

基于 C⁴ISR 系统仿真的报文编解码方法研究

肖凡, 刘忠, 黄金才

XIAO Fan, LIU Zhong, HUANG Jin-cai

国防科技大学 信息系统与管理学院, 长沙 410073

College of Information & Management, National University of Defence and Technology, Changsha 410073, China

E-mail: nudt_xf@163.com

XIAO Fan, LIU Zhong, HUANG Jin-cai. Study of message encoding and decoding method based on C⁴ISR system simulation. Computer Engineering and Applications, 2008, 44(7): 217-219.

Abstract: Several key problems of message encoding and decoding in C⁴ISR system simulation are analyzed, and new methods are put out. The detail design methods of storing rules of message encoding and decoding, data structure and function of encoding and decoding are brought out. And it also gives flow charts of message encoding and decoding. A novel method is designed to solve the problems of encoding and decoding in C⁴ISR system simulation, such as large number of messages and diverse encoding and decoding styles. And it is compared with the normal methods, so its advantage and disadvantage can be clearly shown. Rules of message encoding and decoding are stored in XML. Data structure and function of encoding and decoding are described through C++. The data flow drawing and main flow chart of program are given from the aspect of software engineering. Simulation results show that the design method of message encoding and decoding function is very novel, efficient and extensible.

Key words: message; encoding; decoding; frame

摘要:分析了 C⁴ISR 系统仿真中报文编解码的几个关键问题, 逐个给出了独到的解决方法。从编解码规则存储、数据结构和编解码函数的设计几个方面, 并且结合报文编解码的流程给出了 C⁴ISR 系统仿真中报文编解码的详细设计方法。针对 C⁴ISR 系统报文的复杂性和灵活性, 提出了一种新颖的编解码方法, 有效地解决了报文种类多、编解码方式灵活等问题; 而且, 与传统的方法进行了比较, 从而可以看出其编解码的优缺点。采用现在流行的 XML 文档对编解码规则进行存储; 数据结构和编解码函数采用 C++ 语言进行描述; 从软件工程的角度给出了数据流图和主要的程序流程图。仿真结果证明, 编解码函数的设计方法思路新颖、编解码效率高、可扩展性好。

关键词: 报文; 编码; 解码; 帧

文章编号: 1002-8331(2008)07-0217-03 **文献标识码:** A **中图分类号:** TP391.9

1 引言

C⁴ISR 系统已经成为各国军事专家研究的热点, 随之而来的是大量的 C⁴ISR 系统的计算机仿真。在实际的 C⁴ISR 系统中, 各类信息都是以简练的密文进行传输的, 信息发送的频率极高, 而且考虑到军事信息的保密性, 报文的格式和长度往往会进行变化。这使得在仿真软件的设计过程中要对大量的各类报文进行编解码, 编解码的设计效率与灵活性直接关系到整个仿真系统的成败, 因此, 报文编码在 C⁴ISR 系统的计算机仿真中占有核心地位。

在 C⁴ISR 系统的仿真软件开发过程中, 报文的编解码工作涉及到 C⁴ISR 系统的每个方面。报文编码规则复杂, 不固定而且报文的使用频率极高, 这使得报文编解码成为仿真过程中必须妥善解决而且具有一定难度的核心问题。仿真中报文编解码需要解决以下几个问题:

(1) 如何将各类信息的编解码规则存储为程序语言可以识

别的格式?

(2) 如何确定适应各类编解码规则的报文编解码数据结构?

(3) 如何使得编、解码函数的设计既高效(内存占用少、编解码时间短)又具有编解码信息单元长度的可变性和信息单元内的位置可变性?

以上三个问题在一定程度上反映了报文编解码的设计过程, 解决了这三个问题, C⁴ISR 系统的仿真中报文编解码问题也就基本上得到了解决。为此, 下面从以上几个方面并结合整个编解码流程来给出 C⁴ISR 系统的仿真软件开发过程中的报文编解码方法; 所提出的编解码方法整体思路设计合理, 编解码处理手段新颖, 而且效率很高。

2 信息编解码规则的存储

在 C⁴ISR 系统中, 信息编解码规则非常多, 往往是厚厚的几本编解码规则展现在软件设计人员的面前, 对这些规则进行

作者简介: 肖凡(1980-), 男, 助教, 研究方向: 计算机仿真、信息决策; 刘忠(1968-), 男, 博士, 副教授, 研究方向: 计算机仿真、作战计划生成; 黄金才(1973-), 男, 博士, 副教授, 研究方向: 计算机仿真、信息决策。

收稿日期: 2006-06-20 **修回日期:** 2007-09-16

整理是非常重要的,而且整理的原则应该是一方面方便对数据的查找和修改,另一方面易于程序语言对其的控制。因此,可以考虑采用数据库技术和 XML 等技术对其进行存储。考虑到数据修改的方便性和程序的交互等方面的原因本文采用 XML 文档存储信息编解码规则。

