

# 多功能满足可靠性目标的资源最小化研究文献综述

**摘要：**汽车电子功能安全问题的研究有着不容忽视的重要意义。由于汽车嵌入式系统是分布式的可靠嵌入式系统，处理器性能大幅度提升的同时，却造成系统的故障发生率不断增加，其面临的可靠性问题日益严重，因而满足汽车电子功能安全可靠性目标对确保汽车安全运行十分关键。同时，调度是提高并行分布式系统性能的重要技术手段。因此，在满足可靠性的目标下，研究多个汽车功能共享一组分布式资源的调度问题有着重要的应用现实意义。

## 1 研究背景与意义

汽车自诞生以来，人们就没有停止过对安全驾驶的追求。最早的安全带以及后来的安全气囊等被动安全措施挽救了数千万人的生命。后来发展起来的防抱死制动系统(Anti-lock Braking System, ABS)、电子稳定程序(Electronic Stability Program, ESP)、电子制动力分配(Electronic Brake-force Distribution, EBD)等主动安全功能让汽车安全性再次大幅提升。随着新一代汽车网联化、智能化、电动化的发展趋势，融合主动和被动安全的高级驾驶辅助系统(Advanced Driver Assistant System, ADAS)<sup>[1]</sup>则将新一代汽车的安全性提高到了一个新的高度。汽车嵌入式系统是高度安全关键(Safety-Critical)的高端嵌入式系统<sup>[2,3]</sup>，然而，系统中可能存在多种系统性失效、随机硬件失效、时序失常、功能执行的逻辑顺序打乱和数据损坏等可能导致汽车安全事故的风险。“风险”即这些失效发生伤害或损害的可能性，及伤害或损害所造成的后果的严重性<sup>[4]</sup>，例如，车辆意外的加速、减速与转向，安全气囊非正常弹开，高速行驶时车门突然打开等等。功能安全(Functional Safety)就是为应对上述风险而提出，它是指避免因系统功能性失效导致的不可接受的安全伤害和风险，并已成为电气电子系统研发中的一项重要需求<sup>[5,6]</sup>。2011年，专门针对汽车嵌入式系统的功能安全标准—道路车辆-功能安全(Road Vehicles-Functional Safety)标准规范 ISO 26262 应运而生<sup>[6,7,8]</sup>。该标准几乎涉及到了包括传统汽车和新能源汽车中的所有与功能安全相关的汽车嵌入式产品<sup>[8]</sup>。汽车嵌入式系统自身的安全性应得到保证，即其中可能存在的潜在安全风险应当被控制在可接受的范围之内。ISO 26262 标准要求设计人员提前评估所有潜在的风险，并采取适当的措施尽最大可能地消除这些风险，以确保汽车嵌入式系统中与安全相关的功能，特别是主动安全功能

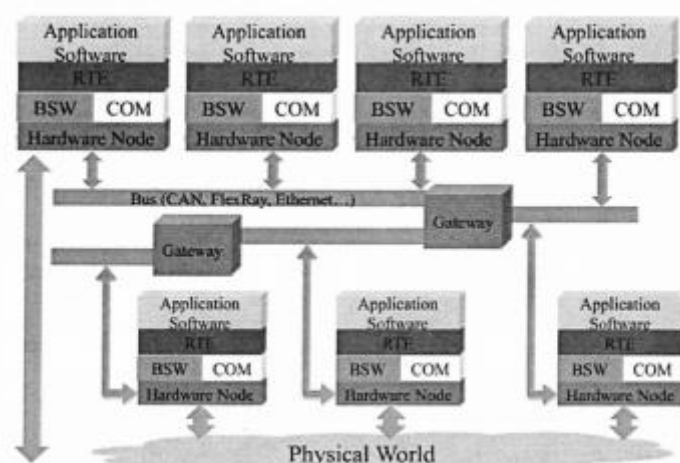
（如防抱死制动、车身电子稳定、电子制动力分配、前方防撞警示、车道维持、车道偏移警示、驾驶者状态监控、指纹辨识免钥、盲点侦测与开门警示、自动停车导引等）的功能安全<sup>[8,9]</sup>。因此，确保汽车嵌入式系统的安全功能在各种严酷条件下保持正常运行，以及驾乘人员及行人的安全，是现代汽车嵌入式系统设计的核心任务。

