

基于缓存结构的民机实时 QAR 数据译码方法

马 驰¹, 乔 添¹, 王 潇¹, 袁 奇², 王 婧², 张怡丰²

(1. 上海飞机客户服务有限公司, 上海 200241;

2. 华南师范大学数据科学与工程学院, 广东 汕尾 516622)

摘要: 新一代通信技术的发展使得 QAR 数据可在飞行中实时传输; 传统译码研究均针对航后 QAR 数据, 难以高效处理实时数据; 为提升实时 QAR 数据的译码效率, 提出了基于缓存结构的译码方法, 将传统的译码表转换为专门的数据结构, 并进行缓存, 避免了译码过程中对配置表的全表搜索, 提升了译码效率; 采用 Go 语言、Redis 缓存、Influxdb2 数据库, 搭建了实时 QAR 数据译码平台; 采用基于 ATG 网络的实测 QAR 数据, 对译码平台的性能进行了仿真测试; 针对实时 QAR 数据, 平均译码延迟约为 5 ms, 测试结果表明基于缓存结构的译码方法能够高效处理实时 QAR 数据。

关键词: 实时数据、QAR 译码、缓存数据结构

Decoding Method of Civil Aviation Real Time QAR Data Based on Cache Structure

MA Chi¹, QIAO Tian¹, WANG Xiao¹, XI Qi², WANG Jing², ZHANG Yifeng²

(1. Shanghai Aircraft Customer Service Co., Ltd., Shanghai 200241, China;

2. Department of Data Science and Engineering, South China Normal University, Guangdong 516622, China)

Abstract: The development of new generation communication technology enables the real-time transmission of quick access recorder (QAR) data in flight. Traditional decoding researches focus on post-flight QAR data and difficulty to efficiently process real-time data. In order to improve the decoding efficiency of real-time QAR data, a decoding method based on the cache structure is proposed. Traditional decoding table is converted into a special data structure and cached, which avoids the full table search of the configuration table during the decoding process, improving the decoding efficiency; Go language, Redis cache, and Influxdb2 database are adopted to build a real-time QAR data decoding platform; The measured QAR data transmitted by the ATG network is used to simulate and test the performance of the decoding platform. For the real-time QAR data, the average decoding delay is about 5 ms, and the test results show that the decoding method based on the cached structure can efficiently process the real-time QAR data.

Keywords: real time data; QAR decoding; cached data structure

0 引言

目前的民机航线数据分析, 航中阶段数据主要为 ACARS 报文^[1], 虽然具有实时性, 但带宽低、费用高、可供分析数据有限; 航后阶段数据主要为 QAR (quick access recorder) 数据, QAR 数据采集参数丰富、采样频率高, 能较好满足数据分析的需求。航后 QAR 数据一般待飞机落地、发动机关车后, 通过地面蜂窝基站或 Wifi 热点将 QAR 数据下传至航空公司以及飞机主制造商的服务器上^[2]。航后数据分析延迟较大, 难以用于部署实时性要求较高的分析模型。受限于通信链路带宽限制, 基于飞行过程中实时数据的分析尚未普及。

卫星通信的发展和 ATG (air to ground) 通信技术的日臻完善将突破民机飞行中数据传输的瓶颈^[3], 使 QAR 数据实时传输成为可能。目前海事卫星提供基于 I4 卫星网络的

SBB 航空数据服务, 每路通信速率最高可达 432 kbps, 且每比特通信成本日益降低^[4]; ATG 技术通过地面专用基站和空中机载设备通信, 将 4G、5G 信号覆盖至空中, 实现飞机与地面之间的实时无线宽带互联^[5]。

目前, 民机航线数据分析主要使用航后 QAR 数据, 即待飞机落地、发动机关车后, 通过地面蜂窝基站或 Wifi 热点将 QAR 数据下传至航空公司和飞机主制造商的服务器上, 对实时 QAR 数据的相关研究较为有限。QAR 数据的实时传输对航线译码技术带来新挑战与新机遇。目前流行的商用 QAR 译码软件均针对航后数据, 如 AGS、Airfase 等^[6], 采用逐个航班译码的方式。每个航班数据为一个 ARINC 717 格式的 dat 文件, 该文件包含一次飞行中的全部参数记录。与航后数据不同, 实时 QAR 数据采用流式传输^[7], 飞机上每秒采集的传感器记录数据拆被分成若干个 IP 数据包^[8], 数据包生成后立即经卫星或 ATG 等宽带网络

收稿日期: 2022-07-18; 修回日期: 2022-01-09。

作者简介: 马 驰(1978-), 男, 山东平度人, 高级工程师, 主要从事航线数据管理、分析及数字化产品研发等方向的研究。

通讯作者: 袁 奇(1987-), 男, 吉林吉林人, 博士, 工程师, 主要从事工业互联网与工业大数据方向的研究。

引用格式: 马 驰, 乔 添, 王 潇, 等. 基于缓存结构的民机实时 QAR 数据译码方法[J]. 计算机测量与控制, 2023, 31(3): 255-261.

