

# 基于多参数协作驱动的智能网络化系统设计应用

王亚军

(华北计算技术研究所, 北京 100083)

**摘要:** 为了提高数据传输效率和精度, 改善网络控制性能, 基于多参数协作驱动, 设计了一种智能网络化系统; 首先, 基于数据采集、线性化处理、参数库、驱动模块、存储与管理等模块设计网络控制系统, 同时给出了多参数协作驱动模块中实时更新的参数与网络信号之间的映射关系; 其次, 由核心控制和中继协作组建智能网络化系统; 该系统的核心控制子系统部署于服务器, 作为智能网络平台的核心; 中继协作子系统部署在中继终端; 该系统的两个子系统基于多参数协作驱动算法实现实时交互; 最后, 实验结果表明, 所设计的系统在执行效率和数据精度等方面具有明显优势。

**关键词:** 多参数; 协作驱动; 智能系统; 网络化平台

## Design and Application of Intelligent Network System Based on Multi Parameter Cooperative Drive

Wang Yajun

(North China Institute of computing technology, Beijing 100083, China)

**Abstract:** In order to improve the efficiency and accuracy of data transmission and the network control performance, an intelligent networked system is designed based on multi parameter cooperative drive. First, the network control system is designed based on data acquisition, linearization processing, parameter library, drive module, storage and management module. At the same time, the mapping relation between the real-time updating parameters and the network signals in the multi parameter cooperative driving module is given. Secondly, an intelligent network system is set up by core control and relay cooperation. The core control subsystem of the system is deployed on the server as the core of the intelligent network platform, and the relay cooperation subsystem is deployed at the relay terminal. The two subsystems of the system are real-time interactive based on the multi parameter cooperative driving algorithm. Finally, the experimental results show that the designed system has obvious advantages in execution efficiency and data accuracy.

**Keywords:** multi parameter; cooperative drive; intelligent system; network platform

### 0 引言

无线传感器网络工作效率和无线通信稳定性<sup>[1]</sup>一直是制约无线应用推广的重要因素, 因此, 一系列研究试图通过网络化系统<sup>[2]</sup>解决上述问题。但是, 改善无线终端与服务器的智能化控制水平<sup>[3]</sup>成为关键。基于网络化平台<sup>[4]</sup>, 如何将终端智能化与数据传输网络化有机融合, 成为重要的研究对象。

文献 [5] 设计了一种基于蓄能器的二级怠速控制系统, 用于满足工程机械节能需求, 优化挖掘机工况, 从而确保取消怠速恢复工作时执行器入口能够快速建立压力目标; 文献 [6] 基于智能网络芯片与对象识别处理器的设计了一种在线强化学习便携式高清处理器; 文献 [7] 通过分布式网络化测量方法提高谐波故障源定位的在线辨识精度。

在上述研究成果基础上, 结合多参数协作控制与驱动, 设计了智能网络化系统。

### 1 多参数协作驱动算法

一个网络控制系统由数据采集、线性化处理、参数库、驱

动模块、存储与管理等模块组成, 结构如图 1 所示。该系统工作流程描述如下:

1) 网络采集系列数据, 组成数据流  $D$ 。该数据流的合成适用于单输入单输出、单输入多输出、多输入单输出和多输入多输出系统, 不同的输入与输出端口的转换与数据归一化处理, 详见公式 (1):

$$D = \begin{cases} \frac{1}{n}f(x_0) \oplus \frac{1}{m}f(x_0), & m=1, n=1 \\ \frac{1}{n}f(x_0) \oplus \sum_{i=1}^n f(x_i), & m=1, n>1 \\ \sum_{i=1}^m f(x_i) \oplus \frac{1}{m}f(x_0), & m>1, n=1 \\ \sum_{i=1}^m f(x_i) \oplus \sum_{i=1}^n f(x_i), & m>1, n>1 \end{cases} \quad (1)$$

式中,  $m$  表示输入端口数,  $n$  表示输出端口数。函数  $f(x)$  用户计算网络中传输的信号。 $f(x_0)$  表示网络初始状态下的信号。