汽车电子系统是典型的异构分布式嵌入式计算系统<sup>[10]</sup>。为了满足人们在安全性和驾驶性能上提出的更高要求，汽车电子系统的体系结构日益复杂，遍布车内的 ECU 个数达到 100 多个<sup>[11]</sup>，并通过多种车辆网络总线和网关实现互联，系统复杂性骤<sup>[11,12]</sup>。现代轿车包含的处理设备由各类异构 ECU、传感器、执行器和网关等物理设备组成，各处理设备之间通过异构网络总线互联，以实现交互和协作功能，系统异构性特征明显<sup>[13]</sup>。与此同时，汽车电子系统软件规模骤增，车内的电子化功能个数达到 800 多个<sup>[14]</sup>。据统计，汽车电子系统相关的成本已经占到每辆汽车总成本中 30%-40%<sup>[16]</sup>，汽车工业相关创新中的 90%都依赖于电子化和软件的驱动<sup>[15]</sup>，车内的代码量达到上千万行<sup>[16]</sup>，软件总容量已达至 230MB<sup>[17]</sup>。汽车产业带来新经济增长方向的同时也在不断革新人们对汽车的认识，汽车已由“会走的一堆钢铁”变成“会走的计算机”。汽车电子系统设计正从过去以 ECU 为中心的思想转换为从整个分布式系统设计的视角出发。与传统的汽车电子产品开发强调单个 ECU 的功能和性能不同，现今的汽车电子技术强调从整体角度出发，着重关注融合了计算技术与网络技术的分布式汽车电子系统面临的性能、实时和可靠等与汽车功能安全紧密相关的问题。其中，由于汽车嵌入式系统是分布式的可靠嵌入式系统(Distributed Dependable/Reliable Embedded Systems)，其面临的可靠性问题日益严重。处理器性能大幅度提升的同时，却造成系统的故障发生率不断增加。首先，汽车嵌入式系统由不同类型的子系统集成，系统的复杂化使得设计难度增加，且当修改软件的某部分功能时，会影响与之有紧密依赖与约束的部分。其次，若某个子系统内发生错误时，易引发其错误在不同的子系统之间传播的风险，系统功能安全面临严峻的威胁。再者，汽车行驶过程中电磁干扰、消息丢失等容易造成 ECU 和通信网络出现瞬时故障。此外，系统中存在大量的功能部件且运行时温度较高，也易造成瞬时故障，因可靠性和安全性方面的问题而引起的汽车召回事件不断发生。所以，

评估功能的可靠性，并采取适当的措施最大可能的满足功能所认证的可靠性目标有着不容忽视的重要意义。

## 2 汽车电子系统结构

现代汽车电子系统的系统结构如图所示，系统中的传感器等处理设备通过与物理世界实现动态交互，通过网络总线交由相应的计算单元处理，并作出正确的行驶、刹车和转向等功能。因此，异构网络化汽车电子系统是一个同时集成了计算系统、网络系统和物理系统的复杂分布式系统，各处理单元通过与汽车外围的物理环境、周围行驶的汽车及基础设施之间进行实时交互，来实现对汽车的精确化、智能化和集成化控制，以提升汽车的行驶安全、驾驶性能、舒适性、可操控性和环保性能<sup>[18]</sup>。因此汽车电子系统是一个典型的信息-物理融合系统(Cyber-Physical System, CPS)<sup>[19-22]</sup>。业界第一本关于 CPS 的专著《嵌入式系统导论- CPS 方法》在序言中多次提到：“CPS 中许多事件是同时发生的；物理过程是很多并行过程的组合；并发是 CPS 的固有特征；CPS 总是面临并发问题的挑战<sup>[23]</sup>。因此，正如以汽车信息-物理融合系统(Automobile Cyber-Physical Systems, ACPS)<sup>[24,25]</sup>的观点来研究汽车特性一样，并发也是 ACPS 的固有属性。因此，以并行与分布式计算的思想研究分布式汽车电子系统的若干问题是一种必然的趋势和可行的方法。



