基于时间触发的CAN协议任务调度优化算法*)

朱智林[1,2] 刘晓华[1] 韩俊刚[3]
(西安电子科技大学软件研究所 西安 710071)[1] (山东工商学院计算机系 烟台 264005)[2]
(西安邮电学院计算机系 西安 710061)[3]

摘 要 在基于时间触发的CAN协议的实时分布系统中,时间触发的CAN协议的任务调度是一个重要的问题。一种多项式时间复杂度的优化算法被提出来构造调度表,其基本周期有4种假定情况,其中在实验中给出了每种算法的渐近性能比,并证明折衷周期算法的性能要优于其它三种算法。

关键词 TTCAN,调度,优化算法,渐近性能比

Optimal Algorithms for Scheduling Based-on the Time Triggered CAN Protocol

ZHU Zhi-Lin[1,2] LIU Xiao-Hua[1] HAN Jun-Gang[3]
(Institute of software, Xidian University, Xi'an 710071)[1]
(Department of Computer, Shangdong Institute of Business and Technology, Yantai 264005)[2]
(Department of Computer, Xi'an Institute of Post and Telecoms, Xi'an 710061)[3]

Abstract Time-Triggered CAN protocol task scheduling is an important problem in real-time distributed systems, which are based on Time-trggered CAN protocol. An optimal algorithm, which has polynomial time complexity, is proposed to construct scheduling tables. For the algorithm, four strategies to determine basic cycle(BC)are presented. Experiment results of the algorithm are given in different problem size and data precision. Asymptotic performance ratio of the algorithm in different BC strategies is shown in detail.

Keywords TTCAN, Scheduling, Optimal algorithm, Asymptotic performance ratio

1 引言

现场总线CAN(Controller Aera Network)应用于实时分布式嵌入式系统中已经有数年的历史了。目前基于时间触发的通信系统模式引起了业界的广泛兴趣,从而大大促进基于时间触发的CAN(TTCAN:Time Triggered CAN)[1,2]的标准化进程。在基于时间触发的CAN系统中,需要对系统中要完成的信息帧集合传输任务设计一个合理的任务调度表,从而实现在实际通信过程中以时间作为信息帧传输的触发器,按照预先设计好的任务调度表完成信息传输任务。文[3,4]中的算法着重研究了任务调度中的负载均衡问题。本文从缩短调度周期的角度出发,提出了一种新的优化调度算法,分析了算法的性能,并给出了相应的仿真结果。

本文第2节介绍基于时间触发的CAN协议以及相关知识,第3节提出了构造调度表一个优化算法,第4节给出了该算法的实验结果。

2 基于时间触发的CAN协议相关知识

2.1 基于时间触发的CAN协议

CAN目前已经成为通信时间是系统关键的制造业界的主流现场总线,CAN的操作遵守事件触发的规范。基于时间触发的CAN是ISO制定的采用时间触发机制的基于CAN的新的通信协议标准,该标准的编号是ISO11898-4,它是对CAN的一种扩展。在实时系统中,相对于事件触发而言,采用时间触发机制,可以提高通信带宽的利用率。

在基于时间触发的CAN系统中,有一个特殊的节点——时间-主机节点,它负责全局的时钟同步,并发送同步信息帧,实现网络中各个节点的同步,同时它也是一个基本周期BC(Basic Cycle)的起始标志。一个BC可以划分为多个时隙,各个时隙的大小不固定,每个时隙开始的时刻由同步信息帧定义。系统中的数据信息帧分别安排在不同的时隙进行传输,一旦时标到达某一时隙的开始时刻,立即触发在该时隙上数据信息帧进行传输。每个数据信息帧都必须在一个时隙内完成传输,以保证数据信息帧传输的时间独立性。

2.2 实时分布式系统的通信特点

在汽车电子控制系统、卫星的星上控制系统、医疗设备的电子控制系统以及纺织设备等典型的实时分布式系统中,需要对被控对象的当前状态进行周期性的数据采集,然后依此采取进行相应的控制措施。例如在小卫星的星上系统中,需要对有效载荷、姿态、电源等设备进行例行的数据采集,又如在汽车的电子控制系统中,要对当前的转速、油压、气压、速度、温度、刹车等状态进行实时采集,等等诸如此类。这些系统中,例行的数据采集是周而复始地不断重复进行的,它们是系统正常工作的一个重要组成部分,一般而言这类数据采集方式是典型的时间触发的通信模式,在系统出现异常时,通常采用事件触发的通信模式,发送相应的控制指令来及时纠正错误,排除故障,保证系统安全。

在时间触发的通信模式中,例行的数据通信的一个循环

*)国家自然基金:No. 90207015。朱智林 博士生,副教授,主要研究方向为软件理论,面向对象技术,嵌入式系统设计;刘晓华 博士生,副教授;韩俊刚 教授,博导。

