
城市地下管线 PHM 系统使用文档

2015-10-28

目 录

1. 软件安装运行说明.....	2
1.1. 初次运行配置.....	2
1.1.1. 配置系统环境变量.....	2
1.1.2. 安装 MATLAB 库.....	2
1.1.3. 配置 oracle 数据库驱动.....	2
1.1.4. 测试数据库连接.....	2
1.2. 软件目录结构.....	3
1.3. 软件运行说明.....	4
1.3.1. 安装并运行服务.....	4
1.3.2. 停止服务.....	5
1.3.3. 卸载服务.....	5
1.3.4. 功能模块运行日志.....	5
1.3.5. 服务运行日志.....	5
2. 软件扩展说明.....	6
2.1. 对象数目扩展.....	6
2.1.1. 管线数目扩展.....	6
2.1.2. 传感器数目扩展.....	7
2.2. CPD 表的配置.....	7
2.2.1. 故障与故障的 CPD.....	8
2.2.2. 故障与传感器的 CPD.....	8
2.3 健康度评估模块的参数配置.....	9

2.4 故障预警模块的参数配置.....	10
3. 其他.....	11
3.1.....	11

1.软件安装运行说明

1.1. 初次运行配置

1.1.1. 配置系统环境变量

解压缩安装包，以管理员方式运行根目录下的 AddPhmPath.bat，写入完成后可能需要注销或重启以使得环境变量生效。

可在命令提示符中输入 `echo %PHM_HOME%` 查看新的环境变量是否生效；注意设置完环境变量后，如果移动了软件文件夹所在位置，需要再次运行 AddPhmPath.bat 重新设置环境变量。

1.1.2. 安装 MATLAB 库

运行 \MCR 配置 中的 MCRInstaller.exe，按步骤安装到某个位置（例如 D:\xxx\）。

1.1.3. 配置 oracle 数据库驱动

将 \MCR 配置 中的 add_classpath.bat 和 ojdbc6.jar 放到之前安装的 MATLAB 库的根目录下（例如 D:\xxx\v716\）

运行 add_classpath.bat 将 oracle 驱动的地址写入 MATLAB 环境，运行成功后会有提示。

1.1.4. 测试数据库连接

以上 3 步成功后，打开 \DBinfo.ini，修改其中的

service_name_oracle: 服务名/SID

username_oracle: 用户名

password_oracle: 密码

database_url_oracle: 将最后的 localhost:1521 改为目标 IP 和端口号

修改完成后，在 cmd 中运行 \linkDB_test\distrib\linkDB.exe（注意，直接打开无效）。初次运行等待时间较长，成功连接上数据库会有提示。

连接成功后，运行 \序列触发器 sql 脚本\create_trigger.sql 给故障预警表 YJ_WARNING_FORECAST 添加触发器。

1.2. 软件目录结构

本软件目录结构及相关说明如下：

根目录.

- |—Csharp 代码
 - | |—PHM 运行失败日志 ----- 存放软件运行异常日志
 - | |—WindowsServicePHM ----- 存放 PHM 服务 C#源码及 EXE 文件
- |—linkDB_test ----- 存放测试数据库连接的 EXE 文件
 - | |—distrib
 - | |—src
- |—MCR 配置 ----- 存放 MCR 配置文件
- |—健康度评估 ----- 存放健康度评估模块的相关文件
 - | |—Gas（燃气管线，暂无）
 - | |—Heat（热力管线，暂无）
 - | |—Rain（雨水管线，暂无）
 - | |—Sewage（污水管线，暂无）
 - | |—Water ----- 存放给水管线的 MATLAB 源码
 - | |—data ----- 存放数据文件

```

|           |—sql ----- 存放评估相关的 SQL 文件
|           |—udf ----- UserDefineFile 存放用户自定义文件
|—剩余寿命预测
|—序列触发器 sql 脚本 ----- 存放触发器相关 SQL 脚本
|—故障诊断 ----- 存放故障诊断模块的相关文件
|   |—Water
|   |   |—data
|   |   |—mat ----- 存放 MATLAB 类型数据文件
|   |   |—sql
|   |   |—udf ----- 存放诊断部分的用户配置文件
|   |—参数更新
|—故障预警 -----存放故障预警模块的相关文件
|   |—Water
|   |   |—data
|   |   |—sql
|   |   |—udf ----- 存放预警部分的用户配置文件
|—通用功能函数 ----- 存放 MATLAB 部分源码
|   |—数据库相关
|   |—配置相关