在体系结构上，分布式汽车电子系统相比集群计算<sup>[26]</sup>、网格计算<sup>[27]</sup>和云计算<sup>[28]</sup>等通用并行分布式系统有显著差异。首先在异构性上，通用分布式系统主要强调所参与的分布式计算处理机的硬件结构和执行性能的异构性，利用局域网或互联网络把地域上分散的计算资源连接起来，简言之，即处理机的异构性与位置的分散性<sup>[29]</sup>；而异构网络化汽车电子系统不仅强调计算单元(如 ECU,

传感器和执行器等)的异构性,更关注网络的异构性,如基于事件触发的 CAN<sup>[30]</sup>,基于时间触发的 FlexRay<sup>[31]</sup>以及 LIN<sup>[32]</sup>,MOST<sup>[33]</sup>甚至 Ethernet<sup>[34]</sup>等各种异构网络总线都集成到了汽车网络中,这些网络技术通过网关以实现互联<sup>[35]</sup>。其次在网络化上,通用异构分布式系统主要强调计算处理单元在物理位置上的分散,而且通常由一个全局主机负责协调多个从属主机的计算和通信,比如当今在产业界和学术界都颇受关注的 Google Mapreduce<sup>[36]</sup>和 Hadoop 等分布式系统<sup>[37]</sup>,它们都存在多个从属主机受控于一个全局主机,这实际上不是完全的分布式系统,而是集中式与分布式相结合的系统;而分布式汽车电子系统中的处理单元不存在任何主从关系,它们是完全自治的实体,并且各处理单元的通信可以通过网络和网关进行直接或间接交互,而不受全局主机和系统的控制,是一种完全自治的分布式系统<sup>[38]</sup>。因此,可将本文所研究的汽车电子系统这一实体描述为计算异构、网络异构、完全分布、并行处理、深度融合与嵌入的异构网络化嵌入式系统。通用异构分布式系统的核心研究内容也就是调度,其目标是在共享计算和网络资源条件下,通过减少并行分布式功能应用的总执行时间,提高系统的吞吐量和计算效率。通用异构并行分布式系统的调度问题往往被认为是任务调度或处理机调度,在满足系统性能、可靠性和任务优先约束等条件下,将可并行任务依据任务调度策略确定任务分配方案和任务执行顺序,以达到减少并行分布式功能应用的总执行时间、提高系统吞吐量、可靠性和安全性等。

### 3 基于可靠性目标的多 DAG 调度资源最小化

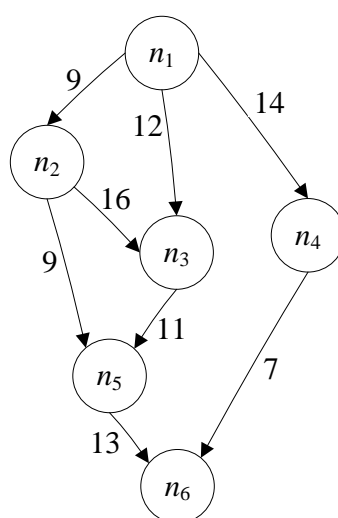
汽车电子系统的异构性、网络化和复杂化使其调度问题变得日益复杂,抽象是解决调度问题的首要步骤。传统的嵌入式系统调度模型已无法精确的表达汽车电子系的并行与分布式本质。因此,需要一种新的形式化的并行与分布式调度模型加以描述,以真实反应汽车电子系统的功能应用特征。