利用 XML 文档存储编解码规则时,关键是 XML 文档结构的设计,因为 XML 文档结构在一定程度上会影响编解码的效率甚至是整个仿真软件的效率。按照逐级分类原则对编解码规则进行分类存储。比如气候情况这样一条信息的编解码规则存取为 XML 文档后,其结构可能如图 1 所示。

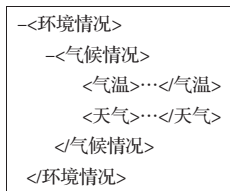


图 1 XML 分类存储编解码规则示例

图中的省略号代表对应的编解码规则。当然,XML 除了提供各项编解码规则外,还应该提供编解码信息单元长度。当把所有的编解码规则都存储到 XML 文档后,就可以利用已有的 XML 文档进行报文编码与解码。当编解码信息单元长度或者是编解码信息单元内的信息位置需要改变时只需要修改 XML 的相应内容。

3 报文编解码的数据结构

为了对报文进行有效的编解码,其编码的数据结构的设计至关重要。在设计报文编解码的数据结构时,既要考虑 C-ISR 系统报文的特点,又要方便程序的开发。

一方面,在报文编解码时应该首先对整条报文编解码加以控制,比如应答控制、重发控制、纠错控制、目的地控制、发送通道控制等。将这种控制信息叫做报头编解码(帧头编解码),其数据结构可设计如下:

```

typedef struct FrameHead
{
    char* frame_number; //报文编号
    bool type; //报文类型(发送还是反馈)
    char* start; //发送方
    char* destination; //接收方
    byte number; //编解码信息单元数量
    short length; //编解码信息单元长度
    bool respension; //应答控制
    char* respension_number; //所应答报文的编号
    byte chunnel; //发送通道
    Unit* info; //第一个编解码信息单元
} FrameHead;

```

另一方面,每条报文都是由若干编解码信息单元组成,编解码信息单元的数据结构的设计直接关系到编解码的效率和可扩展性,其数据结构可设计如下:

```

typedef struct Unit
{
    char* unit_number; //信息单元编号
    bool send_element[MAX_ELE]; //有效信息标识
    Unit* next; //下一信息单元
}Unit;

```

其中,MAX_ELE 为编解码信息单元内最大的信息元素数量(比如,温度即可视为一个信息元素)。

4 报文编解码流程

程序设计时,主要关心两种流程:数据流程和程序流程。前者是从整体上把握编解码设计的体系;后者是详细的功能实现。

整个数据流程是这样的:首先,将编解码规则整理、存储为 XML 文档,然后,利用 XML 文档对报文进行编码,最后,再利用 XML 文档对密文进行解码。其数据流图(DFD)如图 2 所示。

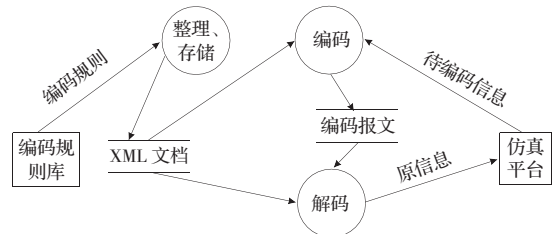


图 2 编解码设计数据流图

在进行程序编解码时,从帧头入手,先对控制信息进行编解码,然后对每个信息单元编码。对信息单元编码时,以信息元素为最小信息体,依照编解码规则对每个信息元素进行编解码。编码设计的程序主要流程如图 3 所示。解码即为编码的逆过程。

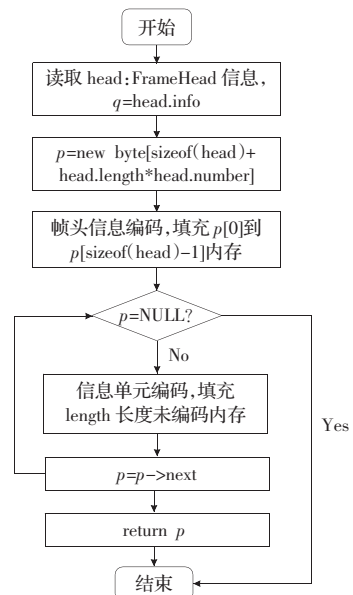


图 3 编码的主要流程图

5 编解码函数的设计

依据编解码规则的 XML 文档和报文编解码数据结构里的信息就可以对报文进行编码和解码了。编码的实质就是将信息依据编码规则转化成二进制的代码;解码是它的逆过程。

编码的实现其实就是对计算机内存的位操作。如果某项信息(信息元素)在编码信息单元内占用了 n_i 到 n_{i+1} 的 K 位,那么编码的任务就是依据实际信息将这 K 位填充。在对位进行操作时,一般的方法是先定位所操作的位所属的字节,然后取出这一字节的信息,再接下来用“|”操作对其进行写操作。比如,要将一个 128 位数据中的第 100 位置 1,先取出第 $100/8+$

