

内燃机装配料信息结构的数学模型及系统变型^①

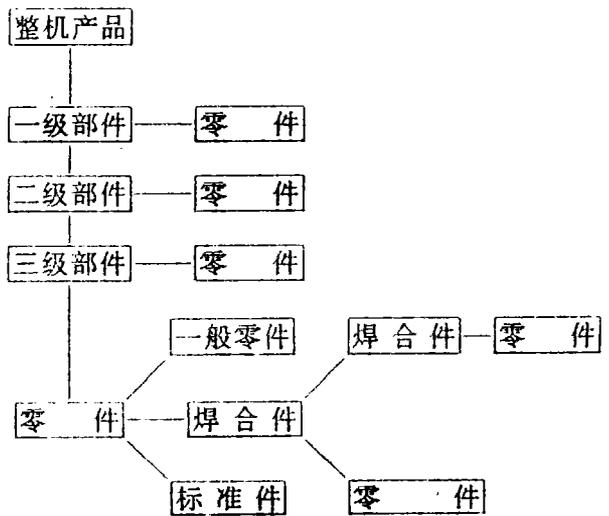
张光澄 雷 勇 张 雷

(四川联合大学(西区),成都 610065)

摘 要 基于文[1]提供的模式,本文旨在建立内燃机装配物料管理信息系统的数学模型,并采用面向对象的程序设计技术,根据不同的驱动因素实现产品的系统变型。

关键词 系统变型 信息系统 面向对象方法

系统变动中的数据重组在产品开发中表现为系统变型。由于用途、规格不同而产生同一型号的系列产品或者由于系统驱动因素的变化需要对同一产品,或者某些零、部件进行选配而造成产品变型,称之为系统变型,系统变型后装配成的最终产品称为变型产品。导致系统变动的驱动因素主要归结为用户需求的变化或生产条件和库存状况的限制等。



(图1 产品零部件分层)

作为动力源的内燃机,通常与汽车、拖拉机、农用车、发电设备配套使用,必须满足不同用途、不同规格的需求,还必须考虑与不同用户、不同主机配套连接设计等问题。因此产品的配套变型设计和管理是一项十分复杂而繁琐的工作。

本文根据内燃机装配物料管理工作的特点,把某种型号的内燃机整机产品(即装配后的最终产品)作为研究的系统,例如型号为490Q(DI)的直喷式柴油机。装配整机产品的零件和部件的全体构成系统的信息基元集,记为S,在S上分别建立系统的横向层次和纵

① 本文1996年3月25日收到。本文为国家自然科学基金资助项目(编号69374036)。

向层次信息结构模型,为实现系统变型,即产品变型奠定基础。

一、系统表示

1. 整机产品的表示

整机产品由若干零件和部件组成,为反映产品结构的层次关系,将零部件分成若干级。分析了其厂所有产品后得知,产品结构的层次最多有四级,部件最多为三级部件。对于焊合件,处理时作为一个零件,但为外协方便仍存储于它的下级零件和焊合件。故可将零部件从横向上分成四个层次:

- 第一层:装配整机的一级部件及所需零件;
- 第二层:二级部件及装配一级部件所需零件;
- 第三层:三级部件及装配二级部件所需零件;
- 第四层:装配及组成三级部件的零件。

各层次间的相互关系如图 1。

2. 产品的数据结构

根据产品结构的层次性可知,内燃机的产品结构是单向树状层次结构,且以图号,父图号为指针,如图 2。

整机产品中任一零部件均有三个数据项:零部件标识,图号,父图号,三者唯一地标识某一零部件。其中:

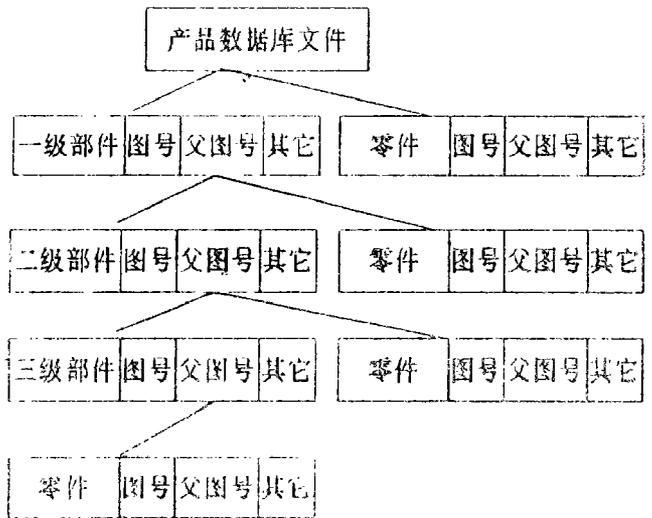


图 2 产品数据结构

- 图号是生产该零部件时的图号;
- 父图号标识该零部件是上一级哪个部件或焊合件的组成部分;
- 零部件标识是两位字符型代码,第一位指明该图号对应的物料是零件还是部件,并指明是哪一级部件;第二位指出该部件是否可分,即其下是否还有零部件,或者指出该零件是一般零件,焊合件还是标准件。