```

1.3. 软件运行说明

本软件以 windows 服务的方式运行，可设置为开机自启，运行过程无需人工干预。运行完成后将数据直接写入数据库的相关表格，需要在数据库中查看运行结果。

1.3.1. 安装并运行服务

以管理员方式运行 \Csharp 代码\InstallService.bat，执行服务的安装，安装完

毕后服务自动开始执行，可在 \Csharp 代码\ServicePHM.log 中查看服务实时运行状态。

1.3.2. 停止服务

在命令提示符 CMD 中输入 net stop PHM 可停止本服务的运行；输入 net start PHM 开启服务；也可在系统的服务管理界面中对本服务进行操作。

1.3.3. 卸载服务

以管理员方式运行 \Csharp 代码\UninstallService.bat，卸载本服务。若服务正在运行，则会先关闭服务，再进行卸载。

1.3.4. 功能模块运行日志

\Csharp 代码\PHM.log 是软件各个模块的运行日志，每执行一次诊断、评估、或预警，都会在 PHM.log 中写入实时运行状态日志。

若该功能模块运行成功，日志会保留一段时间，直到被下一次的日志覆盖；若运行失败，日志会被转入到 \Csharp 代码\PHM 运行失败日志 中保存，日志名称为运行的时间。便于在以后进行集中查看处理。

1.3.5. 服务运行日志

\Csharp 代码\ServicePHM.log 是 PHM 服务的运行日志，用于监控 PHM 服务的运行状态，主要包括服务的开启时刻，关闭时刻，每次执行诊断、评估、预警等功能模块的时间，执行是否成功，服务总运行时间，总运行次数等信息。

若服务启动失败或运行状态异常，能迅速在 ServicePHM.log 反映出来，便于查找异常原因，进行相应的软件维护措施。

2.软件扩展说明

本软件部分内容（核心算法除外）支持用户的自行定义和配置，主要包括对象数目的扩展、条件概率表 CPD 的配置、健康度评估模块中不同影响因素的权值配置、故障预警模块的相关配置等。

2.1. 对象数目扩展

对于同一类型的管线，本软件支持模型结构的横向扩展，即传感器数目和管线数目的扩展。但不支持纵向扩展（如增加故障种类、改变故障的因果关系等）。

以给水管线为例，数目扩展主要是通过修改传感器与管线的 D 矩阵文件——\\故障诊断\Water\data\udf\Dmatrix_Water.txt 实现。打开该文件，可看到传感器名称以及它能监测到的管线编号，如下图所示：

BG1	: 358: 371
SLYJ01_SL01	: 354: 375: 390
SLYJ01_SL02	: 384: 383: 382
DGN01	: 389: 383: 382
SLYJ01_SL03	: 385: 382: 381
SLYJ01_SL04	: 379: 378: 406
SLYJ01_SL05	: 407: 406: 403
SLYJ01_SL06	: 403: 404: 402
SLYJ01_SL07	: 410: 402: 399

第一行中，BG1 代表传感器编号，英文字符冒号“:”作为分隔符，后面的 358 和 371 表示编号为 358 和 371 的管线可以被传感器 BG1 监测到。

2.1.1. 管线数目扩展

如果新增的管线编号 900 也可以被 BG1 监测到，只需在 371 后面添加字符“: 900”即可，第一行变为“BG1 : 358: 371: 900”（注意冒号必须为英文字符）。