#### 3.1 汽车电子系统的调度模型抽象

##### 3.1.1 功能应用的 DAG 抽象

在并行与分布式领域,任务间的优先级约束关系常用有向无环图(Directed Acyclic Graph, DAG)模型表示。如图所示,图中的每个结点代表一个任务;任务的优先约束关系用有向边表示,也即任务之间的通信消息<sup>[39]</sup>。可用一个 DAG

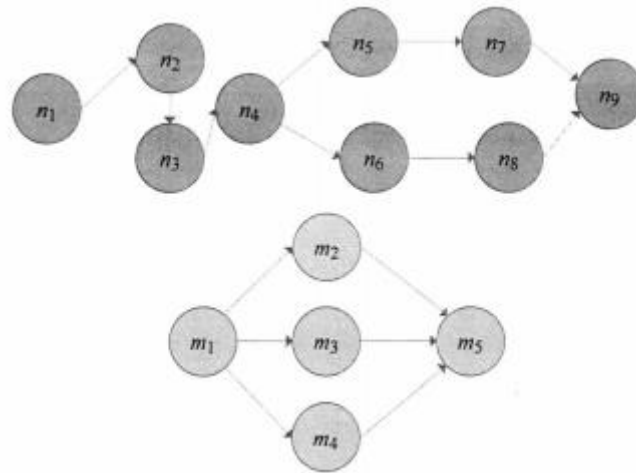
描述一个功能应用，用  $D=\{N,E\}$  表示一个 DAG，其中， $N=\{n_1,n_2,\dots,n_i\}$  表示任务集合； $E=\{e_1, e_2,e_3, \dots,e_{i,j}\}$  表示任务间的通信消息； $pred(n_i)$  表示任务  $n_i$  的直接前驱任务集合， $ind(n_i)$  表示以  $n_i$  的入度，也就是其直接前驱任务集合的个数，当前任务只有在它全部前驱任务的数据到达后才能执行。 $succ(n_i)$  表示任务  $n_i$  的直接后继任务集合， $outd(n_i)$  表示  $n_i$  的出度，也就是直接后继任务集合的个数。没有前驱任务的任务为入口任务，记为  $n_{entry}$ ；没有后继任务的任务为出口任务，记为  $n_{exit}$ ；前面多次提到的功能应用端到端，也即 DAG 中的入口任务到出口任务。DAG 任务模型不仅考虑了任务的优先级，还考虑了任务之间的通信消息，因此适合于异构分布式并行系统的调度问题研究。



### 3.1.2 汽车电子系统的多 DAG 抽象

然而，汽车电子系统是由动力控制子系统、底盘控制子系统、安全控制子系统和车身控制子系统等多个子系统组成的混合系统。每个子系统又可细分多个功能应用(如安全控制子系统包括防抱死制动、线控刹车等)。不同的子系统融合多种网络总线。这些子功能为满足不同的需求，分别会采用不同的设计方法，其组成部件也由不同级别的汽车供应商提供<sup>[40]</sup>，造成不同功能在时间需求和可靠性需求上都不一样，并产生不同的时间关键级别和可靠性关键级别。因此，需以多 DAG(Multiple DAGs)<sup>[41,42]</sup>来表示汽车电子系统调度模型，相互之间具有优先级约束的一组任务表示一个 DAG 功能应用，各种 DAG 共享处理器和通信总线以实现协同调度，由于汽车电子系统是在行驶状态下与物理世界动态交互，并且可能在任意时刻作出加速、刹车和转向等功能，即各 DAG 不一定同时发生，并且可以任意提交，因此属于动态多 DAG 模型，采用多 DAG 动态

环境下的异构网络化汽车电子系统的调度问题将更加复杂。本文用  $DS=\{D_1, D_2, \dots, D_m\}$ 表示多 DAG 调度模型。因此，可将异构网络化汽车电子系统的调度问题形式化为多 DAG 调度问题并进行研究。例如图所示为拥有 2 个 DAG 的多 DAG 调度模型实例，用  $DS_I=\{D_1, D_2\}$ 表示，其中  $D_1$ 为线控刹车功能， $D_2$ 表示转向功能。