2) 参数库中存储多个网络管理参数, 构成参数向量, 形如  $p_1 \in P_1, \dots, p_{\min(m,n)} \in P_{\min(m,n)}$ 。其中, 参数向量的个数由  $m$  与  $n$  的最小值决定, 这样可以有效地避免无效参数的产生。

3) 驱动模块, 访问参数库, 并经第 (1) 步得到  $m$  与  $n$

收稿日期: 2017-07-16; 修回日期: 2017-08-21。

作者简介: 王亚军(1992-), 男, 湖南衡南人, 在读硕士研究生, 主要从事数据分析、数据挖掘和大数据分析等方向的研究。

后，确定第 (2) 步中的参数向量规模，参照公式 (2) 在参数之间建立协作关系。

$$y_c(x) = \frac{1}{\sum_{i=1}^m f(x)} \left( \sum_{i=1}^n f(x) \left( \sum_{i=1}^{\min(m,n)} p_i x_i \right) \right) \quad (2)$$

式中， $y_c(x)$  表示参数协作处理后的网络信号。

4) 对第 3) 步的信号进行线性化处理，此处将驱动与参数协作相融合，有效去除网络信号的噪声干扰，有助于降低网络化管理复杂度。线性化处理后的多参数协作关系如公式 (3)，网络信号  $\overline{y_c}(x)$  详见公式 (4)。

$$\begin{cases} \mathbf{P}^i(x) = \bigcap_{j=1}^i p_j(x_i) \\ p_j(x_i) = \sum_{k=i}^j D_k \cdot x_k \end{cases} \quad (3)$$

式中， $\mathbf{P}^i(x)$  表示多参数协作矩阵， $p_j(x_i)$  表示单个数据流的参数驱动向量。

$$\overline{y_c}(x) = \frac{\left( \sum_{i=1}^n y_c(x_i) \right) \int \left( \sum_{i=1}^{\|D\|} p_i x_i \right)}{\sum_{i=1}^m f(y_c(x_i))} \quad (4)$$

5) 网络化管理模块对线性处理后的网络信号再次实施协作驱动，由式 5) 得到网络存储的实例对象，包括网络数据  $y$  和相应多参数向量  $p_m$ 。

$$\begin{cases} y = \frac{\sum_{i=1}^{\min(m,n)} d(x_i, p_i)}{(D)} \\ p_m = \min_{i=1, \dots, \min(m,n)} \{d(x_i, p_i)\} \end{cases} \quad (5)$$

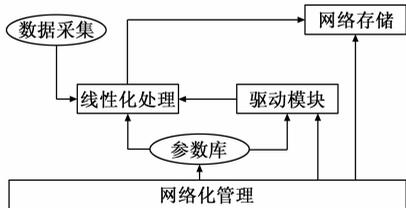


图 1 网络控制系统构成

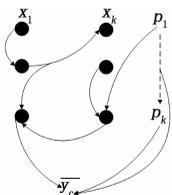


图 2 多参数协作驱动映射关系

图 2 给出了多参数协作驱动模块中实时更新的参数与网络信号之间的映射关系。其中， $k = \min \{m, n\}$ 。参数  $p_1$  到  $p_k$  之间的虚线箭头表示多参数向量，实线箭头表示参数与数据之间的驱动关系，最后生成了网络信号  $\overline{y_c}$ 。

## 2 控制系统设计

### 2.1 系统结构及原理

将上节设计的多参数协作驱动算法应用于智能网络化控制系统，硬件平台以网络数字信号处理为主。这样的硬件平台所支持的数据库和软件平台实现语言具有更好的交互性和拓展性。该系统基于无线传感器构建的网络平台，信号交互方面采

用 C# 语言实现，控制算法实现由 Java 语言实现，智能网络化控制应用由 Android 语言实现。该系统的软件既可以部署在计算机也可以部署在移动终端上。

该系统由核心控制部分和中继协作部分组成结构如图 3 所示。其中，核心控制子系统部署于服务器，作为智能网络平台的核⼼；中继协作子系统部署在中继终端。两个子系统之间基于多参数协作驱动算法实现实时交互性。中继协作子系统以网络化控制和信号传输为主，为核心控制子系统提供可靠的参数。图 3 中，圆角矩形表示核心控制子系统，包括信号序列、线性化处理、参数库、数据库、控制和网络化交互等部分。椭圆虚线内部表示由若干个中继协作终端组成的中继协作子系统，每个终端与核心控制子系统建立独立的点到点链接。