如果新增的管线无法被当前任意一个传感器监测到，则该管线属于不可探测管线，无需添加对应关系。

2.1.2. 传感器数目扩展

如果现在新增了传感器 T800，则需要根据传感器的安装位置找出该传感器能够监测到的管线编号，如 100, 200, 300 和 400。然后在 txt 文件的末尾添加一行“T800 : 100: 200: 300: 400”即可。

通过以上 2 步，就可以实现对管线数目和传感器数目的扩展，但还需要在 CPD 表中添加对应的 CPD 值才能够使软件正常运行（见 2.2. CPD 表的修改）。修改完成下一次执行时，软件就会读入 Dmatrix_Water.txt 中的新型结构，并自动搭建新型贝叶斯推理网络，完成后续的各个功能步骤，无需修改源码。

2.2. CPD 表的配置

条件概率表（CPD）的初始值需要由人工设定，它来自于先验知识，可看作是从以往大量的历史数据中统计出来的结果，体现了：

1. 各故障之间的因果关系大小。
2. 管线运行状态与传感器数值之间的关系。

以给水管线为例，打开 \故障诊断\Water\data\udf\CPD_Water.txt，如下图：

```

# CPD分为2部分，第一部分为故障之间的CPD
# 给水故障结构：腐蚀->破损->渗漏

# 对于离散节点来说，根节点为n个，则其cpd值有2^(n+1)
#####

corrosion      : 0.7, 0.3
broken         : 0.9, 0.2, 0.1, 0.8
leak           : 0.95, 0.2, 0.05, 0.8

#####
# 下面是针对每个传感器的CPD
# 如果是高斯节点，则先写mean，在写cov，用：分割

# 对于高斯节点来说，根节点为n个，则其cpd值有2^n
#####

JS_DMA1_BG1      : 1, 2, 3, 4: 0.1, 0.2, 0.3, 0.4
JS_DMA1_SLYJ01_SL01 : 1, 2, 3, 4, 5, 6, 7, 8: 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8
JS_DMA1_SLYJ01_SL02 : 1, 2, 3, 4, 5, 6, 7, 8: 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8

```

2.2.1. 故障与故障的 CPD

给水管线的故障结构为：腐蚀->破损->渗漏。每个节点都是离散节点，只有2种状态，发生/不发生，因此可用0/1表示。若某离散节点有n个父节点，则其CPD值有 2^{n+1} 个。因此此处3个节点的CPD值分别为2、4、4个。

这里分别用C、B、L代表3种故障，则第一行腐蚀故障的0.7, 0.3代表 $P(C=0)=0.7$, $P(C=1)=0.3$ ，即C不发生的概率为0.7，发生的概率为0.3。

第二行破损故障的0.9, 0.2, 0.1, 0.8代表 $P(B=0|C=0)=0.9$, $P(B=0|C=1)=0.2$, $P(B=1|C=0)=0.1$, $P(B=1|C=1)=0.8$ 。

第三行渗漏故障的0.95, 0.2, 0.05, 0.8代表 $P(L=0|B=0)=0.95$, $P(L=0|B=1)=0.2$, $P(L=1|B=0)=0.05$, $P(L=1|B=1)=0.8$ 。

注意设置数值时必须满足 $P(X=0|Y=0) + P(X=1|Y=0) = 1$ ，即在其他条件相同时，故障发生与不发生的概率之和必须为1。

2.2.2. 故障与传感器的 CPD

见上图的后半部分，传感器的输出是一个连续值，在同一状态下的输出应服从高斯分布 $N(\text{mean}, \text{cov})$ ，因此需要给出管线位于不同状态时传感器输出的均值mean和方差cov，这也可以从历史数据中学习得到。

注意，CPD_Water.txt 中传感器的数目、名称及排序必须与 Dmatrix_Water.txt 中的完全相同，否则会导致某些传感器找不到对应的 CPD 值，引起错误。因此设置 CPD_Water.txt 时必须对照着 Dmatrix_Water.txt 进行。