### 3.2 多 DAG 调度资源最小化问题

各种类型的 DAG 任务在多个处理单元上的调度已被证明是 NP (Non-deterministic Polynomial) 完全难题，即完全多项式非确定性问题<sup>[43]</sup>。从被调度对象的 DAG 类型、资源环境和调度目标等方面对多 DAG 调度可有不同方式的分类，例如：从用户对 DAG 应用的起始和结束时间是否有限制来看，一般可分为无期限约束的 DAG 调度和具有期限约束的 DAG 调度<sup>[44]</sup>；从 DAG 图中的每个结点任务是否能够继续被分割并且能并行在任意数目机器上执行来看，DAG 任务模型又可分为 Moldable 和 Unmoldable 两种类型<sup>[45]</sup>；从任务的调度映射方案是否在 DAG 执行开始前就已确定来看，有动态调度和静态调度两类算法；根据 DAG 任务是否包含必要和非必要计算成分，可分为精确计算 DAG 和非精确计算 DAG 任务的调度算法<sup>[46-47]</sup>；依据 DAG 任务本身大小是否具有随机性，可分为随机 DAG 调度和非随机 DAG 调度等等。关于多 DAG 共享资源混合调度问题的研究主要则是针对 DAG 混合调度的时间最小化、公平性最大化、吞吐量最大化、资源分配优化等方面。

#### 3.2.1 主要调度目标

针对多 DAG 共享异构分布式资源的调度问题，现有提出的目标主要集中在以下几个方面：

#### (1) 资源利用率最大化

从资源的提供者或系统管理的角度来看，在一定的周期内，资源利用率（Utilization of Resource, UR）越高越好。尤其是在效用网格和公有云计算环境下，一般会根据资源类型、性能和资源被使用的时间进行计费，那么资源利用率越高，意味着资源在单位时间内的使用效率和经济效益也越高，因此很多调度算法的重要目标就是资源利用率的最大化。多 DAG 应用混合调度的根本目的就是多用户应用共享资源，降低费用，并尽可能提高资源利用率<sup>[23,36]</sup>。

#### (2) 吞吐量最大化

与资源利用率最大化一样，如果在一定的周期内，系统能够调度完成的 DAG 数量或任务数量越多，则说明调度的效率越高、越合理，在一定的周期内也就能够为更多的 DAG 应用用户提供计算和应用服务。吞吐量最大化既是标准传统的独立任务调度性能衡量指标，也是相关的任务（DAG）调度的重要调度性能指标。

#### (3) 调度长度最小化

如果多个 DAG 对 QoS 的要求是尽快地完成执行，显然，调度算法应该能够使得每个 DAG 的调度长度越短越好<sup>[48]</sup>。不仅是现有绝大多数的单 DAG 调度算法的一个基本调度目标，也是多 DAG 混合调度算法<sup>[23-29,65-67]</sup>的主要调度目标之一。

#### (4) 时间公平性最大化

多个 DAG 共享一组有限数量的分布式资源调度运行，会产生不同 DAG 对资源的竞争，必然存在调度时间上的公平性问题。多 DAG 调度的时间公平性定义及其最大化是重要的调度目标。

### 3.2.2 基本技术和方法

在多 DAG 混合调度过程中，某个 DAG 内部的结点之间的调度关系可根据现有公认和成熟的调度方法，比如用 HEFT 的相关方法来确定。然而，多 DAG 混合调度与单 DAG 调度相比，要解决的重要问题之一就是如何处理好多个 DAG 任务之间的调度关系。现有有关多 DAG 共享异构分布式资源调度的研究

