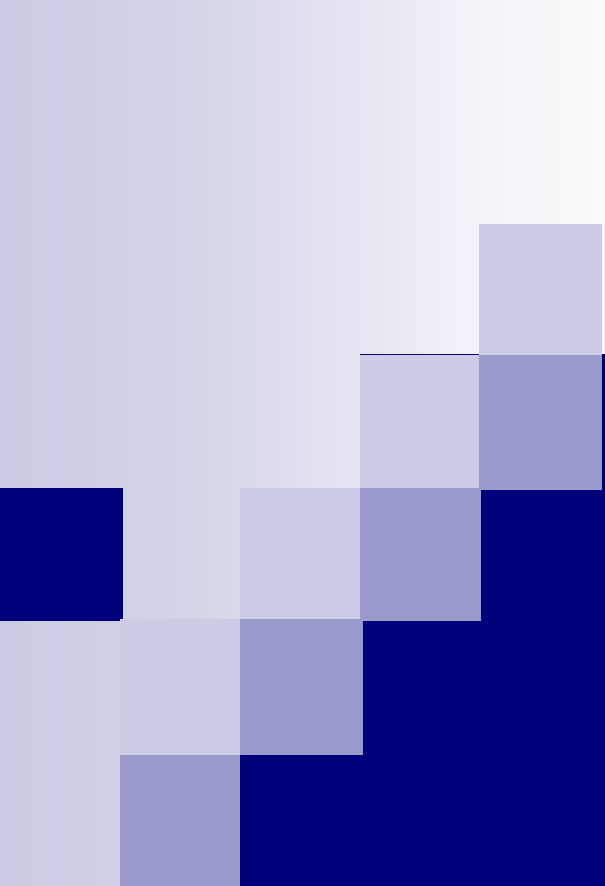




智能控制

主讲人：安剑奇 副教授

中国地质大学（武汉） 自动化学院

2020年3月

第四章 专家控制





专家控制

- 引言
- 专家控制的基本原理
- 专家控制应用举例
- 仿人智能控制



4.1 引言

➤ 专家控制的出现

- 经典控制方法在复杂应用中面临着诸多问题
 - ✓ 难以建立高度非线性、不确定对象的数学模型
 - ✓ 建立的模型过于复杂，无法得到实用的控制策略
 - ✓ 故障诊断、生产调度等高层决策需求
- 传统控制系统排斥人的干预，人-机之间缺乏交互
- 控制器对被控对象在环境中的参数、结构变化缺乏应变能力
- 人类的经验却可以很好的控制复杂的对象



4.1 引言

➤ 专家控制的出现

- 1983年，瑞典自适应控制专家Astrom提出将专家系统技术引入自动控制的思想，并在1986年正式提出了“专家控制”
- 专家系统能处理定性的、启发式或不确定的知识，经过推理来达到任务目标
- 将专家系统和控制理论结合导致了专家控制方法的出现



4.2.1 专家控制系统的基本内容

➤ 什么是专家系统

- 利用大量领域知识的大而复杂的人工智能系统
- 本质：一类包含着知识和推理的智能计算机程序
- 专家系统可以解决的问题包括：
 - ✓ 解释、预测、诊断、设计、规划、监视、指导和控制
- 专家系统主要由知识库和推理机构成



