

DOI: 10.19666/j.rlfd.202505107

DCS 控制组态数据一致性与同步技术研究

宋美艳, 高少华, 薛建中, 胡 波, 刘 畅, 张 军
(西安热工研究院有限公司, 陕西 西安 710054)

[摘 要] 为解决传统分散控制系统 (DCS) 在复杂工业场景中因控制组态数据异步演化引发的系统失稳风险, 构建了覆盖装备全生命周期的动态同步技术体系。针对工业过程控制逻辑偏移的深层原因, 系统分析了 DCS 控制组态数据在转换与同步环节的潜在风险传导机理, 提出了基于动态校验与全链路溯源的同步保障机制。通过构建镜像数字孪生映射模型, 实现物理控制器与上位机的组态数据双向映射, 结合双态协同闭环同步协议栈, 形成“源-存-运”三位一体的技术架构。该体系突破传统同步模式的技术局限, 在复杂操作条件下仍可确保控制组态数据的强一致性维护。经火电厂 DCS 工程实践验证, 所提技术方案显著增强了控制系统对异常工况的容错能力, 有效保障发电机组安全稳定运行, 为能源领域关键信息基础设施的自主可控升级提供了重要技术支持。该研究成果对同类工业控制系统具有重要参考价值, 其设计理念可推广至流程工业、智能制造等多领域场景。

[关 键 词] 分散控制系统; 组态数据; 同步技术; 一致性

[引用本文格式] 宋美艳, 高少华, 薛建中, 等. DCS 控制组态数据一致性与同步技术研究[J]. 热力发电, 2026, 55(1): 160-168.
SONG Meiyan, GAO Shaohua, XUE Jianzhong, et al. Research on data consistency and synchronization technology of DCS control configuration[J]. Thermal Power Generation, 2026, 55(1): 160-168.

Research on data consistency and synchronization technology of DCS control configuration

SONG Meiyan, GAO Shaohua, XUE Jianzhong, HU bo, LIU Chang, ZHANG Jun
(Xi'an Thermal Power Research Institute Co., Ltd., Xi'an 710054, China)

Abstract: To address the risk of system instability in conventional distributed control systems (DCS) in complex industrial settings which is caused by asynchronous evolution of control configuration data, a dynamic synchronization technology system covering the entire lifecycle of equipment is developed. It delves into the potential risk transmission mechanisms of DCS configuration data during conversion and synchronization, and presents a synchronization assurance mechanism based on dynamic verification and full-chain tracing. By creating a mirrored digital twin mapping model, it enables two-way mapping of configuration data between physical controllers and upper-computer systems. Together with a dual-state cooperative closed-loop synchronization protocol stack, this forms a triple-integrated architecture of “source-storage-operation”. Breaking through the limitations of conventional synchronization modes, this system ensures strong consistency of configuration data even under complex operating conditions. Verified in a thermal power plant's DCS project, the proposed technical solution significantly improves the control system's fault-tolerance under abnormal conditions, effectively ensuring safe and stable operation of power generation units. This offers key technological support for the independent and controllable upgrade of critical information infrastructure in the energy sector. The research holds great reference value for similar industrial control systems, and its design concept can be applied to various fields such as process industry and smart manufacturing.

Key words: distributed control system; configuration data; synchronization technology; consistency

收稿日期: 2025-05-22 修回日期: 2025-07-13 接受日期: 2025-07-16 网络首发日期: 2025-10-24

基金项目: 中国华能集团有限公司总部科技项目 (HNKJ25-H08)

Supported by: Science and Technology Project of China Huanneng Group Co., Ltd. (HNKJ25-H08)

第一作者简介: 宋美艳 (1979), 女, 高级工程师, 主要研究方向为工业控制系统研发与应用, songmeiyan@tpri.com.cn.

分散控制系统（DCS）的核心是通过控制组态实现生产工艺的精准自动化，其本质是构建“策略源码-编译代码-运行实例”三层映射的数据闭环。工程师通过上位机组态工具将工艺需求转化为结构化逻辑，经编译生成控制器可执行指令，最终通过 I/O 通道实现闭环控制。然而，组态数据在上位机与控制器间的非对称流动形成潜在风险：组态更新滞后或控制器异常回溯会导致控制逻辑与物理系统失配，这种隐蔽性偏差在连续生产场景中，可能引发参数漂移、甚至联锁失效等问题，严重威胁工业生产安全。

近年来，国内外在控制系统的相关研究上取得了显著进展^[1-2]。同时，数字孪生技术^[3-4]的蓬勃发展为 DCS 持续进化注入了新的活力。本团队在国产化平台研发背景下，已开展了一系列技术攻关，成功开发出适配国产系统的控制系统组态软件^[5]，并对初步的组态数据同步机制进行了积极探索^[6-7]，这些成果为后续的深入研究筑牢了坚实的基础。

基于对传统 DCS 控制组态数据同步机制局限性的深入剖析，结合当前技术发展以及前期工作成果，本文提出了“源码态-存储态-运行态”三位一体数据同步体系，力求全方位突破传统架构所面临的瓶颈，构建起一套完整且高效的全链路同步技术体系。具体而言，本体系从动态校验与溯源、镜像数字孪生映射以及双态协同闭环同步协议栈这三个关键维度入手，系统性地攻克控制组态数据在上位机与控制器间一致性的难题，切实提升 DCS 在复杂工业场景下的运行可靠性与数据一致性保障水平，为工业生产的安全稳定运行提供坚实的技术支撑与保障。