根据不同的 DAG 应用需要和调度目标等也都提出了多种调度方法和技术，但主要的（参考）方法有以下几种：

#### （1）顺序法

顺序法是常见的处理多任务的简单方法，也可以用来处理多 DAG 共享资源调度的问题。具体就是将所有的 DAG 按一定的顺序，调度完毕一个 DAG 后，再调度下一个 DAG，或者按照先来先服务（FCFS）的方法处理多个 DAG 之间的调度关系。

#### （2）轮转法

轮转法（Round-Robin）是处理分时操作系统的多个任务调度问题中简单的传统方法。该方法也进一步被借鉴用于解决处理多个 DAG 的调度问题。具体的方法为，轮流从每个 DAG 中选取一个  $rank_u$  值最高的任务利用 HEFT 的映射方法调度该任务。比如，Bittencour 等人针对多个网格 workflow 调度问题时，提出了一种从入口结点开始深度搜索最长路径的聚簇方法将每个 DAG 的任务聚簇，而每个簇的任务放在同一个资源上执行，那么每个 DAG 会被分为多个簇，然后根据 Round-Robin 方法，按照某种优先级的规则轮流从每个 DAG 中选取若干簇插入资源空闲间隙进行调度。

#### （3）DAG 合并法

合并法由 U.Honig 针对同构环境下的多 DAG 调度问题所提出。几乎同时，该方法也被 Z.Henan 提出用于解决异构分布式系统的多 DAG 调度问题。方法的主要思想是将多个 DAG 通过加入虚假的入口和出口结点来合成一个复合 DAG 后，然后再利用某种单 DAG 的调度方法来混合调度合成后的 DAG。

#### （4）公平调度方法

Z.Henan 和 R.Sakellariou 提出了一种 Fairness-F1 公平调度方法，并已成为目前关于多 DAG 混合调度的研究中衡量时间公平性指标重要的参考方法。其主要思想是，总是先调度多个 DAG 中相对滞后那个 DAG 中的权值最大的一个任务。

## 4 总结

汽车电子系统可靠性目标是功能安全 ISO26262 标准的重要目标，而基于分布式功能的 DAG 模型的研究是当前异构分布式系统的热点。然而，目前大多



数 DAG 任务调度的研究大多数是针对单个 DAG 在多个资源上的调度问题，并且这些研究已经趋于成熟，而针对多 DAG 共享一组分布式计算资源的调度问题研究的较少。针对汽车电子系统的多 DAG 抽象模型，在功能安全可靠目标驱动下，研究其共享一组分布式资源而使其资源利用最小化是目前非常值得研究的问题。