如图 3 所示，核心控制子系统的主要事务功能是参数检索与更新。参数检索功能被网络化需求事例所激活。该事例信号经过线性化处理成为数据库存储与控制对象。根据线性化处理规则，在信号强度与线性化权重之间进行融合处理，从数据库中寻找信号与网络化需求事例一致性最佳的参数序列，并按照网络化控制需求对参数序列进行归一化处理。

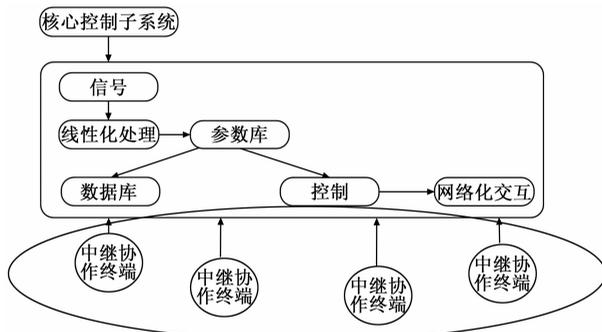


图 3 智能网络化控制系统组成

其中，数据检索与参数库查询的主要目的是线性化处理与参数序列归一化的映射度量。合理的映射度量有助于高效、精确地与网络化控制事例的建立实时交互。核心控制子系统的数据库与参数库的映射度量采用了基于信号强度和参数序列归一化相结合的方法。

该控制系统的信号管理采用了自适应信号输入输出端口的交互方法。该方法，通过自适应调度智能网络控制所需要的输入信号序列和强度，结合数据库与参数库的查询效率和一致性映射强度，实时求解网络化控制系统所需要的设计参数和参数序列。基于网络化控制事例的需求，在参数库中定义对应的多参数驱动列表，并建立最佳控制主键和参数排序方案。控制模块将核心控制子系统的网络化交互参数结构输出到控制系统。中继协作控制子系统获取网络化交互结构后，分发给每个中继协作终端。如果网络化控制效果达到需求，则将满意结果标记反馈给核心控制子系统，并确定参数库数据；如果无法满足需求，再次进入参数库，更新对应参数序列和参数库结构。依次进行上述循环，直至达到网络化控制事例需求。

网络化交互模块成为该系统中的核心控制子系统和中继协作子系统与应用需求之间的主要接口。该接口为整个系统的架构调整和更新提供驱动，并且该交互模块与外界应用保持实时连接，网络化控制事例在具体实施过程中，如果发现应用执行效率下降或遇到未知干扰，应用事例直接通过交互接口与该系

统建立通信，核心控制子系统和中继协作子系统及时被激活，并转入网络化控制进程。为了满足各类网络化控制应用事例，该系统的数据库和参数库结构采用动态布局。

### 2.2 系统实现

图 4 表示图 3 中的参数库详细阐述。其中，以数据库的信号序列为参照，建立参数库，以公式 (3) 求得的  $p$  序列为存储对象。核心控制子系统通过中继协作子系统实时更新参数库，以便保证多参数协作驱动的实时性和可靠性。参数库更新流程描述如下：

- 第 1 步：获取数据库和参数库在服务器端的存放路径；
- 第 2 步：从数据库读取数据表，以数据表文件为元素组成数组；
- 第 3 步：循环访问数组，从数据表文件中获取网络传输信号；
- 第 4 步：根据多获取的信号求出参数；
- 第 5 步：根据公式 (3) 对所求出的参数进行排序；
- 第 6 步：选出  $m$  与  $n$  中的最小值赋值给变量  $k$ ；
- 第 7 步：进行  $k$  次循环，每次循环按照公式 (5) 求解响应多参数值，并存入参数库；
- 第 8 步：按照公式 (5) 求出信号，并求和；
- 第 9 步：返回参数库指针  $P$  和信号和  $s$ 。