任一零件还有如下数据项(即图中的其它):物料名称,装配 I 位号,数量,单件重量,物料规格,修改标识等。

例如,一级部件标识为 10(不可分),11(可分);二、三级部件分别标识为 20(不可分),21(可分),30(不可分),31(可分);零件标识为 40(一般零件),42(标准件);焊合件分级表达,分别标识为 13(一级焊合件),23(二级焊合件),33(三级焊合件)等等。

二、 系统构成

整机产品下属各零部件所构成的分级结构可用各阶构造状态描述,下面着重剖析一阶构造状态。

设 S 为整机产品所有的零件和部件所构成的集合, R_1 为整机下第一层的零部件之集合, 则 R_1 为 S 上的等价关系。故得整机产品第一层零部件的划分 E :

$$E = S/R_1 = \{E_i | i \in I\}$$

其中 E_i 是第一层的零部件; I 为划分的指标集, 它表示整机产品分解成一级部件及相应零件的数目, 即第一层零部件的个数; E 表示整机产品。

显然, $\{E_i\}$ 中存在耦合关系, 因而须考虑 E 上的离散拓朴 $p(E)$ 。由于实际工作中的约束, 我们对 E 的分析不可能也不必要遍及 $p(E)$ 的每一个元素, 于是在 E 上建立半拓朴 E° 。下面以 490Q(DI) 型柴油机的第一层零部件为例作具体分析。

装配 490Q(DI) 的一级部件及相应零件共计 32 个, 以 $E_i (i = 1, 2, \dots, 32)$ 表之, 例如 E_1 表示机体总成, E_2 表示正时齿轮传动总成, E_3 表示气缸盖总成, E_4 表示活塞连杆总成……等等。

产品装配中, 某些零部件关联紧密, 形成耦合:

机体部分: $D_1 = \{E_1, E_2, E_4, E_5, E_6, E_7, E_{12}, E_{14}, E_{32}\}$

缸盖部分: $D_2 = \{E_3, E_8, E_9, E_{10}, E_{29}, E_{30}, E_{25}\}$

燃烧部分: $D_3 = \{E_{11}, E_{15}, E_{16}, E_{17}, E_{18}, E_{19}\}$

润滑部分: $D_4 = \{E_{20}, E_{21}, E_{22}\}$

冷却部分: $D_5 = \{E_{23}, E_{24}\}$

附件部分: $D_6 = \{E_{13}, E_{26}, E_{27}, E_{28}, E_{31}\}$

于是我们得到 E 上的半拓朴 $E^\circ = \{E_i, D_1, D_2, D_3, D_4, D_5, D_6 | i \in I\}$, 称之为“一阶可分析半拓朴”, (E, E°) 构成半拓朴空间。

定义 1, 设 $\rho_1: E^\circ \times A_1 \rightarrow V_1$, 对 $\forall F \in E^\circ, a_{1j} \in A_1$ 有 $\rho_1(F, a_{1j}) \in V_{1j}$, 则称 ρ_1 为一阶处理函数。其中 A_1CA 是第一层的变型因素集, $a_{1j} (j = 1, \dots, m)$ 为 A_1 的任一元素, 即为该层的某种变型因素; V_{1j} 是 a_{1j} 的属性值集合, 令 $V_1 = \cup_{j=1}^m V_{1j}$, 则 V_1 表第一层所有变型因素的属性值集合。

因此, 一阶处理函数的映象 V_1 构成变型产品的一阶结构。由此得到一阶组织结构子系统 (E, E_0) 和一阶可分析结构子系统 $(E^\circ \times A_1, V_1, \rho_1)$ 。其状态图如图 3 所示。类似地, 还可构造二阶组织结构子系统 (E_i, E_i°) , 二阶可分析结构子系统 $(E_i^\circ \times A_2, V_2, \rho_2)$; 三阶组织结构子系统 $(E_{i_2j}, E_{i_2j}^\circ)$, 三阶可分析结构子系统 $(E_{i_2j}^\circ \times A_3, V_3, \rho_3)$ 等等。

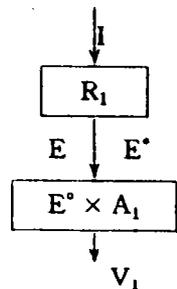


图 3 一阶构造状态图

以上建立了系统的横向层次结构,我们还可建立系统的纵向层次结构,即链丛结构,以及该链丛结构所具有的完备格性质。详见文[1]中信息结构的纵向层次结构。

三、系统变型

如前所述,由于系统驱动因素的变化需要对同一产品,或者某些零、部件进行选配而造成产品变型,即系统变型。选配的方式要有如下四种:

1. 更换:用另一套物料替换产品中的某一套物料,又可分为同机更换,异机更换两种;
2. 增加:在产品中增加某些零件或部件,也可分为同机增加,异机增加;
3. 删除:从产品中删除某些零件或部件;
4. 修改:直接对产品库中某一记录项的内容进行修改。

根据系统变型的驱动因素主要来自用户需求,库存状况和生产条件限制,我们抽象出系统变型因素集 A:

$A = \{ \text{用户需同机更换, 用户需异机更换, 用户需同机增加, 用户需异机增加, 用户需删除, 用户需修改, 库存需同机更换, 库存需异机更换, 库存需异机增加, 库存需删除, 库存需修改, 生产条件需异机增加, 生产条件需删除, 生产条件需修改} \}$ 。

显然各层可分析结构所定义的相应的变型因素集,它们都是 A 的子集。

由于以上系统存在变型,即按照不同的驱动因素改变产品,这样的系统用传统的系统设计方法是难以实现的。传统的系统设计方法是采用自顶向下的结构化程序设计方法,其设计的着眼点是功能,软件最终的结构是依据功能而定。而在系统变型中,功能是最不稳定的因素,采用这种设计方法将导致该系统不稳定。面向对象的设计方法,设计的着眼点是数据而不是功能。当系统功能发生变化时,只须变动局部数据,不致影响整个数据结构。这种设计方法的特点反映了文[1]中信息结构的纵向层次模型所具备的完备格的性质:增加或删除掉有限个链丛,不改变整个信息结构的完备性。

鉴于通用数据库为关系数据库,它要求关系数据。如何把层次型数据转换成关系型数据也是本文要讨论的重点。

四、面向对象程序设计

面向对象(Object Orientation)方法是一种与传统软件工程的功能方法完全不同的方法,它以对象为中心。面向对象的程序设计思路是从确定对象的实体开始,建立对象之间的层次结构,确定对象的属性,定义每一实体功能,最终形成软件结构。系统变型过程中,对象(即数据)具有相对稳定性,系统变型在面向对象程序设计中能得到妥善地处理。面向对象方法有如下特征:

1. 对象是数据和有关操作的封装体(称为类)。

在内燃机产品变型的系统中,将产品的组成数据和对这些数据的操作(增加、更换、删除、修改)封装成一个类。

2. 面向对象方法用消息将对象动态连接一起,其调用方式与传统的模块调用不同,采用了灵活的消息传递方式。

在产品变型的系统中,将引起内燃机产品变型(系统变型)的因素抽象的消息,将产品配套通知卡(见后文解释)的内容作为参数。用户发送消息,将参数传递到封装体(数据库操作),实现系统变型。

3. 对象体现了很好的信息隐蔽特性,任何对象均将其实现细节隐藏在它的内部。因此,无论是完善扩充对象的功能,还是修改对象的实现,其影响仅限于该对象的内部,而不会造成对象界影响。

目前的通用数据为关系数据库,面向对象数据库尚未投入使用。因此系统变型软件开发采用面向对象程度设计语言 C++调用关系数据库的嵌入式软件集成技术。

五、内燃机产品组成数据的规范处理

根据内燃机产品结构可知,该产品结构是单向树状层次结构,它既具有横向结构模型又具有纵向结构模型。从横向层次看,可以构造层次数据库。将用户需求等功能变化归纳成各种操作项目构成因素集,则可利用这些操作对层次数据库进行变换,从而生成所需要的变型产品库。从纵向层次看,上述信息结构模型成以键丛为基本结构的完备格结构。因此,在进行系统变型时,必须考虑到某节点的变换应同与它相连接的键丛一起变换,这样才能保证其结构的完备性。

由于内燃机产品结构为层次型,与关系数据库所要求的关系型不匹配,必须将它转换成关系型。关系数据模型(也称二维表)的特点如下:

1. 不仅实体用关系表示,实体间的联系也用关系表示;
2. 必须使用规范化的关系,即不允许“表中有表”。

鉴于关系数据库的以上特点,必须把内燃机产品库分解为若干并列的层数据库,即:基本产品库、一级部件库、二级部件库、三级部件库和零件库。各层数据库都包含有三个字段:零部件标识(发下简称标识)、图号和父图号,这三个字段作为各层数据库的主键(关键字段),唯一地确定一个零部件在数据库中的横向和纵向位置。