1 控制组态数据不一致性机理分析

控制组态数据的一致性工业过程安全稳定运行的核心要素。然而，由于 DCS 架构的分层特性与工程实践的动态需求，组态数据在“源码态、存储态、运行态”三态间的异步演化已成为影响系统可靠性的关键问题。深入剖析三态数据的转换机理与异步演化规律，是构建高可靠性 DCS 的理论前提，也是制定数据同步策略、优化组态管理流程的实践基础。本文将从三态数据的基本定义出发，系统阐述其转换流程，深入分析异步演化导致的不一致性表现、成因及潜在风险，为构建高可靠的 DCS 组态管理体系提供理论支撑。

1.1 三态数据定义

源码态是指上位机工作站中保存的组态工程文件，包含了用户定义的所有控制逻辑组态。源码态是系统设计和配置的蓝图，通常以可视化的方式呈现，便于工程师理解和修改。

存储态是源码态经过编译后生成的组态工程产物，存储在下位机控制器中。存储态包含了控制器执行控制算法所需的所有信息，是源码态的二进制表示，专为控制器执行优化。

运行态是指控制器在运行过程中，根据存储态生成的实时控制数据。运行态包含了系统的当前状态、输入输出值和报警信息等。运行态直接反映生产过程的实时状态，是控制逻辑的实际执行载体。

1.2 三态数据的转换流程

在典型的 DCS 中，三态数据的转换流程如下。

1) 源码态创建和修改 工程师在上位机使用可视化编程工具创建或修改组态工程，定义控制策略。

2) 源码态编译和下载 组态配置完成后，上位机将源码态编译生成存储态，并通过通信网络将存储态下载到下位机控制器。

3) 存储态加载和执行 控制器接收到存储态并保存后，将其加载并解析到内存中，随后启动控制逻辑执行，生成运行态数据^[8]。

4) 运行态监控和调整 上位机通过通信网络实时监控运行态数据，并根据需要调整控制参数或修改组态配置。

1.3 三态数据异步演化问题

在实际的 DCS 中，源码态、存储态和运行态常呈现异步演化特征，即三态的修改与更新存在时间差。这种异步演化导致三态数据之间存在不一致，进而引发系统性风险。

1.3.1 三态数据不一致主要表现类型

1) 源码态与存储态不一致 源码态已修改但未下装至控制器，导致控制器执行逻辑与最新源码态脱节。

2) 存储态与运行态不一致 控制器已加载新存储态，但运行态未完全更新，造成控制策略执行偏差。

3) 源码态与运行态不一致 源码态已修改但未通过下装影响运行态，或运行态被直接修改但未同步至源码态，形成跨层数据断层。

1.3.2 三态数据异步演化成因分析

三态数据异步演化问题的本质是 DCS 分层架构下数据更新时序的非对称性,其成因可归纳为工程实践中的 3 类典型操作场景。

1) 源码态-存储态失配机理

这类不一致性源于组态变更的“设计-部署”流程断裂。当工程师完成源码态修改后,若未执行编译下装操作,控制器将持续运行历史版本的存储态。其技术本质是 DCS 组态变更需通过“编译-下装-加载”的严格流程实现策略迁移。而工程实践中常因版本管理疏漏或操作流程缺失,导致源码态与存储态产生版本断层。

2) 存储态-运行态失配机理

这类不一致性反映控制器运行时的动态调整需求与传统 DCS 架构的固有矛盾,主要源于 2 种操作模式。

在线参数整定 通过 DCS 厂商专有协议(如私有二进制协议)直接修改控制器内存中的运行态参数,形成“旁路式”调整。这种操作绕过存储态更新机制,导致运行态与存储态产生瞬时偏差。

选择性更新机制 部分 DCS 厂商控制器运行时采用非原子性加载机制,导致存储态更新期间运行态处于中间过渡状态,形成瞬时性版本错位。

3) 源码态-运行态失配机理

这类不一致性打破了传统 DCS 架构中“源码-存储-运行”的单向数据流动约束,通过隐性数据传递链重构了跨层映射关系,其技术本质可分解为 2 个维度。

旁路式干预 通过在线命令强制修改运行态参数(如强制、置数操作),但未在源码态建立对应变更记录,形成“游离态”运行数据。

隐性传导 该路径本质是源码态-存储态失配的衍生现象,源码态修改后未执行编译下装操作,导致控制器持续运行历史版本存储态。

1.4 三态数据的不一致可能导致问题

1) 控制策略执行偏差 存储态或运行态未及时更新,使控制器执行逻辑与源码态脱节,导致控制策略执行偏差,影响生产过程稳定性。

2) 系统振荡与事故 三态数据失配可能引发控制回路振荡,极端情况下导致生产事故。

3) 维护难度增加 数据不一致会干扰故障诊断,提升系统维护成本。

4) 监控失效风险 上位机与控制器数据不一致可能导致监控系统误判,威胁系统安全性。

2 现有技术及其局限性分析

传统 DCS 三态数据同步如图 1 所示。由图 1 可见,当前主流 DCS 通过源码态、存储态、运行态的转换实现控制逻辑的部署与执行,主要采用传统单向同步模式作为基础架构,部分先进 DCS 在此基础上升级扩展了双向闭环同步能力(红色框内表示运行态到源码态的逆向同步)。实际应用中,这 2 种同步模式虽然能够实现基础的控制组态数据管理,但受限于三态转换机制的本质特性,仍然存在三态数据不一致性问题,下面将分别进行详细分析。