发送至地面。地面接收到 XML (extensible markup language) 格式的流式数据包, 译码软件需实现实时接收、实时译码。按照 QAR 数据记录规则, 将 XML 数据转换为各传感器参数的工程值, 并转存至数据库供后续分析使用。实时 QAR 译码是民航数据分析从航后、事后分析走向全航段、全过程分析的必不可少的环节, 相关技术研究和程序开发具有重要意义和价值。

文献 [9-10] 提出了基于 ATG 网络的实时数据的应用方案, 包括机舱音视频实时监控、客舱无线网络、QAR 数据实时译码与监控等模块, 但其中 QAR 译码并没有针对实时数据的特性进行专门优化。本文提出基于缓存数据结构的实时 QAR 译码方法, 将传统的译码表转换为专门的数据结构进行缓存, 避免译码时对配置表的全表搜索, 加快译码速度。本文采用 Go 语言^[11]、Redis 缓存^[12]、Influxdb2 数据库^[13]和监控系统^[14]搭建了实时 QAR 数据译码平台, 对 ATG 网络的实时 QAR 测试数据展开模拟测试。测试结果表明, 所提方法能够高效处理实时 QAR 数据。

1 QAR 数据格式及传输方式

1.1 QAR 数据记录规范

QAR 数据, 即快速存取记录器数据, 是目前民航业用于航后飞行品质监控、燃油效率分析、维修分析等工作的主要数据依据。QAR 数据的记录规范由美国航空无线电协会 ARINC (aeronautical radio inc) 制定^[15]。机载传感器数据通过飞机总线传输, 如 ARINC 429 总线^[16]或 AFDX 总线^[17]。QAR 记录设备从总线上获取数据, 再根据记录配置表将采集到的数据以二进制数据流的形式存储指到记录文件中。

以目前常用的 ARINC 717 协议为例, 记录过程以 4 秒为一个周期, 即 4 秒为一个帧, 每个帧又分为 4 个子帧, 每个子帧对应 1 秒钟内的数据记录。子帧内可包含 256/512/1024 个字槽 (Word), 每个字槽占 12 比特 (bits)。字槽可记录一个或多个机载传感器数据, 数据帧内各字槽记录的数据情况通过译码表进行配置。

1.2 QAR 数据传输方式

受通信链路限制, 目前商用航班的 QAR 数据以航后传输为主, 即待飞机落地后, 通过地面蜂窝或 Wifi 网络传输数据。卫星通信和 ATG 通信的发展, 为 QAR 数据在飞行中的实时传输提供了支撑。以 4G-ATG 网络实际测试为例, 各基站的上行 (空对地) 数据峰值传输速率可达 11.74 Mbps, 平均速率为 6.37 Mbps。各 ATG 基站的下行 (地对空) 数据峰值传输速率为 16.46 Mbps; 平均速率为 8.65 Mbps。对于每秒 1 024 Words 的实时 QAR 数据, ATG 网络完全满足 QAR 数据实时传输需求。

ATG 技术是一种新型空地数据传输解决方案, 通过在航路沿线架设地对空通信基站, 建立上下行数据传输链路。具有高带宽、低成本、低时延等特点, 能够满足飞机航行过程中空地海量数据通信的需求。目前基于 ATG 技术的空地数据链路传输在民航领域应用已比较成熟。美国 GoGo 公司所部署的 ATG 网络已对美国全境和加拿大大部分区域实现

了立体覆盖, 占据了美国各大主流航空公司飞机上网超过 50% 的市场份额。欧洲 Inmarsat 公司联合德国电信、泰勒斯、空客等构建 EAN (欧洲 4G 版的 ATG) 网络, 已经完成 300 个基站的部署, 并投入商用。随着国内 5G 网络的成熟, 5G ATG 将进一步加大差异化竞争优势, 将会成为民航产业智慧飞行、智慧运营的重要基础保障。2021 年 4 月, 民航局正式发布《中国民航新一代航空宽带通信技术路线图》, 首次明确提出以 5G 为代表的新一代航空宽带通信技术应用于中国民航的系统性实施路径。

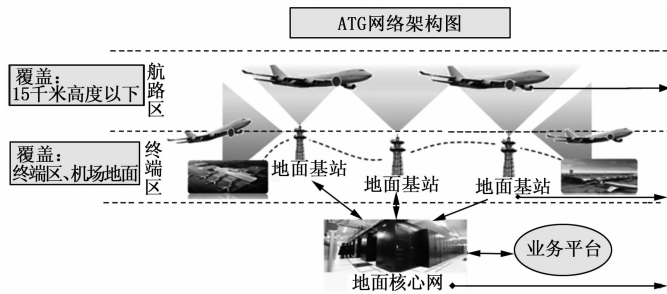


图 1 基于 ATG 网络的实时 QAR 传输

2 基于缓存的实时数据译码

对实时 QAR 数据进行解析, 是航线数据译码工作的新挑战。数据译码是指将原始二进制的 QAR 记录还原为飞行真实数据的过程, 还原后的数据也被称为工程值数据, 工程值数据可分为两大类, 一类是模拟量参数, 例如发动机转速、空调出口温度、液压系统压力等关键部件状态, 又如飞行高度、空速、滚转角、偏航角、俯仰角度等速度姿态信息, 模拟量参数在译码配置时需要记录精度信息; 另一类是开关量参数, 例如发动机启动活门开关状态、液压系统泵的开关状态、发动机是否出现火警等, 开关量参数一般只反映两种状态, 在二进制记录中用 1 个 bit 表示。模拟量、开关量等参数分别记录在不同的 Word 和子帧中, 每个子帧中的不同 Word 记录哪些参数, 可以通过 QAR 记录设备进行客户化配置, 地面系统再根据不同架次的配置信息将二进制记录还原为工程值。