## 参考文献

- [1] Kang J S, Kim J, Lee M. Advanced driver assistant system based on monocular camera[C]// Consumer Electronics (ICCE), 2014 IEEE International Conference on. IEEE, 2014:55-56.
- [2] Sangiovanni-Vincentelli A, Natale M D. Embedded System Design for Automotive Applications[J]. Computer, 2007, 40(10):42-51.
- [3] Chen D J, Johansson R, Lönn H, et al. Modeling support for design of safety-critical automotive embedded systems[M]//Computer Safety, Reliability, and Security. Springer Berlin Heidelberg, 2008: 72-85.
- [4] Sinha P. Architectural design and reliability analysis of a fail-operational brake-by-wire system from ISO 26262 perspectives[J]. Reliability Engineering & System Safety, 2011, 96(10):1349-1359.
- [5] [https://en.wikipedia.org/wiki/IEC\\_61508](https://en.wikipedia.org/wiki/IEC_61508)
- [6] [https://en.wikipedia.org/wiki/ISO\\_26262](https://en.wikipedia.org/wiki/ISO_26262)
- [7] ISO ISO. 2011. 26262—Road vehicles—Functional safety. ISO Standard(2011)
- [8] 耿莉莉. 基于 ISO26262 标准的安全关键嵌入式软件开发技术与工具[D]. 浙江大学, 2013.
- [9] 刘佳熙, 郭辉, 李君. 汽车电子电气系统的功能安全标准 ISO26262[J]. 上海汽车, 2011(10):57-61.
- [10] Pop T, Eles P, Peng Z. Schedulability analysis for distributed heterogeneous time/event triggered real-time systems. In: 15th Euromicro Conference on Real Time Systems. Porto, Portugal: IEEE, 2003, 257-266.
- [11] Albert A. Comparison of event—triggered and time—triggered concepts with regard to distributed control systems. Embedded World, 2004, 2004: 235-252.
- [12] Teepe G, Remboski D, Baker R. Towards information centric automotive system architecture. In: International Congress on Transportation Electronics. New York: USA: SAE, 2002, 387-392.
- [13] Navet N, Song Y, Simonot. Lion F, et al. Trends in automotive communication systems. Proceedings of the IEEE, 2005, 93(6): 1204-1223.
- [14] First S. Challenges in the Design of Automotive Software. In: Proceedings of the Design, Automation, and Test in Europe Conference and Exhibition. Dresden, Germany: IEEE, 2010, 256-258.
- [15] Buckl C, Camek A, Kainz G, et al. The Software Car: Building ICT architectures for future electric vehicles. In: Proceedings of the IEEE International Electric Vehicle Conference. Kuching, Malaysia: IEEE, 2012, 1-8.
- [16] Broy M, Kruger I, Pretschner A, et al. Engineering Automotive Software. Proceedings of the IEEE, 2007, 95(2): 356—373
- [17] Audi A8, 10 Electrical and network systems. <http://www.audionlinetraining.com,2010>.
- [18] 谢勇. 新一代汽车电子系统的网络体系结构若干关键技术研究: [湖南大学博士学位论文 1. 长沙: 湖南大学, 2013.
- [19] 何积丰. Cyber. Physical Systems. 中国计算机学会通讯, 2010, 6(1): 25-29.

- [20] 李仁发, 谢勇, 李蕊等. 信息. 物理融合系统若干问题综述. 计算机研究与发展, 2012, 49(6): 1149. 1161.
- [21] Rajkumar R, Lee I, Sha L, Stankovic J. Cyber Physical Systems: The Next Computing Revolution. In: Proceedings of the 47 ACM/IEEE Design Automation Conferences. Anaheim, USA: IEEE, 2010, 73 1—736.
- [22] 李建中, 高宏, 于博. 信息物理融合系统(CPS)的概念、特点、挑战和研究进展. 中国计算机科学技术发展报告, 2009, 1-17.
- [23] 李实英, 贺蓉, 李仁发. 嵌入式系统导论. CPS 方法. 北京. 机械工业出版社. 2011.
- [24] Work D, Bayen A, Jacobson Q. Automotive cyber-physical systems in the context of human mobility. In: National workshop on high-confidence automotive cyber-physical systems. Troy, USA: MIT Press. 2008.
- [25] NIST SF/USCAR Workshop on Developing Dependable and Secure Automotive Cyber-Physical Systems from Components. <http://varma.ece.cmu.edu/Auto.CPS-2011/>.
- [26] Heath T, Diniz B, Carrera E V, et al. Energy conservation in heterogeneous server clusters. In: Proceedings of the tenth ACM SIGPLAN symposium on Principles and practice of parallel programming. New York, USA: ACM, 2005, 186. 195.
- [27] Buyya R, Murshed M. Gridsim: A toolkit for the modeling and simulation of distributed resource management and scheduling for grid computing. Concurrency and computation: practice and experience, 2002, 14(13): 1175. 1220.
- [28] J. Li, M. Qiu, Z. Ming, G Quan, et al. Online optimization for scheduling preemptable tasks on IaaS cloud systems. Journal of Parallel and Distributed Computing, 2012, 72(5): 666-677.
- [29] 唐小勇. 异构并行分布式系统可信调度理论与方法研究: [湖南大学博士学位论文]. 长沙: 湖南大学, 2013.
- [30] Tindell K W, Hansson H, Wellings A J. Analysing real time communications: controller area network. In: Proceedings of Real-Time Systems Symposium. San Juan, Puerto Rico: IEEE, 1994, 259 263.
- [31] Pop T, Pop P, Eles P, et al. Timing analysis of the FlexRay communication protocol. Real-time systems, 2008, 39(1—3): 205—235.
- [32] abriel C, Horia H. Integrating sensor devices in a LIN bus network. In: 26<sup>th</sup> International Spring Seminar on Electronics Technology: Integrated Management of Electronic Materials Production. Piscataway, USA: IEEE, 2003,150.]53.
- [33] Cooperation M. MOST media oriented system transport—multimedia and control networking technology. MOST Specification Rev, 2004, 2: 2. 2-00.
- [34] ello L L. The Case for Ethernet in Automotive Communications. ACM SIGBED Review,2011, 8(4): 7-15.
- [35] GlaB M, Lukasiewicz M, Teich J, et al. Designing heterogeneous ECU networks via compact architecture encoding and hybrid timing analysis. In: 46<sup>th</sup> ACM/IEEE conference on Design automation. San Francisco, USA: IEEE, 2009, 43. 46.
- [36] ]Dean J, Ghemawat S. MapReduce: simplified data processing on large clusters. Communications of the ACM, 2008, 51(1): 107-113.
- [37] Xie J, Yin S, Ruan X, et al. Improving mapreduce performance through data placement in heterogeneous hadoop clusters. In: 20 1 0 IEEE International Symposium on Parallel&Distributed Processing, Workshops and Phd Forum. Atlanta, GA, USA: IEEE, 20 10, 1-9.
- [38] eller M, Prehofer C, Weiss G, et al. Towards self-adaptation in real-time, networked systems: efficient solving of system constraints for automotive embedded systems. In: 5th IEEE International Conference on Self-Adaptive and Self\_Organizing Systems. Ann Arbor, Michigan, USA: IEEE, 201 1, 79-88.