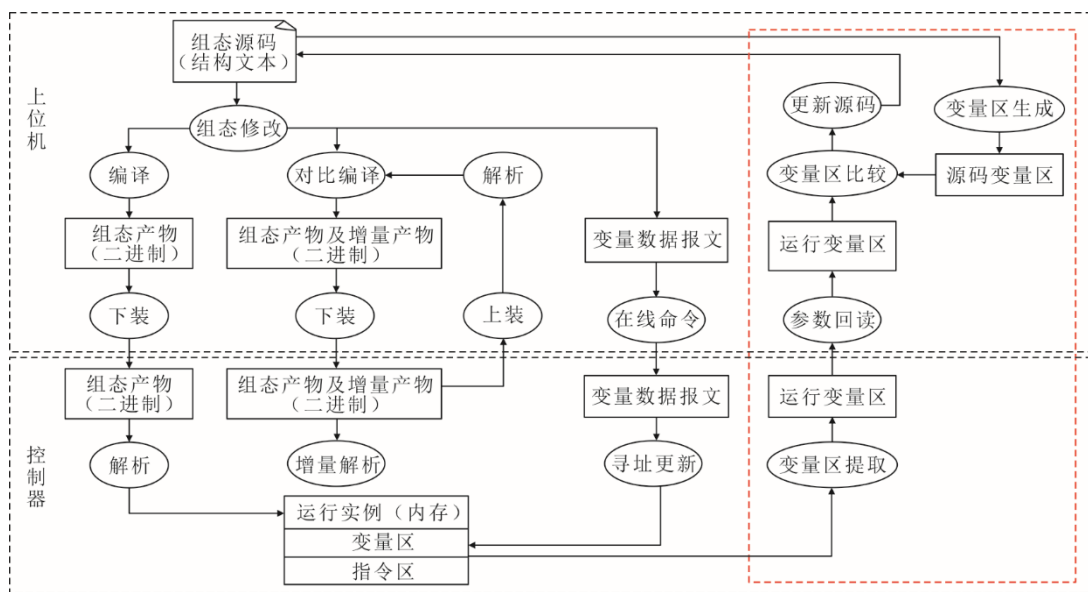


图 1 传统 DCS 三态数据同步

Fig.1 Conventional DCS three-state data synchronization

2.1 传统单向同步模式

传统单向同步模式是一种从源码态到运行态的单向数据流模式。在这种模式下,工程师在上位机完成组态设计或修改后,通过编译生成存储态(组态产物或组态+增量产物),再下装至控制器转换为运行态执行。此过程仅支持从设计到运行的单向传递,无法有效反馈运行态的现场修改。

具体来说,这种模式有2种存储态设计方式:

方式1(组态产物) 存储态仅包含完整组态,控制器需内置增量比较逻辑,实时解析组态差异并更新运行态,对控制器算力要求较高。

方式2(组态+增量产物) 存储态包含完整组态及增量包,控制器完成初次下装后仅需解析增量,将增量比较逻辑上移至上位机,可降低控制器负荷。

这种传统单向同步模式也存在一些局限性:

反馈缺失 运行态的现场调优(如通过在线命令对功能块参数调整)无法自动同步至源码态,需人工反向操作,易引发版本不一致。

控制器依赖 增量比较逻辑的位置(控制器或上位机)影响系统扩展性,如增量比较与解析在控制器侧,则会增加控制器负荷,若控制器解析能力不足,可能限制组态复杂度。

修改差异不可追溯 源码态修改后,缺乏自动化的差异比对机制,导致工程师无法直观追踪组态变更内容,导致关键逻辑调整被遗漏的概率增加。此外,当源码态需要回退至存储态版本时,只能依赖人工备份恢复,这种恢复方式不仅操作复杂,而且容易在恢复过程中引入新的错误。

2.2 双向闭环同步模式

双向闭环同步模式在传统单向同步模式的基础上,增加了一种反馈机制,即通过参数回读功能,将运行态变量区与上位机组态变量区自动进行比对,差异可反向更新至源码态,再经编译下载后更新存储态,形成完整闭环。具体来说,这种模式的特点是现场调优固化,工程师在运行态的临时修改(如PID参数调整)可回传至源码态,避免手动重写组态,有效解决了传统单向同步模式下反馈缺失的问题。然而,这种双向闭环同步模式也存在一些安全风险和模式失效问题。

差异处理的单向性 双向闭环同步模式在差异处理逻辑上呈现本质的单向性特征。当通过参数

回读功能比对运行态与源码态的差异时,系统仅支持“运行态→源码态”的单向覆盖策略。这种设计在支持在线命令修改的场景中形成特定风险,运行态通过临时指令完成动态优化的同时,若工程师对源码态进行了修改,通过参数回读操作可能触发源码态被运行态临时修改逆向覆盖的风险,导致源码态的修改丢失。