传统的航后译码是在航班落地后, 接收到本次飞行完整的 QAR 记录, 再将整段记录转换为工程值。译码时按照译码配置表, 逐个参数地解析二进制记录。由于实时传输情况下, 每次传输的数据只包含一秒钟内容部分 Word 信息, 且这些 Word 并不是按照序号依次排列的, 因此传统的航后译码软件并不能直接解析实时 QAR 数据, 需要开发专门的译码工具。

为提升实时 QAR 数据译码效率, 本文提出基于缓存结构的译码方法。传统航后 QAR 译码方法将译码配置以表格形式存储, 例如采用 MySQL 等关系型数据库中的表结构进行存储。而对于实时数据, 需要进行逐个 Word 的译码, 在接收到 Word 信息后, 译码程序对配置表进行全表搜索, 匹配到对应的译码信息。以每秒钟收到 275 个 Word 的配置情况为例, 传统译码方法需要进行 275 次关系型数据库的全

表搜索, 效率相对较低。本文提出基于 Redis (remote dictionary server) 缓存的实时 QAR 数据译码方法。所提方法将传统译码表转换为索引-参数记录结构, 该结构由索引区和记录区构成, 在收到实时 QAR 数据后, 根据 Word 序号和子帧序号, 读取对应索引位置上的偏移量, 再根据偏移量跳转至记录区, 直接读取该 Word 的译码信息, 避免了配置表全表搜索, 降低了译码复杂度。本文所提译码配置结构由 Redis 缓存存储, 避免了硬盘读取延迟, 可实现配置结构的快速读入。

2.1 实时 QAR 报文格式

实时 QAR 数据由机载 WQAR 设备生成, 并由机载传输设备封装为 IP 数据包, 再通过 ATG 网络或卫星网络传输至地面系统。实时数据的存储形式为 XML 格式^[18], 如图 2 所示。每秒钟的 ARINC 717 子帧记录被划分为多条 XML 记录, 每一条 XML 记录包含多个 Words 信息。以本文中使用的实验数据为例, 每秒的子帧记录分为 16 条 XML 文档, 每条 XML 文档中包含 16 至 17 个 Word 记录, 每个子帧内共 275 个 Word 记录。

```
<method name="publishAvionicParameters">
  <refreshperiod>1000</refreshperiod>
  <context>
    <key name="SEQUENCE_ID" value="1"/>
    <key name="TOTAL_MSG_PARTS" value="16"/>
  </context>
  <parameters>
    <parameter name="FrameCounter" type="0" validity="1" value="1583" time="1609219457331"/>
    <parameter name="WORD1" type="0" validity="1" value="1464"/>
    <parameter name="WORD10" type="0" validity="1" value="28"/>
    <parameter name="WORD100" type="0" validity="1" value="170"/>
    <parameter name="WORD101" type="0" validity="1" value="0"/>
    <parameter name="WORD102" type="0" validity="1" value="0"/>
    <parameter name="WORD103" type="0" validity="1" value="637"/>
    <parameter name="WORD104" type="0" validity="1" value="10"/>
    <parameter name="WORD105" type="0" validity="1" value="68"/>
    <parameter name="WORD106" type="0" validity="1" value="767"/>
    <parameter name="WORD107" type="0" validity="1" value="255"/>
    <parameter name="WORD108" type="0" validity="1" value="390"/>
    <parameter name="WORD109" type="0" validity="1" value="1680"/>
    <parameter name="WORD11" type="0" validity="1" value="4056"/>
    <parameter name="WORD110" type="0" validity="1" value="768"/>
    <parameter name="WORD111" type="0" validity="1" value="0"/>
    <parameter name="WORD112" type="0" validity="1" value="0"/>
    <parameter name="WORD113" type="0" validity="1" value="64"/>
  </parameters>
</method>
```

图 2 . XML 格式的实时 QAR 数据示例

XML 记录通过树形结构存储数据, WQAR 设备生成的 XML 记录格式如图 2 所示。每条 XML 记录对应的树形结构的根节点名为 publishAvionicParameters, 子节点 refreshperiod 记录数据更新频率, 为一秒钟一次; context 节点记录具体的信息, 其 TOTAL_MSG_PARTS 子节点记录本报文中宗的 XML 记录个数, SEQUENCE_ID 子节点记录本条 XML 对应的序号; 机载传感器参数信息包含在 <parameters> 元素中。<parameters> 元素包含多个 <parameter> 子元素, 每个 <parameter> 元素存储一个 Word 记录。<parameter> 元素包含 value、validity 等多个属性, 这些属性信息标识了条数据的 Word 序号、有效性等信息, 各属性的含义如表 1 所示。

表 1 实时 QAR 数据 <parameter> 元素属性及作用属性作用

属性	作用
name	标记 Word 序号, 如 name = WORD100
value	该 Word 记录值, 十进制表示
validity	记录的有效性
time	时间戳, 该条记录生成时间