任意零部件的链关系的确定方法是:根据它的标识值确其层数,,再根据它的父图号查找一级部件。如此反复,可确定与之有关的整条链。

任意零部件的链丛关系的确定方法是:根据它的标识值确定它是否有下一级零部件,

若有则以其图号为下一级的父图号查找所有相关的下一级零部件。如此反复,直到找出该零部件下的所有链丛结构。

由于纵向链丛结构构成完备格,在进行系统变型时,有限条链丛的变化(增加或删除)不会导致整个结构的改变。

纵向链丛结构在关系数据库中用视图来表示。视图中由基表或其他视图导出的虚表,只在数据目录当中保留其逻辑定义,而不作为一个表实际存贮在数据库中。它既可参与数据库操作,又可大大缩小数据空间,有利于提高程序的运行速度。

在产品的系统变型中,将变型产品库定义为一个视图,该视图就是根据变型产品的链丛结构将相关的层数据库连接而成的虚表。

六、系统变型的实现

内燃机产品的系统变型的依据是配套通知卡,即产品变型的技术通知单。为便于计算机管理,设置配套卡表头库,其中数据项(配套卡号编码)是配套卡的顺序编码,由软件自动管理,变型需求则存贮在配套卡操作库中。一张完整的配套卡由表头和操作内容两部分组成(表头库及操作库结构略)。

如前所述,将系统变型的操作方式归纳成四种,即更换、增加、删除、修改。其操作内容分述如下:

同机更换时输入更换零部件和被更换零部件的图号、父图号、标识和层码;异机更换时还应输入各自所在的产品型号。

同机增加时,若物料名称唯一,也可输入物料名称,否则输入增加件图号、父图号,标识和层码;异机增加时还应输入各自所在的产品型号。删除或修改时应输入图号、父图号、标识和层码。

上述操作内容实质上是从横向层次对产品层数据库记录进行变换,即记录的更换、增加、删除和修改。从纵向层次看,在进行上述横向变换时,还必须考虑到系统上某节点的变换应同与之相连接的链丛一起变换,为此引入了事件驱动方法。

系统变型程序设计时,采用将数据嵌入 C++ 之中并用 C++ 操作数据库系统变型的方法。

在程序设计中我们把引起系统变型的因素抽象为消息,将配套通知卡的内容作为参数,而产品变型的迁配卡作为事件来驱动程序运行。事件驱动过程见图 4。

不同的驱动因素引起不同的系统变型,产生不同的消息处理过程.此处仅以零部件更换为例说明消息处理过程:

1. 按需更换零部件所在层的层数据库进行记录替换;
2. 创建变型产品库视图;
3. 从一级部件层开始按照本级部件图号与下一级部件父图号相同的条件,将所有相关的层数据库进行连接,结果送入视图,生成完整的变型产品库。

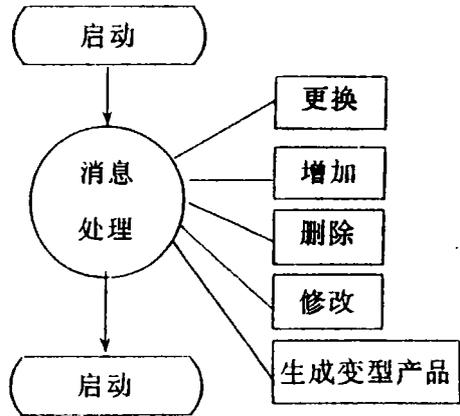


图 4 事件驱动方法

从上面可以看到,消息处理过程包括

横向层次变换(第 1 步)和纵向层次变换(第 3 步)两个方面。因此保证了系统的完整性。

归纳起来,系统变型全过程可用流程图表示如图 5:

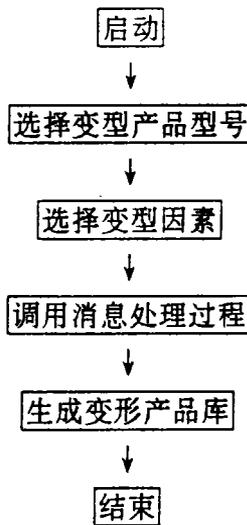


图 5 系统变型总的流程图

参 考 文 献

[1] 张光澄,刘晓石,等,《通用 MIS 信息结构的数学模型》,中国管理科学,第 4 卷第 1 期,1996。
 [2] 张玉琴,蒋维社,《面向对象语言和 C++ 讲座》,软件世界,1995 年第 2 期。
 [3] 史济民,《数据库原理与应用》,高等教育出版社,1987 年。
 [4] Borland C++ 型 Windows 应用程序,清华大学出版社,1993 年。

The Mathematical Model and The system Varying of Information Structure for Assembly Materials of Internal Combustion Engine

Zhang Guang Cheng Lei Yong Zhang Lei
(Sichuan Univerity, Chengdu, 610065)

Abstract: Based on the models which paper[1] has provided, this paper aims at establishment mathematical models of management information system for assembly materials of internal combustion enging. By object orienteds programming(OOP), this paper provides methods of the system varying in developments of internal combustion engine products , according to the different factors.

Key words: System Varying Information System Object Oriented Programing