4.2.1 专家控制系统的基本内容

➤ 专家系统与传统程序的比较

■ 编程思想不同

- ✓ 传统程序 = 数据结构 + 算法
- ✓ 专家系统 = 知识 + 推理

■ 知识表达不同

- ✓ 传统程序：关于问题求解的知识隐含于程序中
- ✓ 专家系统：知识单独组成知识库，与推理机分离

■ 处理对象

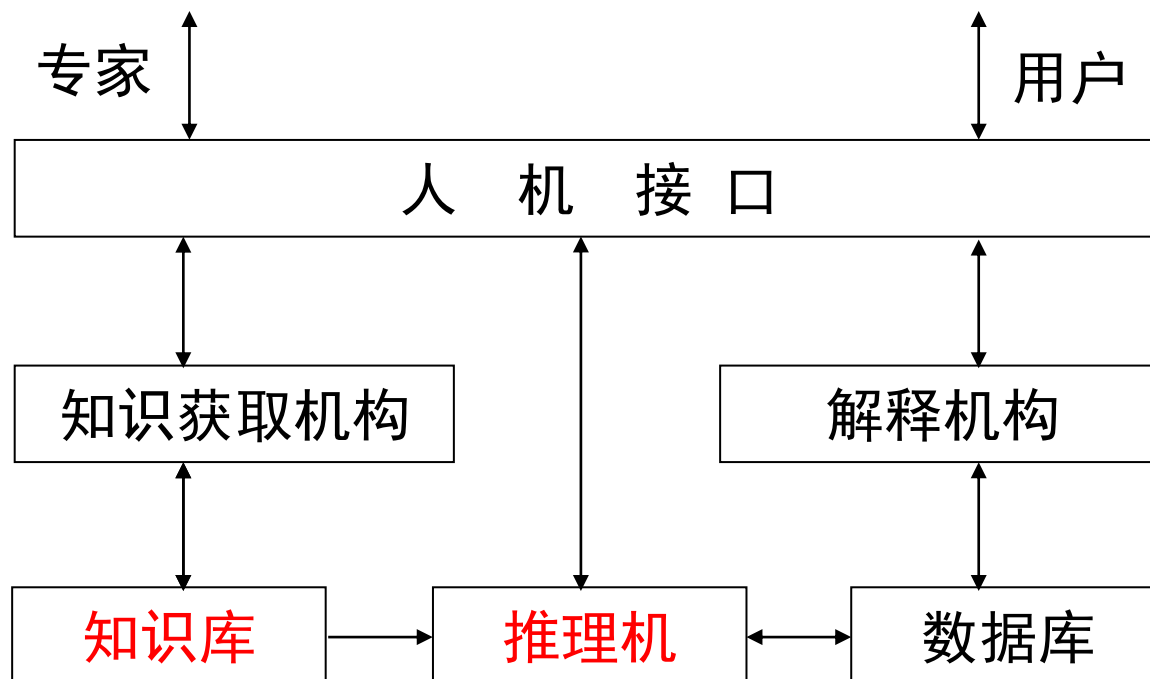
- ✓ 传统程序：数值计算和数据处理
- ✓ 专家系统：符号处理

■ 系统的体系结构不同



4.2.1 专家控制系统的基本内容

➤ 专家系统的基本结构





4.2.1 专家控制系统的基本内容

➤ 数据库

- 存放推理的原始数据、中间结果、控制信息等

➤ 知识库

- 存储领域专家的经验性知识、有关事实、一般常识等
- 知识库存放的信息具有规律性和普遍性，比数据库更稳定
- 来源于知识获取机构，为推理机提供求解问题所需的知识
- 知识库包含三类知识：
 - ✓ 基于专家经验的判断性规则
 - ✓ 用于推理、问题求解的控制性规则
 - ✓ 用于说明问题状态、事实和概念以及当前条件和常识等的数据库



4.2.1 专家控制系统的基本内容

➤ 推理机

- 利用知识库规则，对数据库的信息进行推理，得到结论决策
- 推理机和知识库彼此分离，便于维护和调整
- 推理机包括三种推理方式
 - ✓ 正向推理：从原始数据和已知条件得到结论
 - ✓ 反向推理：先提出假设，然后寻找证据；若证据存在，假设成立
 - ✓ 双向推理：运用正向推理提出假设结论，运用反向推理证实假设



4.2.1 专家控制系统的基本内容

➤ 知识获取机构

- 获取专家的领域知识
- 对知识库进行修改和维护，保持其内容的一致性和完整性
- 使专家系统具有**自学习**和**自适应**的特点

➤ 解释机构

- 对推理过程作出说明，并回答用户问题
- 解释机构使专家系统具有透明性，便于人机交互

➤ 人机接口

- 人机交互的界面
- 可以充分发挥人、机协作



4.2.1 专家控制系统的基本内容

➤ 专家系统的特点

- 启发性：常识、经验性推理
- 透明性：解释机构
- 灵活性：知识库更新方便

➤ 专家系统与专家控制的区别

- 专家系统能完成专门领域的功能，辅助用户决策；专家控制能进行独立的、实时的自动决策。
- 专家控制比专家系统对可靠性和抗干扰性要求更高
- 专家系统处于离线工作方式；而专家控制要求在线获取反馈信息，即处于在线工作方式



4.2.1 专家控制系统的基本内容

➤ 专家系统的实现

■ 知识工程

- ✓ 知识表达：规则、框架
- ✓ 知识的推理：推理的策略、竞争消解
- ✓ 知识库的维护：知识获取、知识优化、规则的管理

■ 系统的组织

- ✓ 知识库、推理机、数据库的信息交流



4.2.2 知识的表达

- 是知识的形式化，用机器表示知识的可行的、有效的、通用的原则和方法
- 知识的类型
 - ✓ 事实性知识、过程性知识、控制性知识
- 知识表达的种类
 - ✓ 图实类：与或图、Petri网、语义网等
 - ✓ 符号类：谓词逻辑表示法、状态空间表示法、产生式规则等
 - ✓ 结构类：框架、脚本、面向对象表示法、神经网络等



