 DOI: https://doi.org/10.12349/iser.v7i1.9035

The embedded-Software Online Incremental updating scheme for Satellite embedded devices

Tianlong Sun

The 10th Research Institute of CETC, Chengdu, Sichuan, 610036, China

Abstract

To Adapt the changing application requirement In Satellite Communication field，providing the online update service is an absolutely effective method to improve the performance of software updating. The embedded software of Satellite terminal device's updating is constrained by the resource of satellite's communication channel, the stability of transmittal, speed, the capacity of storage and the processing, and so on, therefore, the article is trying to design a kind of Online Incremental updating scheme to resolve the above problems.

Keywords

Dicdiff algorithm; Architecture of dictionary; Sub-Sequence Search; Suffix tree; Update online

基于动态数据字典的卫星终端嵌入式软件 OTA 方案

孙天龙

中国电子科技集团公司第十研究所，中国 · 四川 成都 610036

摘　要

在卫星通信领域，为适应不断变化的应用需求，对分布在广泛地理区域的卫通星载设备、卫星终端提供软件远程在轨升级服务，可大幅节省设备维护成本，同时有效提升软件升级效能。卫通星载设备和卫星终端嵌入式软件远程在轨升级和加载运行时通常受制于卫通信道的资源限制、传输稳定性、速率、终端存储和处理能力等因素，因此本文将探讨一套卫通终端嵌入式软件的在线增量高效OTA方案，以适应卫星终端嵌入式设备软件的高效能升级需求的应用场景。

关键词

增量包构造算法；dicdiff算法；字典结构；子序列搜索；后缀树；在轨升级

1 引言

随着卫星通信技术不断发展，基于卫星通信系统的集成应用需求日益增加，卫星终端嵌入式软件作为实现卫星通信应用的重要软载体，为满足不断变化的应用需求，卫星终端将面临跨多区域、小步迭代、频繁变更的软件在轨升级需求。

本方案相较于卫通行业其它卫星终端升级，更优化设计了基于数据字典的增量升级技术，同时借鉴了当前主流智能汽车 OTA 关键技术，在综合效能、压缩效能、升级容错等方面，进行更体系的设计，并在实测场景中展现出更好的升级效能、容错性、稳定性等优势。

【作者简介】孙天龙（1991–），满族，中国黑龙江绥化人，硕士，工程师，从事通信软件系统设计。

2 设计方案

2.1 关键技术概述

本方案实现的关键技术主要包括：1）增量更新算，采用构造增量包和恢复全文件进行升级 2）版本控制与校验，升级文件版本控制和采用 MD5 码生成校验 3）分片传输与断点续传，通过设计传输协议，分片待传输文件，并实现断点记录和续传 4）安全传输和加密 5）多线程和异步处理，使用多线程处理文件传输、解压缩、安装等，并支持后台异步处理 6）回滚机制，升级失败时能回滚到旧版本 7）日志记录与反馈，记录升级过程中的关键事件，如：进度、安装状态、错误信息等）8）降级策略，在版本不兼容或升级性能下降时自动降级到旧版本。

本方案主要探讨基于动态数据字典的增量更新技术在卫通终端嵌入式软件 OAT 中的实现和应用。在卫通信道资源受限、升级效能要求高的背景下，将增量更新技术应用于卫星星载设备及卫星终端嵌入式软件增量升级服务中，可极大地缩小文件传输和存储的时空消耗，非常适应卫星终端远

程升级的应用场景。增量更新机制见图 1。

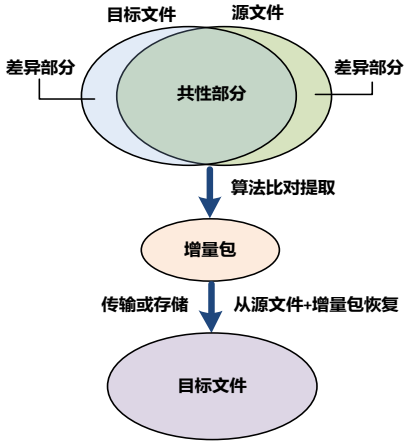


图 1 软件增量更新机制

2.2 增量包构造算法分析

增量包构造是实现增量更新的关键步骤，合成新文件是构造的逆过程，构造增量包的算法主要是使用数据结构、文件对比算法和差异编码算法，这些算法的时空复杂度及压缩率不同，但都是基于有效的方法查询文件的公共数据段（即相同部分内容），以此来构造增量包。

对于公共数据段的查找是否全面且高效，决定了算法的压缩率，也决定了新旧数据包的差异大小，以及增量包的尺寸大小。目前基于子序列搜索的公共数据段选择方法处理长字符串的应用耗时较大，效率较低；基于字典结构的公共数据段选择方法效率较高，常用算法有 dicdiff 动态数据字典算法、rdiff 算法等；基于后缀树搜索的公共数据段选择方法构造增量包的时间较长，数据压缩率较高，常用算法有 bsdiff 算法等。

