4.1 存储规范

构建健康状态评估模型时涉及到大量模型相关数据，为了便于对模型的使用和更新，在存储模型数据时应遵循统一的存储流程和一致的编码及命名方式。为了应对意外事件的发生，还应对相关数据进行备份以便意外事件突发时能迅速恢复数据。其中，相关数据的编码遵循风洞设备PHM模型规范化管理研究中3.2节的模型编码规范，在此不再重复。因此，本小节将从存储流程和数据备份与恢复两个方面进行说明。

4.1.1 存储流程

对模型相关数据的存储流程进行规范就是要对模型存储过程中的各个步骤进行规范。模型相关数据的存储可以分为设计模型存储表、设置表的编码类型、设计表中字段、规范字段命名方式等步骤。

其中，设计模型存储表就是要对模型数据中应包含有哪些数据表、数据表的名称等进行设计并说明，构建出模型数据的整体框架。设置表的编码类型是指为了让计算机能够处理表中存储的数据，要对数据进行一定的编码处理并校验，为了能够支持中文，可以使用utf-8编码并使用utf-8-cgi校验。设计表中字段指对数据表中所包含的多个字段及其类型进行设计并说明，如表中应包含有id、time等字段用来作为表中数据的唯一标识和时间标志，id字段应设计为int类型等。规范字段命名指的是应该对表中各个字段的命名方式进行约束，要规定统一风格的命名方式保证全局数据名称的一致性。其整体的流程图如下图X所示：

以随机过程模型中的维纳过程模型为例，使用上述步骤存储维纳过程模型的具体流程如下：首先设计一张二维表来存储模型的各个参数，可以将其命名为wiener\_parameter；然后在建立该表的时候使用utf-8编码，这个步骤可以使用数

形状

低可信度描述已自动生成

图X 模型存储流程

据库管理工具在创建表格时快速完成；接下来设计表格中需要包含的字段。因为维纳过程涉及到的参数有漂移参数和扩散参数，所以对这两种参数各设置一个字段存放。由于这两种参数的类型均为浮点型，在创建字段时可以设置其类型均为FLOAT，这个步骤在使用数据库管理工具创建字段时也可以快速完成；最后是对刚才所设计的两个字段名称进行规范，将其名称划分为两部分，由于其均为维纳过程的参数，因此字段名称的前一部分设置为wiener，后面使用“\_”符号与第二部分隔开，第二部分名称使用其参数类型的英文名称，如漂移参数第二部分名称设置为drift，扩散参数第二部分名称设置为diffusion，对其进行整理得到两个参数字段的完整名称分别为wiener\_drift和wiener\_diffusion。除了这两个字段之外，也可以根据实际的需求选择是否设置模型参数存储时间、更新时间等参数，这些字段的设计步骤与参数字段的设计步骤相似。按照以上步骤就可以将一个维纳过程模型存储进数据库保存起来。

4.1.2 数据备份与恢复

在存储健康状态评估模型相关的数据文件时，可能会出现由于人为操作失误、软件BUG、硬件故障等原因导致数据丢失或损坏，还可能会由于特殊应用场景下基于时间点的数据恢复或数据迁移等原因需要对数据进行迁移处理，因此应该对健康状态评估模型相关数据进行备份，以便意外事件发生时能够迅速恢复数据。备份的内容应该包括数据库中的模型相关数据、数据库结构和数据库定义文件等。应按照数据库结构（字段）备份数据库中的数据。数据的备份方式主要分为完全备份、拆分备份、事务日志备份、差异备份、在线备份和离线备份几种方式，下面对各种备份方式分别进行介绍。

（1）完全备份

完全备份是指对数据库整体进行复制备份，包括数据库中的用户表、系统表、索引、视图和存储过程等所有数据库对象。由于备份整个数据库，因此这种方式需要耗费较多的时间和空间。

（2）拆分备份

拆分备份是指当数据库的容量大于备份介质的容量时，将数据库拆分成容量小于备份介质容量的若干个子数据库，然后将各子数据库分别备份到不同的备份介质上，保留原有的访问控制策略，并保证原数据库的完整性。

（3）事务日志备份

这种方式需要创建一个事务日志文件，用来记录数据库的改变。在备份的时候只需要复制自上次备份以来对数据库所做的改变，所以只需要很少的时间。

（4）差异备份

差异备份也叫增量备份。它是只备份数据库一部分的另一种方法，它不使用事务日志，相反，它使用整个数据库的一种新映象。它比最初的完全备份小，因为它只包含自上次完全备份以来所改变的数据库。这种备份方式的优点是存储和恢复速度较快。

在对数据库整体进行备份时，可以直接使用mysqldump备份工具或成熟的数据库管理工具提供的可视化界面快速完成，本小节主要介绍对数据库中单张数据表的备份及恢复流程。

在对数据库中存储的数据表进行单独备份时，需要对该表提前进行锁定，允许其他线程继续查询该表但是不能对表中数据进行修改。锁定表之后可以使用SQL语句对数据进行备份和恢复，备份完成之后再解除对该表的锁定。可以使用SELECT INTO OUTFILE语句备份数据，并用LOAD DATA INFILE语句恢复数据。但是这种方法只能导出数据的内容，而不包括表的结构，如果表的结构文件损坏，必须要先恢复原来的表的结构。对表中数据进行备份的步骤及SQL语句大概如下：

（1）首先锁定数据表，避免在备份过程中表被更新，使用SQL语句“LOCK TABLES tbl\_name READ”；

（2）然后导出数据，使用SQL语句“SELECT \* FROM tbl\_name INTO OUTFILE ‘tbl\_name.bak’”或是使用mysqldump备份工具在命令行窗口直接备份；

（3）备份完成后解锁表，使用SQL语句“UNLOCK TABLES”。

使用备份数据还原数据表时的步骤及SQL语句如下：

（1）首先为表增加锁定，使用SQL语句“LOCK TABLES tbl\_name WRITE”；

（2）然后恢复数据，使用SQL语句“LOAD DATA INFILE ‘tbl\_name.bak’ REPLACE INTO TABLE tbl\_name”；

（3）最后对表进行解锁。

数据表备份及恢复的完整流程如下图X所示：

形状

低可信度描述已自动生成

图X 数据表备份及恢复流程