模式失效 当源码态进行较大规模的修改(非在线命令修改)时,双向闭环同步模式可能面临失效风险。由于运行态是由存储态解析生成的,而存储态又直接依赖于源码态的编译产物,因此源码态的大修改可能破坏现有的同步链路。这种破坏可能导致闭环同步流程中断,使得模式无法正常工作。

传统单向同步模式以稳定性为核心,适用于组态固定的场景,但其缺乏反馈机制,容易导致版本不一致和控制策略执行偏差;双向闭环同步模式在单向同步模式的基础上虽然引入了反馈机制,但需配套严格的变更管理流程,又面临着安全风险和模式失效等问题。

3 关键技术创新

本研究基于DCS控制组态的“源-存-运”全链路数据环境特征,针对1.3.2节所述的源码态-存储态、存储态-运行态以及源码态-运行态3类失配场景机理分析,结合既有技术优化方向,又构建了覆盖组态数据全生命周期的“三位一体”保障体系。该体系通过动态校验与溯源机制、镜像数字孪生映射机制、双态协同闭环同步协议栈三大技术创新,形成从数据源头治理到运行态校正的完整技术链,有效解决工业控制场景下组态数据跨生命周期一致性保障难题。

3.1 动态校验与溯源机制

针对源码态-存储态失配场景,提出基于“编译体-源文件”深度绑定的动态校验与溯源机制。图2为动态检验CRC与溯源对比。如图2可见,该机制通过构建编译阶段与运行阶段的双向关联通道,突破传统CRC校验^[9]“知错难溯”的技术瓶颈,形成“校验-溯源-比对-更新”的完整闭环保障链路,同时突破传统单向同步模式只能单向固化修改的局限性,构建起从存储态到源码态的反向更新通道,确保源码态修改后能够准确反映到存储态,避免因编译下装操作未执行造成的版本断层。

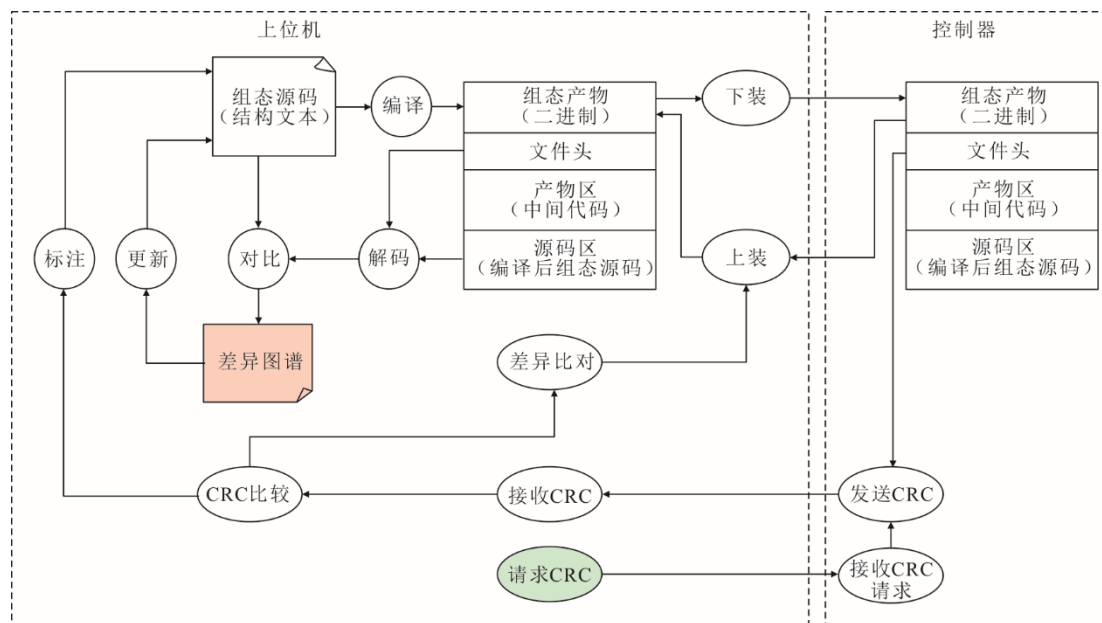


图2 动态检验 CRC 与溯源对比

Fig.2 Dynamic examination of CRC and traceability comparison

3.1.1 一体化编译载体构建

重组态产物数据结构，设计双模态数据封装架构，在编译生成产物区（中间代码）时实施双层数据封装策略。

1) 源数据封装 将组态源码的全量元数以树状结构形式封装至产物区尾部，形成独立的源码区模块。该模块完整保留原始组态工程的语义信息，为后续溯源分析构建全量数据档案库，确保任何修改均可追溯至具体工程对象，避免因版本管理疏漏导致的源码态-存储态失配问题。

2) CRC 校验值植入 在组态产物的文件头区域，植入动态生成的 CRC 校验值。该校验值基于源码区全量元数据计算生成，为后续版本一致性监测提供基础数据支撑，实现修改行为的原子化记录，以便在源码态修改后，能够及时发现与存储态的不一致情况，确保两者的一致性。