2.2 译码缓存结构

为提升实时 QAR 数据的译码效率, 本文提出了基于缓存数据结构的译码方法。所提译码配置结构由索引区和参数记录区构成, 如图 3 所示。

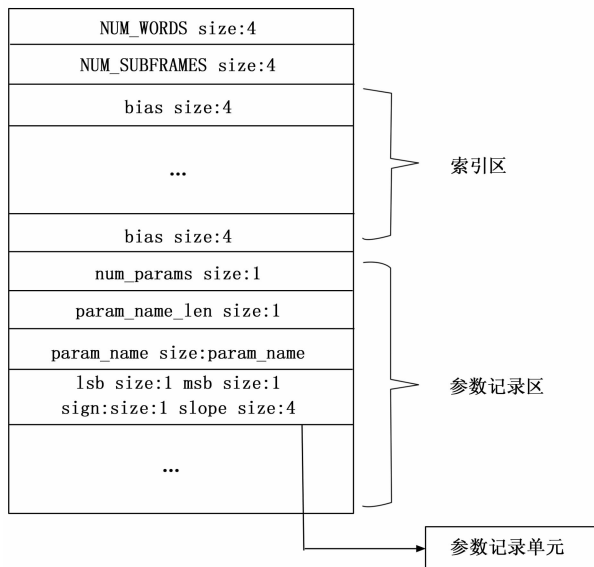


图 3 基于缓存的译码配置结构

不同 Word 的译码配置信息存储在参数记录区。参数记录区由参数记录单元构成。每个单元记录单个参数的译码配置信息, 包括参数名称长度 (占用字节数)、参数名称、起始位 (lsb)、终止位 (msb)、正负情况 (sign)、精度 (slope)。在一个子帧内, 单个 Word 可存储多个机载传感器参数, 因此其译码配置信息中可包含多个参数记录单元, 如图 3 所示。在参数记录区, 参数记录单元连续存储, 同一子帧同一 Word 对应的参数记录单元存储在一起, 对于单个参数记录单元, 所存储内容依次如下:

- 1) 参数名称长度, 占用 1 字节;
- 2) 参数名称, 按其长度以字符型存储;
- 3) lsb, 起始位, 占用 1 字节
- 4) msb、终止位, 占用 1 字节
- 5) sign, 正负情况, 占用 1 字节
- 6) slope, 精度, 解码时将二进制记录转为十进制后再乘以精度, 为传感器记录值, 采用浮点型存储, 占用 4 字节。

一个 Word-子帧中可包含多个参数记录单元。如表 2 中, Word 283, 子帧 1 共记录了 3 个参数, 分别为 WINDSHST、WSALRTLTV 和 ALERTNO, 表示风切变状态、风切变告警级别和风切变告警代码^[20]。

表 2 Word-子帧对应的译码配置信息示例

Word	子帧	参数名称	lsb	msb	sign	slope
283	1	WINDSHST	1	3	0	1
283	1	WSALRTLTV	4	5	0	1
283	1	ALERTNO	6	8	0	1

参数记录单元连续地存储在参数记录区中。已知记录

数据所在子帧和 Word 序号, 可通过索引区查找其对应的译码配置信息在记录区中的位置。索引区总大小为 $NUM_WORDS \times NUM_SUBFRAMES \times 4$ 字节, 其中 NUM_WORDS 为 Word 总数, $NUM_SUBFRAMES$ 为子帧总数, 对于当前主流民航机型, $NUM_WORDS = 1024$, $NUM_SUBFRAMES = 4$ 。索引区为每个给定的 Word 和子帧分配 4 字节存储空间, 用于记录所对应的译码信息在记录区的偏移量。根据 Word 和子帧序号由小到大的次序, 偏移量在索引区依次排列。译码时, 根据 Word 序号和子帧序号, 从索引区读取偏移量, 再跳转到记录区对应位置。由于给定子帧和 Word 可对应多个参数记录单元, 该位置前 4 个字节记录该子帧和 Word 记录的参数个数。

以上译码数据结构通过 Redis 存储。Redis, 即远程字典服务, 是开源的缓存键-值型数据库, 采用 ANSI C 语言编写, 具有存取效率高、支持网络等优势, 其数据存储可基于内存, 亦可提供持久化保存, 并提供多种语言的 API 接口^[12]。

2.3 基于缓存结构的实时译码过程

由于各架次飞机采用的译码配置不同, 实时 QAR 以 UDP 数据流^[19]的形式发送至地面。当地面服务器接收到实时 QAR 数据报文后, 首先获取架次信息, 再根据该架飞机的译码配置从 Redis 中获取译码缓存数据结构, 然后解析数据包中各 Word 的记录值并在缓存结构中匹配到译码信息, 将二进制记录值转换为参数工程值。

整体译码流程如图 4 所示, 具体译码过程如下:

1) 通过数获取模块接收实时 QAR 数据报文。机载 WQAR 设备将从飞机 ARINC 总线上采集到的机载传感器数据转换为 XML 记录并以 IP 数据文的形式发送给机载传输设备, 如 ATG 设备, 传输设备对 XML 数据进行整理, 将一秒钟内的 XML 记录整合成一个 UDP (user datagram protocol) 格式的数据包; 传输设备对数据包进行 RSA 加密、CRC 校验之后通过 ATG 网络或卫星网络发给网络服务商的地面数据中心; 网络服务商对机载传输数据进行解密和校验, 并将解密后的数据发送给数据使用方, 如航空公司或飞机主制造商, 传输过程也采用 UDP 模式。实时数据的译码工作由数据使用方完成, 图 4 中的 UDP Server 模块与网络服务商建立传输连接, 接收数据报文, 并提取报文类型、MSN 号、发送时间等信息, 再转发至数据解析模块。

2) 实时数据报文内容采用 UTF-8 编码, 具体格式为:
类型 & 机号 & 时间 & XML 数据体

其中“&”为数据分隔符; “类型”为 R 或 H, R 表示实时数据, H 表示历史数据; “机号”为飞机尾号; “时间”为数据发送时间, 时间格式“yyyyMMddHHmmssSSS”(年月日时分秒毫秒); “XML 数据体”: 机载 QAR 数据内容, 采用 XML 方式组织数据, 具体介绍见上文中 2.1 小节。数据包格式示例如下:

R&B6556&20170613142130123&<XML 数据体>

3) 根据报文中的飞机尾号信息, 如 B6556, 获取对应架次的译码版本。

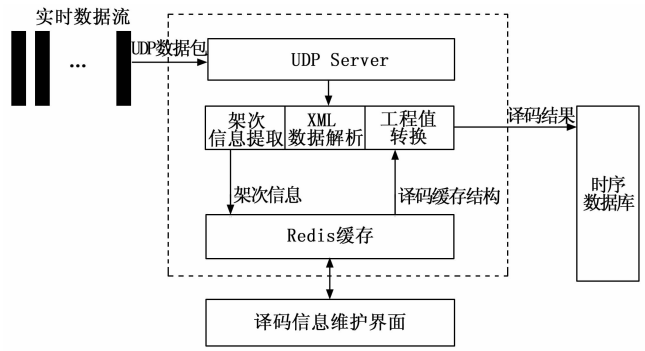


图 4 译码流程示意

4) 根据译码版本, 从 Redis 缓存中获取相应的译码数据结构;

5) 根据 XML 记录起始标记, 对数据报文中的 XML 数据体进行分割, 依次获取 XML 记录;

6) 对于各条 XML 记录, 根据 <parameter> 中的 <name> 属性和 <value> 属性, 提取其中的 Word 序号及其记录值;

7) 序号为 1 的 Word 为同步字。QAR 记录中采用巴克码^[8]标识子帧序号。巴克码是一种具有特殊规律的二进制码组, 其自相关函数的主瓣峰值是旁瓣峰值的 13 倍, 因此常用于帧同步。根据 Word1 的记录值可得到当前数据包对应的子帧序号, 如表 3 所示;

表 3 Word1 记录值所代表的子帧序号

Word1 记录值(十进制)	子帧序号
1 464	1
2 631	2
3 512	3
583	4

8) 根据 Word-子帧序号, 从译码缓存结构中读取参数个数及各参数的译码信息;

9) 根据各参数的起始位和终止位, 获取记录中相应的 bit 信息, 再根据正负情况转换为十进制, 乘以精度后得到各参数记录的工程值;

10) 将译码结果结果写入实时数据库。

2.4 跨子帧参数处理

对于部分精度较高的参数, 当其所需的记录 bit 数超过 12 位时, 采用多个 Word 或同一 Word 的多个子帧进行记录。参数记录跨多个子帧时, 由于各子帧信息分布在不同的数据报文上, 难以在单个报文的译码中直接计算, 本文在采用时序数据库中的聚合功能进行处理。

以某机型经度和纬度参数配置为例, 经度参数由 Word 228 第一子帧和第二子帧记录, 纬度参数由 Word 228 第三子帧和第四子帧记录。译码时, 分别解析各子帧上的工程值, 记为 LATH、LATL、LONGH 和 LONGL, 并写入时序数据库。在使用经度和纬度信息时, 对数据库中的记录进行聚合, 聚合代码如图 5 所示。首先提取数据库中的

LONGH、LONGL、LATH、LATL 字段, 通过时序数据库提供的 aragateWindow 语句^[13], 进行窗口聚合操作, 窗口大小为 8 秒钟, 聚合后计算各字段平均值, 再分别将 LONGH 与 LONGL, LATH 与 LATL 的平均值相加, 得到经度和纬度计算结果。

```

from(bucket: "atg")
  |> range(start: -2h)
  |> filter(fn: (r) => r["_measurement"] == "conf1")
  |> filter(fn: (r) => r["_field"] == "LONGH" or r["_field"] == "LONGL" or r["_field"] == "LATH" or r["_field"] == "LATL")
  |> aggregateWindow(every: 8s, fn: mean, createEmpty: false)
  |> pivot(rowKey: ["_time"], columnKey: ["_field"], valueColumn: "_value")
  |> map(fn: (r) => ({ r with lat: r.LATH + r.LATL, lon: r.LONGH + r.LONGL }))
  |> filter(fn: (r) => exists r.lat and exists r.lon )
  |> sort(columns: ["_time"], desc: false)

```

图 5 跨子帧参数的聚合计算示例

本文采用 InfluxDB 时序数据库存储译码后的工程值结果。InfluxDB 是由 InfluxData 开发的开源时序型数据库, 专注于海量时序数据的高性能读、高性能写、高效存储与实时分析, 数据库中每条记录均带有时间戳。InfluxDB 使用场景广泛, 在目前主流时序数据库中装机量排名前列, 其典型应用包括智慧物联网监控分析, 如石油化工、采矿、智能制造等领域内的设备数据采集、分析、告警, 以及金融、可穿戴医疗设备、车联网、智慧城市等多种行业。同时, InfluxDB 还广泛用于日志数据存储与分析、大数据平台软件及系统监控等业务领域。实时 QAR 每条数据报文均带有时间戳, 符合时序数据库的存储特性, 本文将 InfluxDB 应用于航空实时数据存储, 以提升数据的存储和读取效率。InfluxDB 中的数据操作可采用 InfluxQL 语句。InfluxQL 是用于在 InfluxDB 中进行数据增加、查询、探索的类似于 SQL 的查询语法。在 InfluxDB 中, 每条时序数据被称为一个数据点 (point), 一系列存储点构成的数据集形成一个存储桶。图 5 中的示例代码采用 FluxQL 语句编写, 语句 from (bucket: " atg") 表示对名为 atg 的存储桶 (bucket) 进行数据操作。针对每个存储桶可制定一个保留策略, 即每个数据点保留的持续时间, 从而减少失去时效性的历史数据对存储空间的挤占。图 5 中 range (start: -2h) 表示选取近两个小时内的数据, 确定时间段后通过 filter 语句筛选经纬度字段。由于经度参数和纬度参数记录均跨越了两个子帧。因此采用 aggregateWindow 语句进行数据聚合, 得到 8 秒钟窗口内的数据记录。由于 8 秒钟的时间窗口可覆盖一个完整的帧, 因此可进行跨子帧数据计算, 得到经纬度记录的工程值。