2.3 系统架构和流程设计

本方案设计在本地开发部署软件在线升级服务端，提供对不同空间区域卫星终端软件在线升级服务。卫星终端作为客户端，服务端综合考虑时间效率、压缩率等因素选取基于动态字典的增量包构造算法（dicdiff 算法）进行设计，结合当前本地软件文件通过实现增量更新算法逆过程恢复构建待升级新版数据文件。

在线增量升级嵌入式软件设计采用分层、组件化、服务化逻辑架构进行设计，逻辑架构分为基础构建层、服务层、应用层三层架构来实现。其中基础构件层主要包括：服务订阅组件、线程组件、文件组件、定时器组件、序列化组件等基础服务。服务层主要包括：dicdiff 增量算法服务、数据压缩服务、文件传输服务等。应用层主要包括：升级控制、升级状态监视、信道通信状态监视等。增量升级流程见图 2。

2.4 算法和处理流程

本方案提升升级效能的关键在于提高增量包的数据压缩率。算法采用基于动态字典的 dicdiff 算法，实现高数据压缩率、高时间效率的安装文件增量包升级，主要实现查找和编码两方面的解决方案。查找方案寻求在旧版本文件中查找到尽可能多的公共数据段，减少增量包的差异信息，从而提升增量升级包的数据压缩率。基于动态字典算法，根据新旧文件数据，获取前部数据构造为新文件的后部数据的已知数据集，形成动态字典，并根据增量包构造逐步完善动态字典以支持查询能力的提升。同时，需要通过后缀数组、前缀数组来构建字典目录用于提高内容的索引查询效率。基于动态字典的增量包构造流程见图 3。

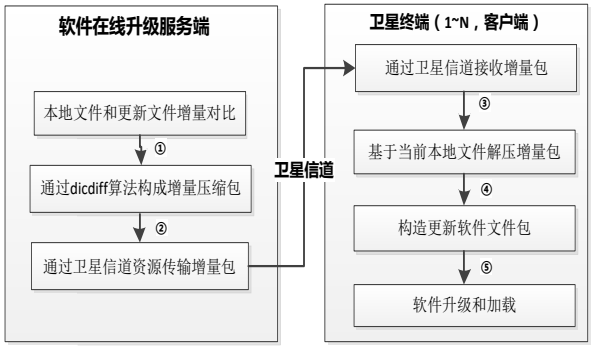


图 2 系统增量升级流程设计

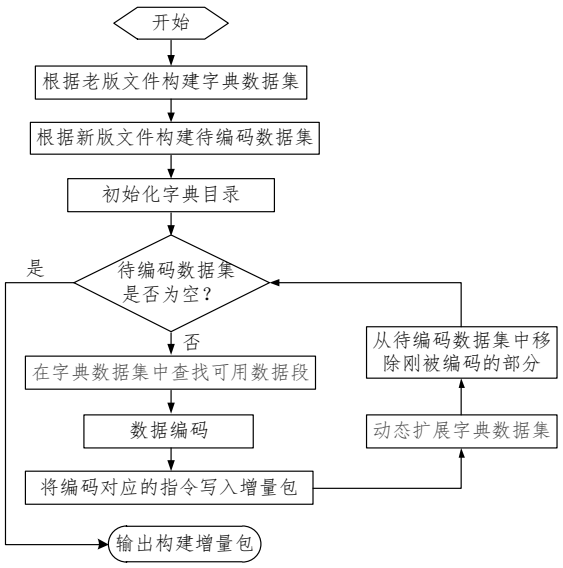


图 3 基于动态字典算法的增量包构造流程

动态字典由字典目录、动态调整的字典数据集两部分构成。字典目录由新版文件后缀数组 $N_{\text{suffixArray}}[\ ]$、新版文件名次数组 $N_{\text{rankArray}}[\ ]$、老版文件后缀数组 $O_{suffixArray}[\ ]$ 组成。字典数据集初始为老版文件，在编码过程中动态地由已被编码的数据作为已知数据补充到初始字典数据集，进行迭代完善。

该算法在已知数据集与待编码数据集之间查找最长公共前缀，用于解决基于字典目录在字典数据集中查找可用数据段。该算法需解决查询和编码问题域，其方法具体如下：

查询采用基于老版文件后缀数组的二分查询算法，在老版文件中查找可用于构造新版本文件的片段，首先确定待

编码数据头部（字符串）位置在新版文件中的位置 P，然后查找出老版文件字符串某个位置 P_{old}，使 P 位置的后缀和 P_{old} 位置的后缀包含长度最大的公共前缀，其长度为 L_1。编码采用在已编码的新版数据中查找用于构造编码数据的片段，找出新版本文件字符串某个位置 P_{new}，使新版文件字符串在 P_{new} 位置的后缀与其在 P 位置的后缀包含长度最大的公共前缀，其长度为 L_2。

