满意的。当 Q_3 和 Q_4 任务释放后,控制器在 4 次相同实验下的系统阶跃响应如图 3 所示,此时系统呈现出时变且不稳定的状态。

实验 2 运用 ADW 方法设计基于网络时延的最优控制器。其中的参数设置如下:由 2.3 节的讨论可知 $M = 1$ 是合适的;取 $M_1 = 2$; N 的取值越大,越接近于由模型(4) 所得的控制输出结果,实验中设 $N = 3$ 。控制器相关参数设置如表 1 所示。在 Q_3 和 Q_4

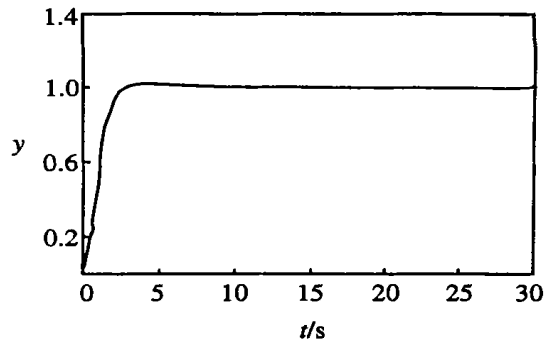


图 2 Q_3 和 Q_4 未参与网络活动的系统阶跃响应

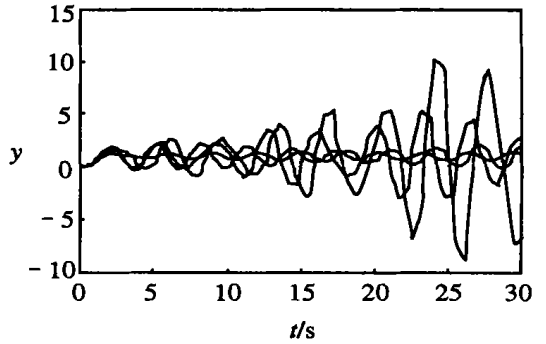


图 3 Q_3 和 Q_4 任务释放后的系统阶跃响应

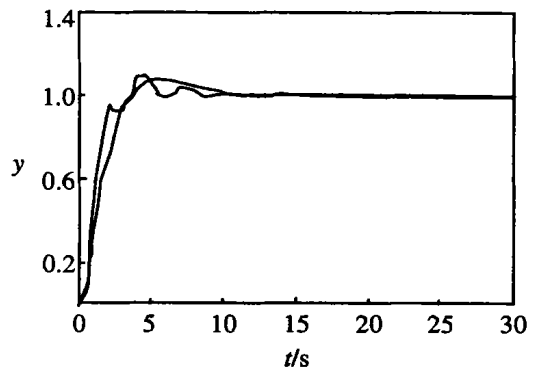


图 4 基于网络时延的最优控制阶跃响应

表 1 控制器参数

N	τ_k / ms	δ_N / ms	LQ 参 数	
1	τ_k	50	0	2.747 0, 5.506 1, 0
2	$50 < \tau_k$	150	100	3.356 3, 6.722 5, 0.304 1
3	$\tau_k > 150$	240		4.442 0, 8.891 6, 0.846 5

任务释放的情况下,实验重复两次,此时的响应曲线如图 4 所示,它在一定范围内仍是时变的,但控制品质却是令人满意的。

实验 3 改变实验 2 中的参数 N 和 M_1 , M 仍为 1,重复实验 2 的过程。控制器的参数如表 2 所示,典型的实验结果如图 5 所示。当 $N = 2$ 和 $N = 5$ 时,图中表达的事实是:随着 N 的增大,基于时延的控制

更加接近于依据时延的连续控制,因此控制效果也更好。

表 2 $N = 5$ 时的控制器参数

N	τ_k/ms	δ_N/ms
1	$\tau_k = 50$	0
2	$50 < \tau_k < 100$	75
3	$100 < \tau_k < 150$	125
4	$150 < \tau_k < 200$	175
5	$\tau_k > 200$	240