1=13 个字节的信 息, 然后再将这个字节的信 息“1”上“1 000”。这样, 循环 K 次便可完成这个信息元素的编码。有一种情况就是连续的 K 位信息在一个字节内, 这时, 只需要进行一次操作便可完成此信息元素 K 位的编码。这种编码方法的优点是辅助编码的空间占用少; 速度也比较高, 尤其是信息元素编码都在一个字节内时。

在编码时, 还可以采取另外一种函数实现上更加快捷的方法。下面结合一个例子来说明。假设信息编码单元的长度为 128 位(FrameHead.length=128), 某信息元素位于第 S 到 E 位, 其值十进制数值为 test, 对其进行编码设计如下:

```
byte* encode (FrameHead head)
{
    byte* unit=new byte[sizeof(head)+head.length*head.number];
//编码内存
    if(! unit) return NULL;
    byte midway[128];          //辅助空间,方便处理
    memset(midway,0,128);     //初始化
    .....
    for(int k=S-1;k<E;k++)    //编码信息暂时放在 midway 中
    {
        midway[k]=test % 2;
        test /= 2;
    }
    .....
//其他信息元素编码
    .....
    for(int i=0;i<(128/8);i++) //编码信息转换到编码单元 unit
    {
        byte total=0;
        for(int j=0;j<8;j++)
        {
            total += midway[i*8+j];
        }
        unit[i]=total;
    }
    .....
    return unit;
}
```

由上可知, 这种编码方法的思想是先将编码信息的每一位放在一个字节空间中, 然后按由低到高的顺序将每 8 个字节空间的信息转换到信息编码单元中的一个字节。这样的好处是便于处理跨字节的信息元素编码, 思路新颖。

显然, 可以根据以上思路设计出两种解码函数, 它们分别是上述编码函数的逆过程。经过测试, 发现在信息单元长度不超过 512 位时, 上述两种方法的效率基本一样; 在信息单元长度很大时, 第二种方法会稍快些。

6 编解码仿真算例

按照以上思路, 飞机 1 报告自身位置的报文编解码的设计

如下。

飞机自身的位置信息属于态势信息, 包括高度(H)、经度(LONG)和纬度(lat)。编解码的任务就是处理这三种信息。

首先, 将编解码规则整理成 XML 文档。这一步是编解码的前提和基础。一般, 一个 XML 文档对应一种规则, 整理一次以后, 以后的编解码即可反复利用此 XML 文档。

然后, 填充编解码的帧头信息(帧头编解码)。编码时可设计如下:

```
FrameHead PosHead;Unit *p=NULL;    //帧头,信息单元申明
//信息单元处理
    p->unit_number="State_SelfPos";
    memset(p->send_elemet,0,MAX_ELE);
    p->send_elemet[1]=p->send_elemet[2]=p->send_elemet[3]=1;
//假定高度、纬度和经度是规则中的前 3 个信息元素
    p->next=NULL;
//帧头处理
    PosHead.frame_number="PosHead_06-2-5_1";//填写报文编号
    PosHead.type=1;          //报文类型(发送)
    PosHead.start="plane_1"; //发送方
    PosHead.destination="alert"; //接收方
    PosHead.number=1;        //编码信息单元数量
    PosHead.length=128;      //编解码信息单元长度
    PosHead.responion=1;     //应答控制(确认收到)
    PosHead.responion_number=NULL; //所应答报文的编号
    PosHead.chunnel=1;       //发送通道
    PosHead.info=p;          //第一个编解码信息单元
```

接下来, 调用编码函数进行编码, 这样就将信息转化成为了二进制代码。即:

```
byte* sendcode=encode(PosHead);
最后,接收方解码此信息,调用解码函数:
FrameHead* message=decode(sendcode);
```

7 结束语

文中提及的报文编、解码方法, 在仿真软件的开发中已经得到应用。实践证明, 这种编码和解码效率很高、设计灵活, 完全可以满足 C⁴ISR 系统仿真的需要, 而且, 它的可扩展性好, 可以为其他环境下仿真的通信编码提供框架和设计思路。

参考文献:

- [1] 张连庆, 蔡亚军. 面向信息对抗的 C⁴ISR 系统作战效能评估指标研究[J]. 情报指挥控制系统与仿真技术, 2005, 27(4): 44-45.
- [2] Jacobson. Object-Oriented Software[M]. Englewood Cliffs, NJ: Prentice Hall, 1990.
- [3] 陈平, 褚华. 软件设计师教程[M]. 北京: 清华大学出版社, 2004.
- [4] 张维明. 信息系统建模[M]. 北京: 电子工业出版社, 2002.
- [5] Ranenbaum A S. Distributed operating systems[M]. [S.l.]: Prentice Hall Inc, 1996.
- [6] Horowitz E. Computer algorithms C++: C++ and pseudocode versions[M]. US: Comoputer Science Press, 1996.