3.1.2 实时版本监测体系

建立基于 CRC 校验的实时比对机制。

1) 编译阶段 上位机在生成组态产物时，通过标准 CRC 校验算法生成校验码，并写入产物文件头部的固定区域，确保源码态修改后的组态产物能够被准确校验。

2) 运行阶段 当上位机与控制器建立连接后，采用周期性同步策略主动获取控制器端组态产物文件中的 CRC 值。

3) 比对策略 通过实时比对上位机产物文件的 CRC 值与当前控制器产物文件的 CRC 值，构建版本一致性监测体系。当检测到校验偏差时，系统自动对上位机组态页的在线状态进行可视化标注^[10]，以提示用户可能存在源码态-存储态失配情况，及时采取措施解决。

3.1.3 双向溯源对比机制

设计差异溯源与同步更新流程：

1) 用户触发比对 当用户选择执行比对操作时，上位机自动请求并获取控制器中的组态产物文件，以确定源码态与存储态是否存在不一致情况。

2) 源码区解析 对控制器组态产物中的源码区进行结构化解析，提取封装的全量元数据，为后续差异分析提供准确依据。

3) 差异分析 将控制器解析后组态数据与上位机组态数据进行逐项对比，通过差异标记引擎定位不一致项，生成结构化差异报告，明确指出源码态修改后未反映到存储态的具体内容。

4) 可视化呈现 通过差异报告生成组态数据差异图谱，以分级显示方式突出显示存在偏差的变量配置、逻辑连接等关键信息，使用户能够直观地了解源码态-存储态失配的具体情况。

5) 同步修正 提供两种修正模式。一种为手动修正，用户基于差异图谱直接修改上位机组态文件。另一种为自动同步，通过上装（从控制器获取

更新)或下装(向控制器推送修正)操作,实现源码态与存储态的组态数据强制一致,确保源码态修改能够准确地反映到存储态,避免因操作流程缺失导致的版本断层问题。

3.2 镜像数字孪生映射机制

镜像数字孪生映射机制通过构建上位机组态模型与控制器运行实例的镜像关联体系,针对存储态-

运行态失配及源码态-运行态失配场景,实现控制参数在源码层与运行层特征提取与动态比对。图3为数字孪生模型构建与对比。该机制以“源码态与运行态一致性”为核心目标,建立跨层数字孪生模型,通过上位机主导跨层建模比对机制,有效减轻控制器实时运算压力,同时突破异构数据模型下组态数据比对难题,确保源码态-运行态的一致性。

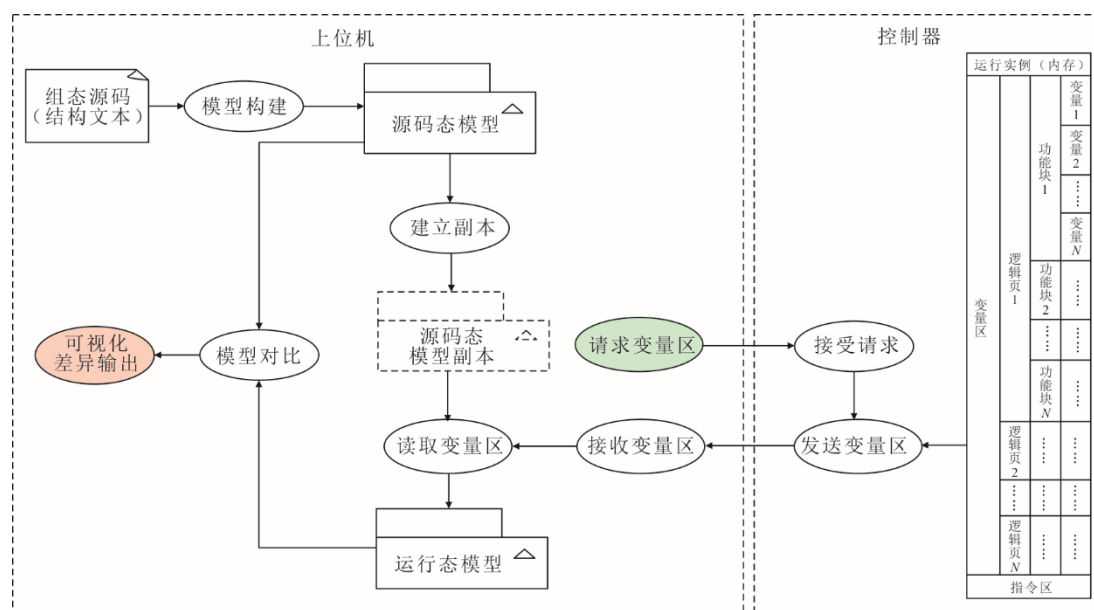


图3 数字孪生模型构建与对比

Fig.3 Construction and comparison of digital twin models

3.2.1 一致性前提约束

建立镜像数字孪生映射需满足以下前置条件:

1) 源码态完整性校验 通过3.1节所述动态校验机制与溯源机制,确保上位机组态文件与控制器存储区数据的强一致性,作为数字孪生体建模的基准数据源。

2) 镜像关联基础 仅当存储态数据与源码态完全同步时,方可启动孪生模型构建流程,避免因基础数据不一致导致映射误差累积,在反映源码态与运行态差异同时也反映了存储态与运行态的差异。

3.2.2 数字孪生建模流程

构建双向映射模型需执行以下标准化流程。

1) 源码态数字孪生体构建

语法解析与元数据提取 对组态工程文件执行语法解析操作,深入分析其内部结构,识别各类语法元素与逻辑规则,最终精准提取所有变量定义表,该表涵盖变量数据类型、变量地址、变量值等