4.2.2 常用知识表达方法

➤ 产生式规则

- 产生式规则主要描述了前提和结论之间的关系。
- 其一般形式为：
 - ✓ IF <前提> THEN <结论>
- 在控制中，前提可以是控制的条件，如输出误差，结论是控制大小
 - ✓ IF 偏差 $|e|$ 增大， THEN 控制输入 u 增大
- 产生式通常用于表示事实、规则以及它们的不确定性度量，适合于表示事实性知识和规则性知识。



4.2.2 常用知识表达方法

➤ 谓词逻辑

- 一种符号逻辑，将命题解析成主词、谓词、逻辑连接词和量词
- 谓词刻划命题中个体词性质或事物之间的关系，命题的核心
- 例如 “温度是30℃、相对湿度50%的所有车间” 可表示成以下谓词公式的形式：
 - ✓ $(\forall x) \text{Room}(x) \wedge \text{Temp}(x, 30) \wedge \text{Humi}(x, 0.5)$
 - ✓ 其中Room、Temp、Humi为谓词， x 为变元



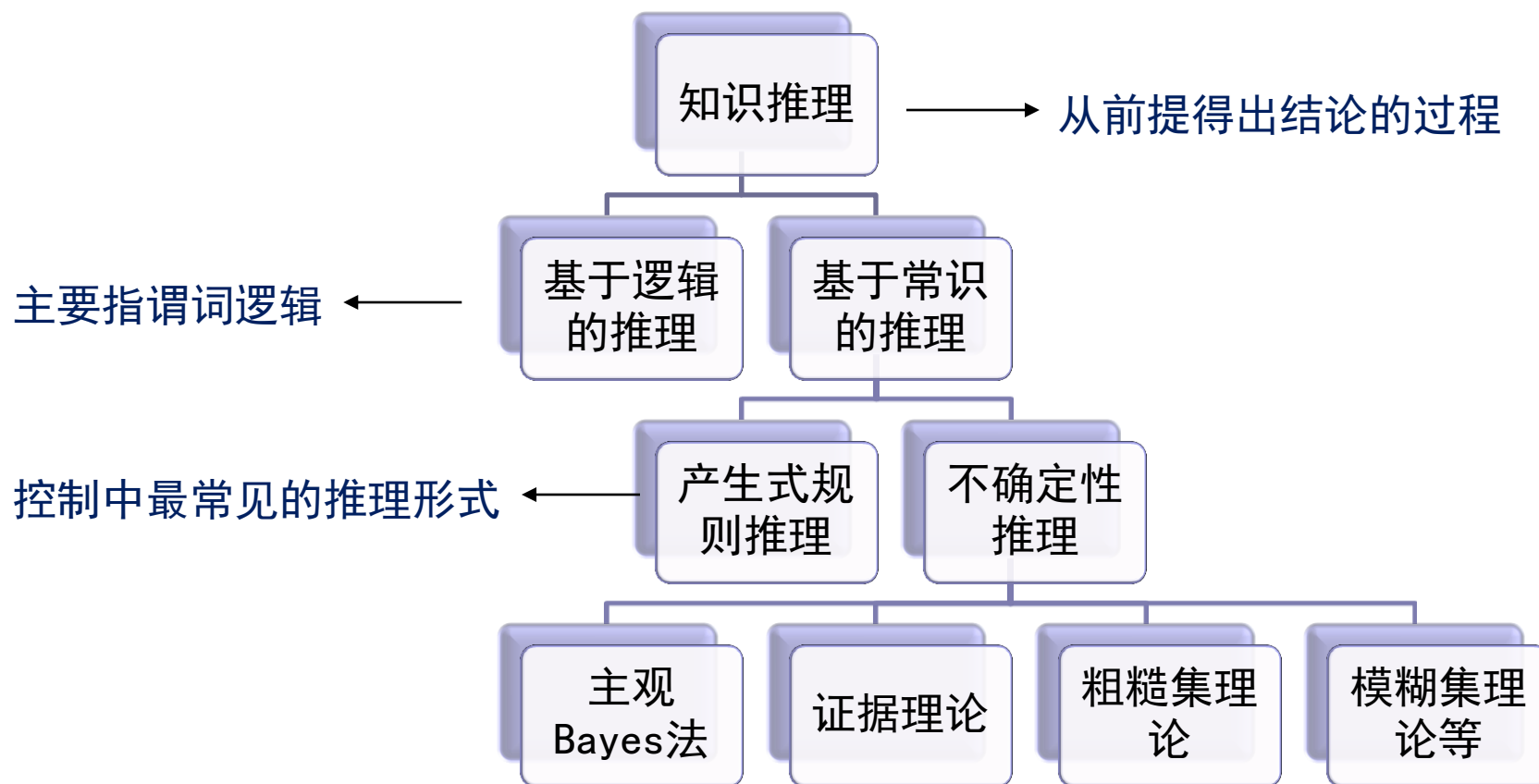
4.2.2 常用知识表达方法

➤ 面向对象表示方法

- 对象是指人类认识的客观事物，在形式上用以下四元组表示：
 - ✓ 对象::=<ID,DS,MS,MI>
 - ✓ ID(Identifier)为对象标识符，即对象名。如电机、回转窑。
 - ✓ DS(Data Structure)为对象的数据结构，描述了对象的一组状态属性。例如温度、压力等事实性知识。
 - ✓ MS(Method Set)为方法集合，可以用来描述各种决策。例如，对应不同状态下的控制策略。
 - ✓ MI(Message Interface)为对象接收外部信息的窗口，可以用来控制信息的流向，驱动事件的运行。



4.2.3 知识推理





4.2.3 知识推理-基于谓词的推理

➤ 假如假言推理

$$(E \rightarrow H) \wedge E \rightarrow H$$

- ✓ 代表“如果具有 $E \rightarrow H$ 的蕴涵关系，且证据 E 正确，则结论 H 正确”的推理规则

➤ 在一阶谓词逻辑的系统中

- ✓ 基于归结原理的推理方法已相当完备
 - ✓ 具有绝对的严格性和可靠，可实现推理的自动化
 - ✓ 但推理效率低
- 在专家控制系统中，主要采用常识性推理



4.2.3 知识推理-产生式规则推理