称为一个调度周期，它的大小是固定的，一般是由多个 BC 构成。数据信帧的传输时段安排表称之为任务调度表，它由多个 BC 行组成，每行最少有一个时隙，因而任务调度表也可称为系统矩阵 SM(System Matrix)。一般典型的实时分布式系统中，一个调度周期中要传输的数据信息帧很多，采用手工的方式构造既费时，又无法达到优化的目的。要改善实时控制系统的性能，构造一个最优的 SM 是实时分布式系统的关键。

3 构造 SM 算法

3.1 问题描述

设系统要传输的信息帧集合 L 由 n 个任务组成，$L=\{a_1, a_2, \cdots, a_n\}$，任务 a_i 表示信息帧 i，$s(a_i)$表示任务 i 的执行时间(即传输信息帧 i 所需要的时间)。

系统任务调度周期记为 T_{SM}，$T_{SM}=< BC_1, BC_2, \cdots, BC_m>$，即 T_{SM} 分别由 m 个时间片断大小为 BC 的 BC_1，BC_2，…，BC_m 基本周期构成。

在任意 BC_i 上安排的任务记为 π_i，$Level(BC_i)$为 BC_i 周期内任务的实际持续时间，具体描述如下，其中 $1\leqslant i\leqslant m$。

$\pi_i=(a_i^1, a_i^2, \cdots, a_i^k)$，且 $\sum_{j=1}^{k} s(a_i^j)\leqslant BC_i$，$a_i^j\in\pi_i$，$\pi_i\subset L$

$Level(BC_i)=\sum_{a\in BC_i} s(a)$，且 $Level(BC_i)\leqslant BC$

若系统的调度周期（即 SM 的持续时间)记为 T，则

$$T=\sum_{i=1}^{m} Level(BC_i)$$

3.2 构造 SM 的基本思想

(1) 任务集 L 分类　将 L 集合中划分为 p 个子集，即：$L=\{C_1, C_2, \cdots, C_P\}$，其中：

$C_1=\{a_1^1, a_1^2, \cdots, a_1^{k1}\}$，$n\geqslant k1\geqslant 1$

$C_2=\{a_2^1, a_2^2, \cdots, a_2^{k2}\}$，$n\geqslant k2\geqslant 1$

……

$C_p=\{a_p^1, a_p^2, \cdots, a_p^{kp}\}$，$n\geqslant kp\geqslant 1$

且对任意 i，$p\geqslant i\geqslant 1$，均满足

$s(a_i^j)=s(a_i^k)$，$1\leqslant j, k\leqslant ki$，$j\neq k$

(2)确定 SM 基本周期

- 最小周期：$Min_{BC}=MAX\{ s(a_i) \mid a_i\in L, 1\leqslant i\leqslant n\}$
- 最大周期：$Max_{BC}=\sum MAX\{ s(a_i) \mid a_i\in C_i, 1\leqslant i\leqslant p\}$
- 平均周期：$Avg_{BC}=(Max_{BC}+Min_{BC})/2$
- 折衷周期：合并任务 L 分类集中仅有一个元素的集合 C_i($p\geqslant i\geqslant 1$)，得到一个新分类集，记为：$L=\{C_1, C_2, \cdots, C_d\}$(其中 $p\geqslant d\geqslant 1$)。

$Ex_{BC}=\sum MAX\{ s(a_i) \mid a_i\in C_i, 1\leqslant i\leqslant d\}$

(3)构造 SM 算法　在确定基本周期 BC 之后，然后按照下列算法构造 SM。

```
Algorithm A
{  input L;
   determined basic cycle BC;
   m=1;
   while L≠ϕ
   {  Level(BCm)=0;Jm=ϕ;
      get element a from L;
      L=L-[ a ];
      if Level(BCm)+ a≤BC
      {  Jm=Jm+[ a ];Level(BCm)=Level(BCm)+S(a); }
      else
      {  m=m+1;Level(BCm)= S(a); }
   }
}
```

3.3 算法性能分析

构造 SM 并使得 m 最少是一个典型的最优化问题，算法 A 可以求得 SM 的近似最优解。算法 A 的时间开销主要集中在基本周期 BC 的确定上，对于最小周期、平均周期和最大周期的确定而言，算法 A 的时间复杂度为 $O(n)$；对于折衷周期的确定而言，由于需要对 L 的元素排序后进行数据划分，因此算法 A 的时间复杂度为 $O(n\log n)$。衡量近似最优算法 A 的好坏用采用最坏渐近性能比[5,6] R^{∞}。

最坏情况的渐近性能比(R^{∞})：对于输入的任务序列 L，理论上 SM 的时间记为 T，算法 A 所构造的 SM 时间记为 m * BC(m 是基本周期数)，则最坏情况渐近性能比定义如下：

$$R^{\infty}=\inf\{r\geqslant 1; \exists N>0, T\geqslant N \text{ 时，有} \frac{m\times BC}{T}\leqslant r \text{ 成立}\}$$