3 译码平台搭建及实验结果

本文采用 Python 将传统译码配置表转换为缓存译码结构并在 Redis 中存储; 采用 Go 语言开发译码程序, 译码后的工程值采用 Influxdb2 时序数据库存储, 并搭建了数据监控界面。

为验证本文所提实时译码方案的性能, 采用 ATG 测试飞行中的实测数据作为输入, 开展模拟译码实验。实验中

Redis、Influxdb2、前端展示平台和译码程序运行在同一台计算上, CPU 型号为 11th Gen Intel (R) Core (TM) i5-1135G7 @ 2.40 GHz, RAM 大小 16.0 GB。实验结果显示, 针对每秒钟的实时 QAR 数据, 平均译码时间约为 5 ms, 表明本文所提方法能够高效译码实时 QAR 数据。

图 6 根据译码参数生成的飞行轨迹监控图。飞行轨迹根据不同时刻的经纬度数据绘制。经纬度的计算方法在 2.4 小节中给出。



图 6 基于实时 QAR 译码结果的飞行轨迹图

实时 QAR 数据可用于飞机关键部件的状态监控, 与基于 ACARS 报文的实时监控相比, 可利用数据量大幅增长。例如, 对于发动机振动, 传统报文监控在一次航班中只能发送起飞、巡航的若干固定时刻的数据, 或当振动值超出一定门限值时发送告警报文。报文方式数据量的限制分析能力和实时性。

图 7 为发动机低压转子 (N1) 振动和转速监控图, 可用于发动机性能的实时监控。在飞行中, 发动机振动监控仪接收 4 个加速度传感信号 (每台发动机 2 个加速度传感器) 传来的振动信号和 4 个发动机转速信号, 通过窄幅范围内的振动计算得到发动机的不平衡状态, 进而计算出振动值。本文所提方法可用于在飞行过程中监测发动机振动状态, 导致发动机振动原因可归为两类: 一类属于与转子转速有关的规律性振动, 如转子不平衡、以及由转子带动的传动齿轮等其他结构产生的激振力; 另一类属于与转速无关的非规律性激振力, 如静转子磨损、压气机喘振、振荡燃烧等。通过监控振动异常值及趋势变化, 结合用户自定义的预警值提前执行维修工作, 将维修关口前移, 避免振动异常可能导致的空停风险, 降低对发动机叶片、转静子叶片的损耗, 延长发动机在翼时间, 提升可靠新, 降低维修成本, 避免 AOG, 提升旅客舒适度。

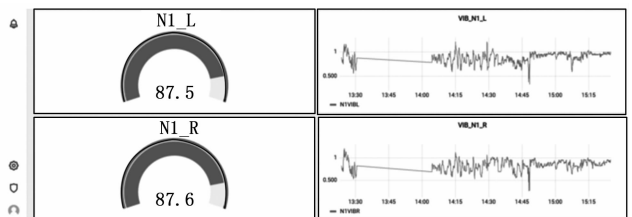


图 7 基于实时 QAR 译码结果的发动机转速振动监控

图 8 为发动机滑油压力实时监控示例, 以监控发动机滑油管路堵塞等故障。润滑系统是航空发动机重要子系统,

其主要功能包括润滑、冷却、清洁等 3 个方面。发动机滑油系统设计的目的是保护发动机并将发动机损伤降到最小程度。发动机异常运行时，多个发动机信号监控发动机滑油系统的性能，包括滑油位、滑油压力和滑油温度。滑油压力是航空发动机的重要运行参数，滑油压力过高或过低都会对发动机运行造成影响，尤其是当滑油压力低于门限值时需要飞行机组关停发动机，以避免发动机在润滑不良状态下运行而造成损伤。油压力传感器的作用将相应的直流输出信号（等于轴承的供油压力）提供给飞机。滑油压力传感器一般测量滑油泵出口和齿轮箱之间的压力差。并将压力差转换成电信号发送到数据总线上。对滑油压力参数的实时监控能够使地面人员及时识别飞机状态，当状态出现异常时协助飞行人员提早采取措施，提升飞行安全性，并提前在地面做好维护准备，以缩短飞机离场时间，提升维护效率。

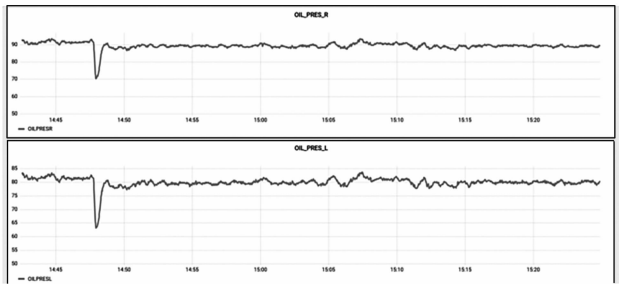


图 8 基于实时 QAR 译码结果的发动机滑油压力监控

除上述飞机关键部件参数监控应用外，基于本文所提实时数据译码方法，还可以开展空地宽带环境下的实时数据分析服务，解决目前实时监控带宽小（ACARS-5 kHz）、费用贵（800/M）的问题。结合机载端 ATG 设备取证工作和卫星融合通讯的推进，可面向航司提供含 ATG/海事/KaKu 卫星等综合地面实时监控和分析能力。特别是针对空中突发的瞬态异常，如滑油急性泄漏、发动机喘振等问题。