例如：对于 Dmatrix 的第一条记录 “BG1 : 358: 371”，其对应的 CPD 记录为 “BG1 : 1,2,3,4: 0.1,0.2,0.3,0.4”。这里用 A 代表管线 358 的渗漏故障，B 代表管线 371 的故障，则 CPD 含义如下表：

A（发生为 1）	B（不发生为 0）	高斯分布均值 mean	高斯分布方差 cov
0	0	1	0.1
1	0	2	0.2
0	1	3	0.3
1	1	4	0.4

即：当 A，B 均不发生时，传感器数值在 1 左右，方差为 0.1；

当 A 发生而 B 不发生时，传感器数值在 2 左右，方差为 0.2；

当 A 不发生而 B 发生时，传感器数值在 3 左右，方差为 0.3；

当 A 发生而 B 也发生时，传感器数值在 4 左右，方差为 0.4；

以此类推，若某传感器可以监测 3 个管线的渗漏状态，则其 CPD 值应为 8 个。即对于传感器高斯节点来说，n 个管线对应 2^n 个 CPD 值，配置时一定要保证数目的匹配，否则会报错。

因此，如果在 Dmatrix_Water.txt 中增加或修改了传感器数目和管线数目，必须要在 CPD_Water.txt 进行相应的修改或增加，并保证相关的匹配；否则会导致传感器找不到对应的 CPD，程序便无法运行。

2.3 健康度评估模块的参数配置

在健康度评估模块中，我们设置了影响健康度的 3 个要素：相对维修费用，相对破坏程度和专家建议，并依次分配影响权值为 0.4，0.4 和 0.2。以给水管线为例，打开 \\健康度评估\\Water\\data\\udf\\健康度权值表.txt，如下图：

```
# 三个要素：分别占据一定的权重

# 相对维修费用
repair_cost_weight = 0.4
repair_cost_corrosion = 1
repair_cost_broken = 50
repair_cost_leak = 100

# 相对破坏程度
damage_weight = 0.4
damage_corrosion = 1
damage_broken = 20
damage_leak = 100

# 专家建议
expert_weight = 0.2
expert_corrosion = 1
expert_broken = 10
expert_leak = 100

# 分级标准 满分100分 数字表示多少分以上才能得到该评价
very_healthy = 80
healthy = 60
semi_healthy = 40
ill = 20
badly_ill = 0
```

用户可根据实际情况修改上面三个因素的权值，一般确定后就不需要改动了。而本软件通过健康股评估算法对当前诊断结果进行处理后，会得到一个健康度分数，位于 0-100 之间。

最后一项分级标准，如 `very_healthy = 80` 表示分数在 80 以上才认为是“非常健康”；同理，`ill = 20` 表示分数在 20~40 之间的为“疾病”状态，而低于 20 为“严重疾病”状态。

用户可通过修改分级标准，来改变属于不同级别的阈值。

2.4 故障预警模块的参数配置

故障预警模块的用户可定义参数位于 `\故障预警\predict_config.ini` 中，包括 `n_train` 和 `n_predict`：`n_train` 代表预测所用到的记录数目，`n_predict` 代表要预测的数目。

故障预警即是利用最近的 `n_train` 条诊断记录来预测未来的 `n_predict` 次诊断结果。设置 `n_train` 是为了舍弃较早之前的数据，避免很久以前的数据对近期预

测的影响，用户可根据实际情况调整 `n_train` 的值。

而 `n_predict` 则是根据故障诊断的执行频率和故障预警的执行频率相除得到。即若一天执行 24 次诊断（每小时一次），并执行 1 次预警，则 `n_predict` 应该设置为 24，这样可以保证每次预警可以预测出未来一天内所有的诊断结果。因此 `n_predict` 在设置完成后不建议用户自行修改。

3.其他

3.1