P_{new} 的数值取值可按如下顺序选择：

$N_{suffixArray}\left[N_{rankArray}\left[P\right]\pm1\right]$、 $N_{suffixArray}\left[N_{rankArray}\left[P\right]\pm2\right]$、$N_{suffixArray}\left[N_{rankArray}\left[P\right]\pm3\right]\cdots\cdots$

当 L_1 大于 L_2 时，可以返回获取位置信息 P_{old}，当 L_1 小于 L_2 时，可以返回获取位置信息：P_{new} + 老版文件字符串长度。应用增量包的时空复杂度对于给定的 P 的值，查询的时间复杂度为 $O\left(\ln\left(n_{oldSize}\right)\right)$，其中 $n_{oldSize}$ 为老版文件字符串长度。编码的时间复杂度为 $O(C)$，其中 C 为小数值常数。该算法基于动态字典单次查询最长公共前缀的时间复杂度为 $O\left(\ln\left(n_{oldSize}\right)+C\right)$，整个增量包构造过程需多次执行该查询过程，若增量包构造过程中，P 的取值从 1 到 $n_{newSize}$（新版文件字符串长度），则总查询时间复杂度为 $\left(n_{newSize}\bullet O\left(\ln\left(n_{oldSize}\right)+C\right)\right)$。采用该算法，时间复杂度对于具备较强的计算和处理能力的服务端是可行的。同时，卫通终端作为客户端，构造基于增量包形成的新升级文件，其时空复杂度分别为 $O\left(n_{oldSize}\right)$、$O\left(n_{newSize}+n_{oldSize}+n_{deltaSize}\right)$，$n_{deltaSize}$ 为增量包尺寸。通过该算法查询到越多的公共数据段，增量包的尺寸将越小，卫通终端构造恢复升级文件效率越高。

3 实验验证和结语

在卫星星载设备或卫星终端的硬件仿真环境下的国产嵌入式 Reworks6.1 操作系统环境中部署升级客户端软件，在本地工控机部署服务端软件，进行升级效率仿真验证，本验证不考虑卫通信道效率的影响，仅通过实验数据验证对比增量包与整包升级的效率差异。

本方案通过增量更新机制实现软件快速升级的效果主要考虑增量升级的时间和压缩率指标，因此重点针对增量算法进行实际数据验证，选取 6 类不同尺寸和升级内容的软件升级包样本分别进行测试。通过实验结论数据整体评估来看采用增量升级效率相较传统方法优势明显，增量升级方法为整包升级耗时约 15~52%。

表 1 样本数据测试增量包效率提升对比

样本	整包升级耗时	增量 / 整包耗时占比
1 号	386S	33.5%
2 号	1339S	52.3%
3 号	671S	39.6%
4 号	922S	37.1%
5 号	525S	44.7%
6 号	851S	15.4%

参考文献

[1] 陈志龙，倪桂强，姜劲松，罗健欣，严英姿，基于动态字典的增量更新算法，解放军理工大学学报，2015

[2] 卿太杰，基于动态词典的文本压缩研究 北京邮电大学，2006

[3] 魏爽, 王晓楠，杨璟安，基于动态字典学习的欠定盲语音重构算法 . 计算机工程与设计, 2022

[4] 杨明，一种基于改进差别矩阵的核增量式更新算法. 计算机学报, 2006

[5] 刘佳新, 严书亭，贺春亮，任家东，基于序列树的增量式序列模式更新算法，计算机工程与应用，2011

[6] 陶再平，俞瑞钊，序列模式增量式更新的快速算法，计算机应用，2006

[7] 韦涌泉，董振辉，张红军，一种基于文件的嵌入式星载软件在轨升级方法，单片机与嵌入式系统应用，2018

[8] 李松涛，许良刚，林冬武，何树生，一种可软件在线升级的嵌入式系统及在线升级方法，现代电子技术，2011

[9] 邓强强，付士，杨茜，朱艳芳，一种用于嵌入式软件的在线升级方法及系统，测控技术，2020

[10] 杨光海，彭晓乐，张川，公丕华，远程升级控制系统嵌入式设备程序的设计方法，工业控制计算机，2016

[11] 贺翔，陈崇森，基于WiFi的车载终端远程软件升级方法，单片机与嵌入式系统应用，2014

[12] 张治安，基于中间件对嵌入式设备远程升级的设计与实现，吉林大学，2020.