➤ 产生式规则

IF <前提> THEN <结论>

■ 模式匹配

- ✓ 将数据库中的数据与知识库中的规则中的前提进行匹配
- ✓ 完全匹配或足够近似匹配，则把这条规则标记为触发规则

■ 竞争消解

- ✓ 符合匹配的规则不止1条，存在竞争问题
- ✓ 根据一定的准则，从所有的触发规则中选1条，作为启用规则
- ✓ 或根据所有触发规则的结论进行综合，以获得一个最终的结论

■ 执行操作

- ✓ 根据推理的结论进行操作，并更新数据库的内容



4.2.3 知识推理-产生式规则推理

➤ 推理策略

■ 正向推理

- ✓ 一般情况下，从前提到结论的正向推理能满足控制的需要
- ✓ 但推理过程的中间结论过多，推理效率可能不高

■ 反向推理

- ✓ 从假定一些结论反向推理，寻找支持结论的证据
- ✓ 如果证据存在，则假设成立
- ✓ 若假设的好，效率高，但相对激进

■ 双向推理

- ✓ 综合正向和反向的推理



4.2.3 知识推理-产生式规则推理

➤ 正向推理

- 从已知**事实出发**，逐步推导出最后结论
- 推理机的工作过程，**事实驱动方式**
 - ✓ 用数据库中的事实与知识库中的规则前提条件进行匹配
 - ✓ 按冲突消解策略从匹配的规则中选择一条规则
 - ✓ 执行选中规则的动作（依次），不断获取中间结果。
 - ✓ 将中间结果作为新的前提，重复上述工作，直到得出结论或不再产生新的结果为止



4.2.3 知识推理-产生式规则推理

➤ 反向推理

- 首先**提出假设**，然后验证这些假设的真假性，**找到假设成立的所有证据或事实**
- 推理机的工作过程，**目标驱动方式**
 - ✓ 在知识库中寻找结论与假设目标一致的规则
 - ✓ 将所选规则的前提作为新的子目标假设
 - ✓ 重复上述工作，直到假设与事实匹配或不存在新的激活规则



4.2.3 知识推理-产生式规则推理

➤ 双向推理

- 推理机的工作过程

- ✓ 即自顶向下、又自底向上作双向推理，直至某个中间界面上两方向结果相符便成功结束

- 该方法较正向或反向推理所形成的中间结果少，在复杂系统中推理效果更高



4.2.3 知识推理-产生式规则推理

➤ 推理策略的选择

■ 推理的目标

- ✓ 如果目标是从一组给定事实出发，找出所有可能的结论，可使用正向推理。（例如疾病诊断）
- ✓ 如果目标是证实或否定某一特定结论，那么，通常使用反向推理。（例如电路设计）