关键元数据。其中,变量数据类型包括整数、浮点数、布尔值等多种类型,用于界定变量的存储与运算方式;变量地址是变量在系统中的唯一标识,用于指引数据的存储位置;变量值则反映变量在当前状态下的具体取值。元数据全面且准确地描述组态工程文件中的变量,为数字孪生体的构建提供可靠的数据支持。

数字孪生体基准模型构建 基于提取的元数据,开展结构化数字孪生体模型的构建工作。在构建过程中,严格遵循变量的层级关系,按照“逻辑页-功能块-变量”的对应顺序与逻辑架构进行搭建,确保所构建的数字孪生体在逻辑架构上与控制器运行态的架构高度一致。通过这种方式形成的数字孪生体模型,能够准确地反映物理实体的组成结构以及各组成部分之间的逻辑关系,为后续实现物理实体与模型之间的双向映射和数据交互提供了精准的逻辑框架。

2) 运行态物理孪生体构建

基准模型副本创建 以源码态数字孪生体模型为原始蓝本,复制其结构与属性,生成一个初始的基准模型副本。该副本在逻辑架构上与源码态模型保持高度一致,为后续数据映射提供可靠载体,确保运行态能够准确反映源码态的修改。

物理孪生体模型构建 读取控制器运行实例数据中的变量区数据,同时对基准模型副本中的变量定义表进行逐项遍历。根据变量地址的匹配,将控制器中对应的变量实时值写入副本的相应位置。完成所有变量值的映射更新后,形成运行态物理孪生体模型,实现与物理实体状态动态同步,为数字孪生应用提供虚拟映射基础,及时发现并解决源码态-运行态失配问题。

3.2.3 变量级差异比对

设计变量比对引擎,通过结构化校验实现源码态-运行态的精准差异定位,包含以下关键技术。

1) 双模型同步遍历机制

以源码态数字孪生体模型为校验基准,同步遍历运行态物理孪生体模型。通过变量地址定位 2 个模型中的对应变量,并对比变量值,自动记录差异变量的变量名、源码值、运行值 3 列对比数据,形成结构化差异分析表,以便及时发现源码态-运行态的不一致情况。

2) 可视化追溯系统

基于结构化差异分析表,使源码态与运行态的差异得以全面且直观地呈现,其具体功能如下:

差异图谱 根据差异分析表构建差异视图,直观展示组态数据的差异,便于用户迅速定位差异,便于用户迅速定位源码态-运行态的不一致情况。

过滤引擎 支持按变量类型、修改时间等多维度过滤差异数据表,助力用户聚焦源码态-运行态的关键差异,提升分析效率。

组态图联动 点击差异变量可直接跳转至对应功能块逻辑图,实现空间定位与可视化溯源,进而深入探究源码态-运行态不一致的根源。

3.3 双态协同闭环同步协议栈

针对存储态-运行态失配以及源码态-运行态失配场景,双态协同闭环同步协议栈集成参数回读与参数回写双通道机制(图 4),构建闭环数据校正体系,该体系紧密衔接数字孪生体的构建及其差异比对结果,实现控制参数在线无损同步,确保源码态与运行态、存储态与运行态数据强一致性。该协

议栈通过双向数据流管控,实现“设计-运行”双态的动态均衡,同时与动态校验与溯源机制、镜像数字孪生映射形成技术闭环,从而有效规避传统双向闭环同步模式的失效问题,解决因在线参数整定、选择性更新机制以及旁路式干预等操作导致的瞬时偏差和版本错位问题。

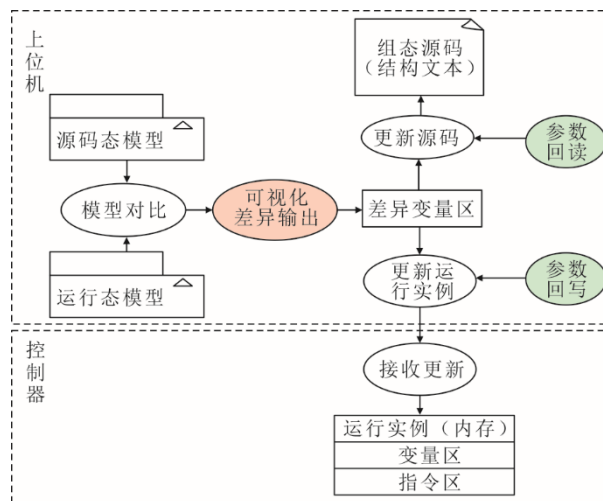


图 4 双态协同闭环同步参数回读与参数回写协议栈
Fig.4 Parameter read back and parameter write back protocol stack

3.3.1 参数回读通道

1) 组态反写

基于变量级差异比对所生成的结构化差异分析表,上位机组态编程系统自动触发增量补丁文件的生成。利用源码态与运行态数字孪生体,通过深度比对运行态数据与源码态数据,精准定位差异点,并据此生成包含差异部分数据的增量补丁文件。随后,通过合并逻辑,将增量补丁文件无缝整合至组态工程文件中,实现运行参数向源码态的逆向同步。在合并过程中,系统严格遵循组态工程的结构规范,确保数据的准确性和一致性,同时详细记录同步操作的日志信息,以便后续的追踪与审计,保障整个反写过程的可追溯性与可靠性。