4 结束语

本文针对实时 QAR 数据的特性，设计了基于缓存的译码配置数据结构，基于缓存数据结构设计了实时 QAR 译码算法。基于 Go 语言，实现了所提译码算法，译码过程中，根据 Word 和子帧序号能够直接匹配到相应的译码信息，避免了译码表全表搜索，提升了译码效率。根据真实 ATG 试飞实验的数据，本文搭建了地面模拟系统，测试结果显示，所提方法能够高效译码实时 QAR 数据。通过航中实时数据与地面航后数据的融合分析，可对关键参数实现全航段、全周期的连续监控，进一步扩展现有故障识别的范围和精度。实时 QAR 数据分析是民航领域新兴的技术，也是未来的发展趋势，目前相关研究较少。在本文的研究基础上，还可以进一步开展以下方面的研究：

实时 QAR 数据清洗技术：能够处理不同记录频率（1/4 Hz、1/2 Hz、8 Hz 等）的参数，并对译码结果进行数据校准，识别异常值并处理，对异常值或空值通过算法进行插补；

多架次实时数据并行处理技术：探索基于 GPU 的加速算法，支持航线多架次、高通量数据的并发 QAR 数据的实时接收处理，采用多进程等技术并行处理各架机的实时数据，同时对数据丢包、飞行参数不连续、数据安全性等问题逐一研究。

实时数据监测与分析技术：通过机理+数据混合模型，计算能够反映飞机关键部件性能的监控指标，并对超出警戒值的指标告警。通过建立算法、模型对监控的实时数据进行评估。

实时 QAR 数据动态缓存技术：探索实时数据高效缓存方法，优化实时数据存储结构，能够同步存储译码后的实时数据；提供高效读写接口，能够支持基于实时数据开展实时状态监控、飞行品质分析、燃油效能分析等工作；设计数据动态释放机制，根据数据时效性、系统内存容量、读写耗时等因素，优化存储性能^[21]。

参考文献：

- [1] ROY A. Secure aircraft communications addressing and reporting system (ACARS) [J/OL]. 20th DASC. 20th Digital Avionics Systems Conference, https://xueshu.baidu.com/user-center/paper/show?paperid=73e9afa0324f8571486880d84be96f4b&site=xueshu_se.
- [2] WANG L, WU C, SUN R. An analysis of flight Quick Access Recorder (QAR) data and its applications in preventing landing incidents [J]. Reliability Engineering & System Safety, 2014, 127: 86-96.
- [3] CHEN X, LIN T, KHAN M. Afshang and M. Mozaffari., 5G Air-to-Ground Network Design and Optimization; A Deep Learning Approach [J/OL]. 2021 IEEE 93rd Vehicular Technology Conference (VTC2021-Spring), <https://www.xueshu.com/publication/3104541194>.
- [4] HICEV S D. Introduction to inmarsat broadband global area network for mobile backbone networks [J]. Bulletin of Electrical Engineering and Informatics, 2020, 9 (2): 843-852.
- [5] MIAO W, LUO C, MIN G, et al. Unlocking the Potential of 5G and Beyond Networks to Support Massive Access of Ground and Air Devices [J]. IEEE Transactions on Network Science and Engineering, 2021, 8 (4): 2825-2836.
- [6] HAN J, JIAO Y. Conception of QAR Data Application Mode Based on Virtual Integration [C] //2018 IEEE International Conference of Safety Produce Informatization (IICSPI). IEEE, 2018: 392-395.
- [7] KARIMOV J, RABL T, KATSIFODIMOS A, et al. Benchmarking distributed stream data processing systems [C] //2018 IEEE 34th International Conference on Data Engineering (ICDE). IEEE, 2018: 1507-1518.
- [8] SACCONI G, MATYAS M, MEAD R. Transition to IP-based links for aviation [C] //2018 Integrated Communications, Navigation, Surveillance Conference (ICNS). IEEE, 2018: 1-9.
- [9] 孔祥骏, 秦海涛, 王淑敏, 等. 一种基于空地宽带通信的应用

系统及方法 [P]. 中国: 110278023, 2019-09-24.