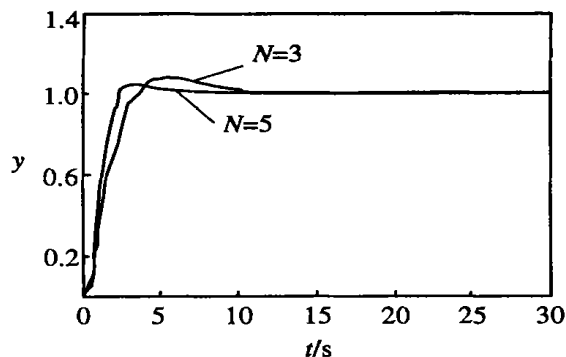


图 5 不同 N 值的实验结果

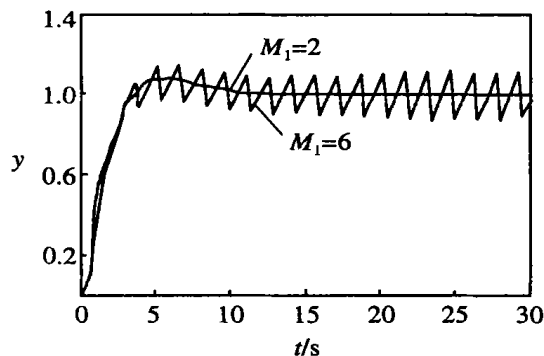


图 6 不同 M_1 值的实验结果

控制器参数不变, N 仍取 3, $M_1 = 2, M_1 = 6$, 所得控制曲线如图 6 所示。可见系统的品质有所降低, 这是因为大的时延估计窗口会降低对控制器至执行器之间时延估计的精确度。

5 结 语

本文从网络控制系统的实际应用背景出发, 在不附加网络同步时钟和对时延特征的离线假设的情况下, 运用通讯技术中的网络协议对网络时延进行在线估计; 在此基础上, 运用最优控制为网络控制系统设计控制器; 最后通过基于 CAN 总线的实验系统所得结果, 验证了上述设计的有效性。

参考文献(References):

- [1] Nilsson J. Real-time control systems with delays [D]. Sweden: Lund Institute of Technology, 1998.
- [2] Cervin A. Improved scheduling of control tasks [A]. Proc 11th Euromicro Conf Real-time Systems [C]. York, 1999. 4-10.
- [3] Fengli Lian. Analysis, design, modeling and control of networked control systems [D]. Michigan: The University of Michigan, 2001.
- [4] Astrom K J, Wittenmark B. Computer-controlled Systems [M]. Third Edition. New Jersey: Prentice Hall, 1997.
- [5] 魏震, 沈钢, 谢剑英. 基于 CAN 的分布式实时系统仿真平台 [J]. 系统仿真学报, 2002, 14(5): 534-537.
(Wei Zhen, Shen Gang, Xie Jianying. The real-time distributed simulation platform based on CAN bus [J]. J System Simulation, 2002, 14(5): 534-537.)
- [6] Tindell K, Burns A, Welling A J. Calculating controller area network message response times [J]. Control Engineering Practice, 1995, 3(8): 1163-1169.

第五届“何潘清漪优秀论文奖”征文启事

“何潘清漪优秀论文奖”征文活动自 1991 年开展以来, 现已举办了四届, 共评选出获奖论文 9 篇。第五届“何潘清漪优秀论文奖”评选活动即将开始, 只要在全球各地用中文发表的关于 DEDS 方面的论文都有资格参加。授奖仪式拟于 2004 年 6 月在第五届全球智能控制与自动化大会上举行。

论文要求: 1. 文章必须是用中文正式发表过的; 2. 来稿应是该文在所发表刊物的抽印页或复印页, 一式五份; 3. 来稿请在首页左上方注明“何潘清漪优秀论文奖征文”字样。

截稿日期: 2004 年 3 月 31 日

联系地址: 510640 广州市五山华南理工大学