上述算法中， $P$  向量通过  $k$  次循环求得  $p$  组成的向量， $s$  表示网络化控制系统传递的网络信号。

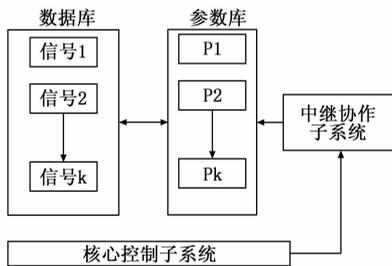


图 4 智能网络化系统参数库

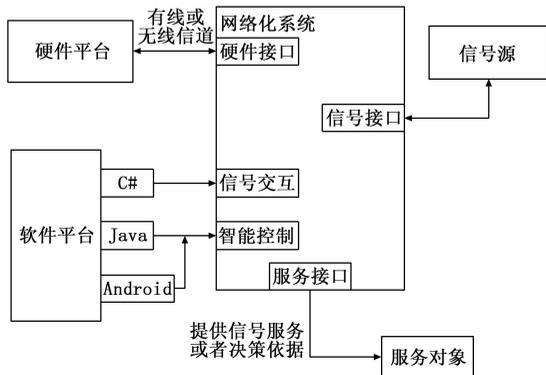


图 5 智能网络化系统外部应用接口

图 5 给出了智能网络化控制系统与外部硬件平台、软件平台、信号源和服务对象各类接口的连接。该系统的执行流程 INS with MPCD 的执行步骤描述如下：

- 输入：公共对象 DObject；
- 公共区域：
- 1) 多参数协作驱动算法清空参数库；
  - 2) 实时获取参数函数；

- 3) 智能网络化系统构造函数；
  - 4) 获取智能网络化系统实例函数；
  - 5) 声明硬件平台及其接口实例；
- 私有区域：
- 1) 初始化硬件集；
  - 2) 初始化软件集；
  - 3) 声明软件平台接口并激活开发接口实例；
  - 4) 声明信号源接口实例；
  - 5) 获取网络化服务对象；
  - 6) 提供或反馈服务决策数据；
- 循环结构：
- 1) 循环更新信号数据库；
  - 2) 循环更新参数库；
  - 3) 循环多参数协作控制。

### 3 系统性能验证

为了验证本文所设计的智能网络化系统的工作效率和数据精度，设计了一组实验，原系统运行统计数据与所设计系统记为 INS with MPCD 的性能进行对比。采用 VC++6.0 实现图 5 中的硬件接口与服务接口，部署在 2 台服务器上，使用 C# 实现信号交互接口，为传感器节点、汇聚节点和服务器之间的通信建立无线连接服务。采用 Android 和 Java 技术开发适用于中继节点的协作控制应用，包括参数库更新流程伪代码，参数库和数据库均采用 SQLite 部署，并位于汇聚节点和服务器端。无线传感器网络组成按照图 1 部署，并按照图 5 所示接口建立外围连接，服务应用为提供温度数据。此外，无线传感器网络平台参数详见表 1。

表 1 实验环境

参数	值	参数	值
传感器数	50 个	服务器 CPU	Intel Xeon E3 v3
温度传感器	CC2530	服务器硬盘	WD 1TB/7200 转
无线通信模块	IEEE 802.11ac	实验时间	70 分钟

图 6 给出了所设计的智能网络化系统的执行效率随着实验时间的变化情况，同时与原系统进行了对比。图 7 给出了原系统转发的数据、INS with MPCD 转的数据与实际发送数据的误差对比情况。因为 INS with MPCD 系统采用了多参数协作驱动，且基于网络化控制，从而具有较高的执行效率和数据精度。

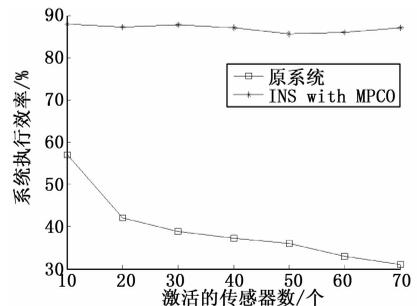


图 6 系统执行效率

### 4 结束语

无线数据传输效率和数据精度成为制约网络控制性能的关键因素，为了解决上述问题，本文设计一种基于多参数协作驱

(下转第 251 页)