- [10] QIN H, KONG X, SHU P. Real-time downloading and analysis of QAR data using Air-to-Ground wireless communications [C] // 2019 IEEE 1st International Conference on Civil Aviation Safety and Information Technology (ICCASIT), 2019, 519 - 524.
- [11] DONOVAN A A A, KERNIGHAN B W. The Go programming language [M]. Addison-Wesley Professional, 2015.
- [12] CARLSON J. Redis in action [M]. Simon and Schuster, 2013.
- [13] LAST M, KLEIN Y, KANDEL A. Knowledge discovery in time series databases [J]. IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics), 2001, 31 (1): 160 - 169.
- [14] CHAKRABORTY M, KUNDAN A P. Grafana [M] Monitoring Cloud-Native Applications. Apress, Berkeley, CA, 2021: 187 - 240.
- [15] Aeronautical Radio. INC, Air/Ground character-oriented protocol specification ARINC specification [S]. Annapolis: ARINC, 1998.
- [16] ARINC characteristic 717-12, flight data acquisition and recording system [S]. US: Airlines Electronic Engineering Committee, 2006.
- [17] FUCHS C M, Schneelee A S. The evolution of avionics networks from ARINC 429 to AFDX [J]. Innovative Internet Technologies and Mobile Communications (IITM), and Aerospace Networks (AN), 2012, 65: 1551 - 3203.
- [18] BRAY T, PAOLI J, SPERBERG-MCQUEEN C M, et al. Extensible markup language (XML) 1.0 (Third Edition) [J/OL]. World Wide Web Consortium (W3C), 2004. <http://www.w3.org/TR/REC-xml>.
- [19] LARZON L A, DEGERMARK M, PINK S. UDP lite for real time multimedia applications [M]. Hewlett-Packard Laboratories, 1999.
- [20] 崔 锋. 航空器运行典型大气环境建模与仿真 [D]. 中国民航大学, 2016.
- [21] 马 驰. 民机运行支持数据湖设计与实现 [J]. 计算机测量与控制, 2021, 29 (7): 175 - 179.
- [12] KLEINROCK L, TOBAGI F A. Packet switching in radio channels. I: Carrier sense multiple-access modes and their throughput-delay characteristics [J]. IEEE Transactions on Communications, 1976, 23 (12): 1400 - 1416.
- [13] 深圳市锐能微科技有限公司. 单相计量芯片 RN8209C、RN8209D 用户手册-v1-5 [Z]. 2015.
- [14] Semtech Corporation. Datasheet SX1276/77/78/79 [Z]. 2016.
- [15] 陈炳斌. 基于 LoRa 的物联网通信系统设计与实现 [D]. 天津: 天津大学, 2020.
- [16] KUO F F. The ALOHA system [J]. ACM SIGCOMM Computer Communication Review, 1974, 4 (1): 5 - 8.
- [17] ZIOUVA E, ANTONAKOPOULOS T. CSMA/CA performance under high traffic conditions: throughput and delay analysis [J]. Computer communications, 2002, 25 (3): 313 - 321.
- [18] 范 奕. 基于 CSMA/CA 改进的 Ad Hoc 网 MAC 协议设计 [D]. 成都: 电子科技大学, 2007.
- [19] 金伟妹. 用于工业物联网环境的 LoRa 多点通信协议研究 [D]. 浙江工商大学, 2019.
- [20] YANG Y, YUM T S P. Delay distributions of slotted ALOHA and CSMA [J]. IEEE Transactions on Communications, 2003, 51 (11): 1846 - 1857.
- [21] SHAHIN N, ALI R, KIM S W, et al. Cognitive backoff mechanism for IEEE802.11ax high-efficiency WLANs [J]. Journal of Communications and Networks, 2019, 21 (2): 158 - 167.
- [22] LI Y, ZHOU G, ZHENG N, et al. An adaptive backoff algorithm for multi-channel CSMA in wireless sensor networks [J]. Neural Computing and Applications, 2014, 25 (7): 1845 - 1851.
- [23] PANG Q, LIEW S C, LEE J Y B, et al. Performance evaluation of an adaptive backoff scheme for WLAN [J]. Wireless Communications and Mobile Computing, 2004, 4 (8): 867 - 879.
- (上接第 240 页)
- 联网通信方式相比, 该系统具有较远的通信距离、较低的开发运行管理成本和较高的灵活性等优点。本方案的设计符合万物互联的发展趋势, 能较好的满足经济实惠的远距离用电信息监控系统的需求。依托 LoRa 无线通信技术, 笔者后续将对组网算法做进一步研究, 以进一步提高无线传感器网络的性能, 增强其实用性。

参考文献:

- [1] 蒲泓全, 贾军营, 张小娇, 等. ZigBee 网络技术研究综述 [J]. 计算机系统应用, 2013, 22 (9): 6 - 11.
- [2] 杨观止, 陈鹏飞, 崔新凯, 等. NB-IoT 综述及性能测试 [J]. 计算机工程, 2020, 46 (1): 1 - 14.
- [3] VANGELISTA L. Frequency shift chirp modulation: The LoRa modulation [J]. IEEE Signal Processing Letters, 2017, 24 (12): 1818 - 1821.
- [4] 赵永志. 基于 BLE-Mesh 与 NB-IoT 技术的远程抄表系统设计 [J]. 电子制作, 2022, 30 (7): 47 - 49, 36.
- [5] 马和平. NB-IoT 和 LoRa 相结合的无线技术在水厂中的应用研究 [J]. 工业控制计算机, 2021, 34 (11): 43 - 44.
- [6] 廖志远, 杨顺韬. LoRa 技术在低功耗智能抄表系统中的应用研究 [J]. 广西民族大学学报 (自然科学版), 2021, 27 (4): 87 - 90.
- [7] 付建文, 蒋昱麒. 基于 LoRa 技术的远程抄表系统设计 [J]. 电子设计工程, 2019, 27 (15): 157 - 160, 165.
- [8] 蓝 岚. 基于 LoRa 技术的费控表和集抄系统的研究及应用 [J]. 农村电气化, 2021, 10 (3): 5 - 7.
- [9] 孙 曼, 张乃谦, 金立标, 等. 基于 LoRa 标准的 MAC 层协议研究 [J]. 电视技术, 2016, 40 (10): 77 - 81.
- [10] 柳永波. 基于 LoRa 的无线自组网 MAC 协议研究 [D]. 西安: 西安电子科技大学, 2017.
- [11] 陈孝松. LoRa 低成本全双工网关的设计与实现 [J]. 无线电