4 实验结果

数据精度是指在 L 集合中任务的最小持续时间和最大持续时间之比。在$[a,b](a,b\in Z)$区间上数据随机均匀分布的原则下，把数据按照数据精度分为百分之一、千分之一、万分之一和十万分之一四类，每类采集了 1000 组数据，每组实验仿真的数据量从 2^6 到 2^{20} 不等，对最小、平均、最大和折衷这四种确定 BC 的方法进行了仿真实验，实验结果如表 1 所示。

表 1　算法的 R^{∞} 实验数据表

r(精度) \ 周期(R^{∞})		R^{∞} 最小	平均	最大	折衷
$[2^6, 2^{10})$	0.01	1.34	1.22	1.53	1.29
	0.001	1.37	1.50	1.93	1.21
	0.0001	1.38	1.50	2.00	1.27
	0.00001	1.37	1.60	2.00	1.23
$[2^{10}, 2^{14})$	0.01	1.35	1.04	1.05	1.05
	0.001	1.35	1.27	1.29	1.20
	0.0001	1.35	1.45	1.92	1.21
	0.00001	1.35	1.50	2.00	1.09
$[2^{14}, 2^{18})$	0.01	1.35	1.02	1.02	1.02
	0.001	1.34	1.04	1.03	1.03
	0.0001	1.34	1.23	1.48	1.20
	0.00001	1.34	1.39	1.85	1.16
$[2^{18}, 2^{20})$	0.01	1.34	1.02	1.01	1.01
	0.001	1.34	1.01	1.01	1.01
	0.0001	1.34	1.02	1.04	1.04
	0.00001	1.34	1.07	1.14	1.13

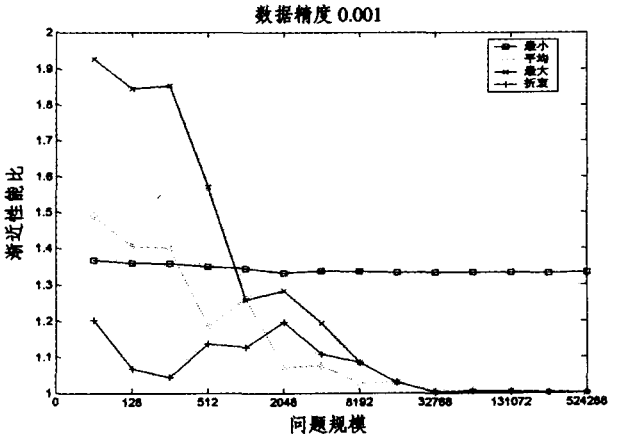

图 1　0.001 精度下算法的 R^{∞} 实验曲线

各种精度的数据实验结果表明，随着 n 的增大，四种周期的 R^{∞} 波动逐步趋稳；最小周期的 R^{∞} 基本稳定在 1.3 与 1.4

(下转第 225 页)

表 1 数据集 S_1 的实验结果

算法名称	平均聚类准确率(%)	平均迭代次数	实验次数	输出神经元数目
K 均值聚类	57.6	21	100	8
SOM 聚类	78.4	34	100	8
改进的 SOM 聚类	82.7	52	50	8

表 2 数据集 S_2 的实验结果

算法名称	平均聚类准确率(%)	平均迭代次数	实验次数	输出神经元数目
K 均值聚类	46.3	14	100	5
SOM 聚类	60.2	53	100	5
改进的 SOM 聚类	84.5	84	30	6

表 3 样本变化率实验结果

	S_1 集合样本变化率(%)	S_2 集合样本变化率(%)
SOM 聚类	23.6	36.7
改进的 SOM 聚类	5.23	10.01

从表 1 和表 2 的平均迭代次数可以看出 K 均值聚类收敛速度最快,可处理的数据集合最大。但是,所得到的结果通常只是局部最优,或者说聚类准确率较低。K 均值方法比较适合于对实时性要求较高,对精度要求不高的信息分类。在构件库管理系统中构件的入库分类属于预处理过程对于实时性没有较高的要求,因此不宜采用 K 均值聚类。改进的 SOM 聚类准确率大大优于传统的 SOM 聚类和 K 均值聚类,但是改进的 SOM 聚类方法较传统的 SOM 聚类法收敛速度略慢。这主要是由于将阈值函数由线性递减改为正弦递增势必影响其收敛速度造成的,而且训练过程还包含了神经元生长过程也会增加训练时的迭代次数。由表 3 的样本变化率可以看出在样本输入次序发生变化之后改进的 SOM 聚类的聚类结果的变化远远小于传统的 SOM 聚类,所以在实验过程中改进的 SOM 聚类的多次试验的结果基本相同,因此可以适当减少实验次数,这样在时间上可以弥补其收敛速度较慢的缺点。