2) 校验保障

为确保回读参数符合组态工程规范,参数回读通道实施了严密的校验保障措施。首先,对回读参数进行语法层面的校验,检查参数格式、数据类型等是否符合组态工程的编译要求,及时发现并纠正潜在的语法错误,保障参数在组态工程中的可编译性。其次,运用语义检查技术,深入分析回读参数的语义信息,验证其是否与组态工程的逻辑语义相匹配,确保参数在实际运行中的合理性和有效性。

参数回读完成后,通过控制器下装更新存储态,实现源码态与存储态的组态数据强制一致。

3.3.2 参数回写通道

1) 事务性写入

同样基于变量级差异比对所生成的结构化差异分析表,将变更后的参数通过事务性写入机制正式写入控制器的运行实例。为确保更新过程的数据一致性,避免因写入中断或并发冲突导致部分参数更新失败、数据错乱,系统采用分段锁定协议。在写入过程中,将控制器内存划分为多个数据段,对每个数据段进行锁定操作,确保在同一时间段内只有一个写入任务能够对特定数据段进行操作。系统将参数按照预定的规则和顺序,通过控制器的内存映射接口批量写入运行实例。批量写入操作在保证数据完整性的同时,可显著提高写入效率,减少对控制器运行的干扰。事务性写入机制对参数变更进行严谨的操作和管理,确保每一次写入都能够顺利完成,保障控制器运行态数据的准确更新。

2) 动态适配

在参数写入过程中,系统实时监测控制器的运行状态,根据其运行节奏和负载情况,动态调整参数写入的策略和节奏。确保在参数更新过程中,控制器能够持续稳定地执行控制任务,对生产过程不产生任何可见的中断或扰动。动态适配技术使得控制系统能够在不影响生产的前提下,及时响应组态工程的变更需求,保持源码态与运行态的一致性。

4 工程实践验证

4.1 实践场景

在某大型火电厂 1 000 MW 机组的实际 DCS FAT(工厂验收测试)期间,对所提出的控制组态数据一致性保障方法进行了全面的实践验证。该 DCS 由 50 对冗余控制器构成,控制器具备“选择性更新机制”,即功能块参数修改需通过在线命令,下装操作不会对已存在功能块参数进行更新。

在 DCS 上位机侧部署所开发的控制组态数据一致性同步工具,依据三态数据异步演化的成因分析,设计了 3 类共 9 组典型变更场景进行验证,其中 3 类场景采用典型的控制组态异常变更操作,具体包括源码态的逻辑组态页修改、在线参数整定、离线参数修改,每类场景通过选取关键控制系统控制器作为验证对象,如协调控制系统(CCS)、汽轮机调节系统(DEH)和锅炉燃烧系统(BMS)。每组

场景均独立重复 10 次,每次累计 200 次增量修改,以形成具有统计意义的样本量,旨在检验所提出技术在不同操作场景下的适用性和可靠性。详细验证场景设置见表 1。

表 1 验证场景设置
Tab.1 Validation scene settings

场景	验证系统	重复次数	修改量
逻辑组态页修改	CCS	10	200
	DEH	10	200
	BMS	10	200
在线参数整定	CCS	10	200
	DEH	10	200
	BMS	10	200
离线参数修改	CCS	10	200
	DEH	10	200
	BMS	10	200

具体验证步骤如下:

1) 逻辑组态页修改场景 在上位机组态软件中完成 200 次增量逻辑变更(功能块增/删、连接关系调整),仅执行“保存”不触发“下装”。随后启用动态校验与溯源功能,对源码态-存储态进行差异比对与自动同步。记录每次检测到的差异条目。

2) 在线参数整定场景 保持控制器在线,在上位机组态软件中实时修改 200 次功能块参数,操作后不保存工程。关闭并重启组态软件,通过参数回读功能完成源码态-运行态/存储态-运行态一致性校验,统计差异条目与参数回读耗时。

3) 离线参数修改场景 断开控制器,执行 200 次功能块参数修改并保存工程,随后执行“下装”操作。在下装后启动参数回写功能,实现源码态-运行态/存储态-运行态一致性校验,统计差异条目与参数回写耗时。

4.2 验证效果

表 2 汇总了 3 类场景 9 组验证的全部差异检测与同步数据。表 2 中:理论差异数=验证系统数×重复次数×单次修改量;同步成功率=检测差异数/理论差异数;偏差率=(理论差异数-检测差异数)/理论差异数。总体样本量为 54 000 次差异事件(3 系统×3 场景×10 次×200 处修改)。在所有验证中,所开发的控制组态数据一致性同步工具均实现 100% 差异捕获率与零偏差,表明其在源码态-存储态与源码态-运行态/存储态-运行态双向同步中的完备性。在同步性能方面,在线参数整定的参数回读平均耗时 2.5 s,离线参数修改的参数回写平均耗时