- [39] Yong XIE, Gang ZENG, Yang CHEN, et al. Worst Case Response Time Analysis for Messages in Controller Area Network with Gateway. *IEICE Transactions on Information and Systems*, 2013.
- [40] Marco Di Natale, Scuola Superiore S. Anna. Design and development of component-based embedded systems for automotive applications. In: *Proceedings of the 13th Ada—Europe International Conference on Reliable Software Technologies*. Venice, Italy: Springer-Verlag, 2008, 15-29.
- [41] Zhao H, Sakellariou R. Scheduling multiple DAGs onto heterogeneous systems. In: *20th International Parallel and Distributed Processing Symposium*. Rhodes Island, Greece: IEEE, 2006, 14.
- [42] 田国忠, 肖创柏, 徐竹胜等. 异构分布式环境下多 DAG 工作流的混合调度策略. *软件学报*, 2012, 23(10): 2720-2734.
- [43] J D ULLMAN. NP-complete scheduling problems[J]. *Journal of Computer and System Sciences*, 1975, 10(3): 384-393.
- [44] E BYUN, S CHOI, M BAIK, J GIL, C PARK AND C HWANG. MJSA: Markov job scheduler based on availability in desktop grid computing environment[J]. *Future Generation Computer Systems*, 2007, 23(4): 616-622.
- [45] W CIRNE AND F BERMAN. A model for moldable supercomputer jobs[C]. *Proc. of the 15th International Parallel and Distributed Processing Symposium (IPDPS'01)*. IEEE Computer Society, 2001.
- [46] GEORGIOS L, STAVRINIDES HELEN D. Scheduling multiple task graphs with end-to-end deadlines in distributed real-time systems utilizing imprecise computations[J]. *Journal of Systems and Software*, 2010, 83(6): 1004-1014.
- [47] GEORGIOS L, STAVRINIDES HELEN D. Scheduling real-time DAGs in heterogeneous clusters by combining imprecise computations and bin packing techniques for the exploitation of schedule holes[J]. *Future Generation Computer Systems*, 2012, 28(7): 977-988.
- [48]