结论 为了克服软构件刻面分类法需要人工建立和维护术语空间的缺点,本文提出了一种基于 SOM 聚类的构件自动分类方法。该方法将聚类分析技术应用于软构件的分类中,定义了构件类别树,实现了摆脱术语空间的软构件自动分类,而且在分类时充分考虑了实际系统部署时构件之间的相互关系,使分类的结果更有利于构件的检索和基于构件系统的搭建。针对 SOM 聚类的聚类结果与样本输入次序有关和需要预先确定拓扑结构的缺点,文中提出了一种改进的 SOM 聚类算法并在实验中将其性能与 K 均值聚类和传统的 SOM 聚类算法加以比较,实验数据表明改进的 SOM 聚类算法聚类的准确率大大提高,而且可以通过在训练过程中生长神经元是找到最优的拓扑结构。

参 考 文 献

1 Poulin J S, Yglesias K P. Experiences with a faceted classification scheme in a large reusable software library(RSL). In: Proc. of Seventeenth Annual International Computer Software and Applications Conference, Phoenix, AZ, 3-5, 1993. 90～99

2 Morel J M, Faget J. The REBOOT environment. In: Proc. of the 2nd Intl. Workshop on Software Reusability Advances in Software. Lucca: IEEE Computer Society Press, 1993. 88～97

3 NEC Software Engineering Laboratory. NATO standard for management of a reusable software component library Volume 2. Tokyo: NATO Communication and Information System Agency, 1991

4 Chang JC, Li KQ, Guo LF, Mei H, Yang FQ. Representing and retrieving reusable software components in JB(Jadebird) system. Electronica Journal, 2000, 28(8): 1～6

5 Han Jiawei, Kamber M. Data mining: concepts and techniques. Jim Gray: Series Editor Morgan Kaufmann Publishers, 2000

6 Kohnen T. The self-organizing map. Neurocomputing. 1998, 21: 1～6

7 Yang Kiduk, Jacob E. A hybrid approach to generating and utilizing faceted vocabulary for knowledge discovery on the web. In: Proc. of the 2004Joint ACM/IEEE Conference on Digital Libraries. New York: ACM Press, 2004. 395～395

8 Yao Haining, Etzkorn L. Towards a semantic-based approach for software reusable component classification and retrieval. In: Proc. of the 42nd Annual Southeast Regional Conference. New York: ACM Press, 2004. 110～115

(上接第 215 页)

之间,比较平稳;折衷周期和最大周期的 R^{∞} 在 n 足够大时趋于一致;R^{∞} 具体变化如图 1 所示,图 1 是在精度为千分之一时,算法的实验结果曲线,一般而言,当 $n>2^k$ 时(k 是整数,k 随精度变化),最小周期算法好于最大和平均周期算法,当 $n<2^k$ 时,平均周期、最大周期和折衷周期算法好于最小周期算法;折衷周期算法始终优于最大周期算法;平均周期算法在 $n<2^j$ 时(j 是整数,j 随精度变化),当 n 足够大时,平均周期法优于折衷周期法。

小结 本文给出了基于时间触发的 CAN 协议构造 SM 的一种优化算法,详细介绍了构造 SM 的过程,简要分析了算法的性能,给出了实验分析结果,指出在绝大多数情况下,折衷周期算法的性能要优于其它三种算法。

本文进一步的工作是:在构造好基于时间触发的 CAN 协议任务调度的 SM 之后,对 SM 中相同性质的任务如何调整,以降低通信过程中的同步信息帧的复杂度。

参 考 文 献

1 Hartwich F ,Müller B,F Führer Th. Timing in the TTCAN network [A]. In: Proc. 8th Intl. CAN Conf. [C], Las Vegas, Nv, 2002

2 Führer Th,Müller B,Dieterle W. Time Triggered Communication on CAN[A]. In: Proc. 7th Intl. CAN Conf. [C]. Amsterdam, Netherland, 2000

3 Fonseca J,Countinho F,Barreriros J. Scheduling for a TTCAN network with a stochastic optimization algorithm [A]. In: Proc. of 4th FET2001,IFAC conf. on Fieldbus System and their Application[C]. Nancy,France,2001

4 Countinho F. Using Genetic algorithms to reduce jitter in control variables transmitted over CAN [c]. In: Proc. of ICC'2000 [C], Amsterdam,Netherland,2000

5 Coffman E G,Courcoubetis C. Perfect packing theorems and the averge-case behavior of optimal and online BIN packing [J]. SIAM Review 44,2002. 95～108

6 Xiaodong GU. Performance Analysis and Improvement for Some Linear On-Line Bin-Packing Algorithms [J]. Journal of Combinatorial Optimization,2002,6:455～471