2.3 s, 将传统人工核对同步由小时级压缩至秒级。

表 2 验证结果
Tab.2 Validation results

场景	验证系统	理论差异数	检测差异数	同步成功率/%	偏差率/%	回读时间/s	回写时间/s
逻辑组态页修改	CCS	6 000	6 000	100	0		
	DEH	6 000	6 000	100	0		
	BMS	6 000	6 000	100	0		
在线参数整定	CCS	6 000	6 000	100	0	2.7	
	DEH	6 000	6 000	100	0	2.3	
	BMS	6 000	6 000	100	0	2.5	
离线参数修改	CCS	6 000	6 000	100	0		2.5
	DEH	6 000	6 000	100	0		2.1
	BMS	6 000	6 000	100	0		2.3

通过在某大型火电厂 DCS FAT 期间的实践验证, 所提出的控制组态数据一致性保障方法在多个关键指标上均表现出色。该方法显著提升了组态数据的一致性, 有效解决了三态数据的失配问题, 确保了三态数据的一致性。同时, 参数回读与参数回写的高效同步大幅提升了运维效率, 减少了故障排查时间。这些结果充分证明了所开发工具在实际工程应用中的有效性, 为火电厂 DCS 的稳定运行提供了有力的技术支持。

5 结 语

本研究针对 DCS 组态数据一致性管理的行业共性难题, 系统性构建了“源-存-运”三位一体协同保障体系, 通过动态校验与溯源机制、镜像数字孪生映射及双态协同闭环同步协议栈三大核心技术突破, 实现了控制参数从设计到运行的全生命周期精准管控。该体系突破传统同步模式的技术局限, 形成了覆盖组态设计、编译下装、运行监控的闭环校正技术链, 为复杂工业场景下的安全稳定运行提供了多层防护保障体系。

【参 考 文 献】

- [1] 裴坤, 刘志勇, 吴欣. DCS 组态在线下载技术分析与设计[J]. 自动化与仪表, 2016, 31(5): 62-65.
QIU Kun, LIU Zhiyong, WU Xin. Analysis and design of on-line downloading technique for DCS configuration[J]. Process Automation Instrumentation, 2016, 31(5): 62-65.
- [2] 徐建飞, 罗昊, 陈通, 等. MELTAC 平台安全级 DCS 在线下装技术研究[J]. 核动力工程, 2013, 34(增刊 2): 70-72.

- XU Jianfei, LUO Hao, CHEN Tong, et al. Research on software online downloading technology based on MELTAC 1E-DCS system[J]. Nuclear Power Engineering, 2013, 34(Suppl.2): 70-72.
- [3] 刘墨迪, 刘鑫宇. 数字孪生技术在工业生产中的应用研究[J]. 中国机械, 2024(31): 71-75.
LIU Modi, LIU Xinyu. Application research of digital twin technology in industrial production[J]. China Machinery, 2024(31): 71-75.
- [4] 唐昕, 孙宇菲, 王钰珏, 等. 面向工业数字孪生的三层知识图谱结构设计方法[J]. 计算机科学, 2024, 51(增刊 1): 63-68.
TANG Xin, SUN Yufei, WANG Yujue, et al. Three layer knowledge graph architecture for industrial digital twins[J]. Computer Science, 2024, 51(Suppl.1): 63-68.
- [5] 胡波, 李响, 陈宏君, 等. 全国产分散控制系统组态软件框架的设计和实现[J]. 热力发电, 2021, 50(12): 13-20.
HU Bo, LI Xiang, CHEN Hongjun, et al. Design and implementation of configuration software architecture for domestically make distributed control system[J]. Thermal Power Generation, 2021, 50(12): 13-20.
- [6] 高少华, 樊启祥, 薛建中, 等. 一种分散控制系统离线组态对比方法和系统: CN113778038B[P]. 2022-02-11[2025-04-12].
GAO Shaohua, FAN Qixiang, XUE Jianzhong, et al. Method and system for offline configuration comparison of distributed control systems: CN113778038B[P]. 2022-02-11[2025-04-12].
- [7] 高少华, 刘畅, 张军, 等. 一种工业控制系统控制器组态参数回读方法、系统及设备: CN117032196A[P]. 2023-11-10[2025-04-12].
GAO Shaohua, LIU Chang, ZHANG Jun, et al. Method, system, and device for reading back configuration parameters of industrial control system controllers: CN117032196A[P]. 2023-11-10[2025-04-12].
- [8] 王德康, 王为民, 裴坤, 等. 集散控制系统嵌入式实时高级语言的设计与实现[J]. 化工自动化及仪表, 2001(1): 41-46.
WANG Dekang, WANG Weimin, QIU Kun, et al. Design and implementation of embedded real-time advanced language for distributed control systems[J]. Chemical Automation and Instrumentation, 2001(1): 41-46.
- [9] 朱正鹏, 朱旭峰, 李宾, 等. 一种位宽可变的 CRC 校验算法及硬件实现[J]. 航天控制, 2019, 37(2): 42-48.
ZHU Zhengpeng, ZHU Xufeng, LI Bin, et al. Data width variable CRC verification algorithm and hardware implementation[J]. Aerospace Control, 2019, 37(2): 42-48.
- [10] 薛建中, 宋美艳, 张军, 等. 分散控制系统逻辑组态监控方法、系统、设备及介质: CN117006476A[P]. 2023-11-07[2025-04-12].
XUE Jianzhong, SONG Meiyuan, ZHANG Jun, et al. Monitoring method, system, device and medium for logical configuration of distributed control system: CN117006476A[P]. 2023-11-07[2025-04-12].

(责任编辑 杜亚勤)