■ 搜索空间的形状

- ✓ 事实较少，正向推理
- ✓ 结论较少，反向推理



4.2.3 知识推理-产生式规则推理

➤ 竞争消解

- 竞争：多个规则匹配
- 消解策略：根据问题定义优先级，决定首先应用哪些规则
 - ✓ 先到先得
 - ✓ 复杂的优先
 - ✓ 没用过的优先
 - ✓ 刚用过的优先
 - ✓ 产生数据冗余最小的优先
 - ✓ 随机选取



4.2.3 知识推理-不确定性推理

- 不确定性可以理解为在缺少足够信息的情况下做出判断
- 不确定性推理是另一大类重要的常识性推理
 - ✓ 主观Bayes法
 - ✓ 证据理论
 - ✓ 粗糙集理论
 - ✓ 模糊集理论
- 知识用不确定性度量来表示
- 适用于描述不确定、不知道、模糊的对象
- 推理过程就是不确定性的传播过程



4.2.3 知识推理-不确定性推理

- 由于知识本身的不精确和不完全，采用标准逻辑意义下的推理方法难以达到解决问题的目的
- 知识库是其核心，但往往包含模糊性、随机性、不可靠性或不知道等不确定性因素的知识
- 为了解决这种条件下的推理计算问题，不确定性推理方法应运而生



4.2.3 知识推理-不确定性推理

- 在领域专家给出的规则强度
- 用户给出的原始证据的不确定性
- 定义一组函数，求出结论的不确定性度量
 - ✓ 不确定性的传递算法
 - ✓ 如何把证据及知识的不确定性传递给结论
 - ✓ 如何把初始证据的不确定性传递给结论



4.2.3 知识推理-不确定性推理

➤ 不确定性推理的两条研究路线

✓ 模型方法

- 在推理一级上扩展确定性推理
- 不确定证据和知识与某种度量标准对应
- 更新不确定性的算法，构成相应的不确定性推理模型

✓ 控制方法

- 在控制策略一级上处理不确定性
- 无统一的不确定性处理模型
- 其效果依赖于控制策略



4.2.3 知识推理-不确定性推理

➤ 主观Bayes方法

- ✓ 如果以主观概率 $P(E)$ 代表论据 E 为真的可能性
- ✓ $P(H)$ 代表假设结论 H 为真的可能性
- ✓ $P(E | H)$ 代表假设 H 为真时，论据 E 为真的可能性

➤ 当论据 E 为真时，代表假设 H 为真的可能性 $P(H | E)$ 可用 Bayes公式求取：

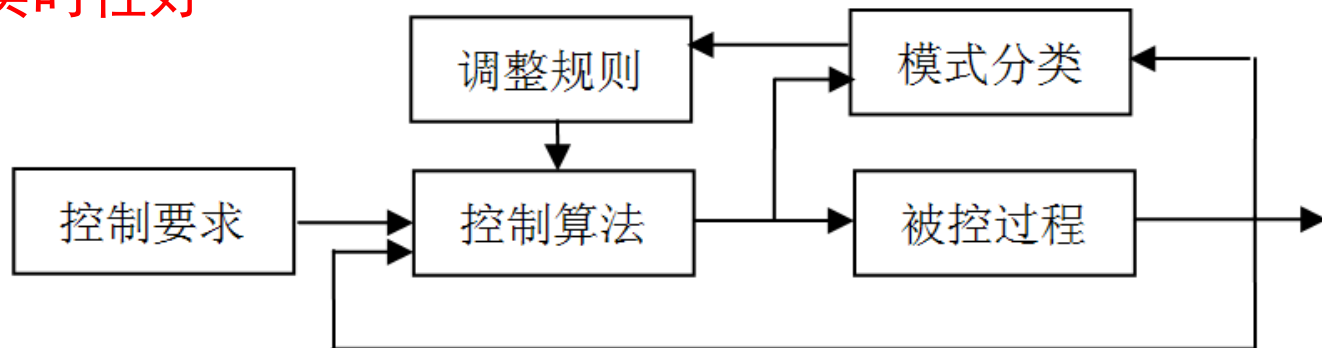
$$P(H | E) = \frac{P(E | H)P(H)}{P(E)}$$



4.2.4 专家控制系统的设计

➤ 专家控制器

- 基于规则的自适应专家控制器是自适应控制在知识层面的推广
- 通过对被控过程运行状态的识别，根据专家经验调节
- 不需要构造复杂的知识库、推理机
- 直接使用IF-THEN规则来描述专家经验，采用正向推理
- 结构简单、实时性好

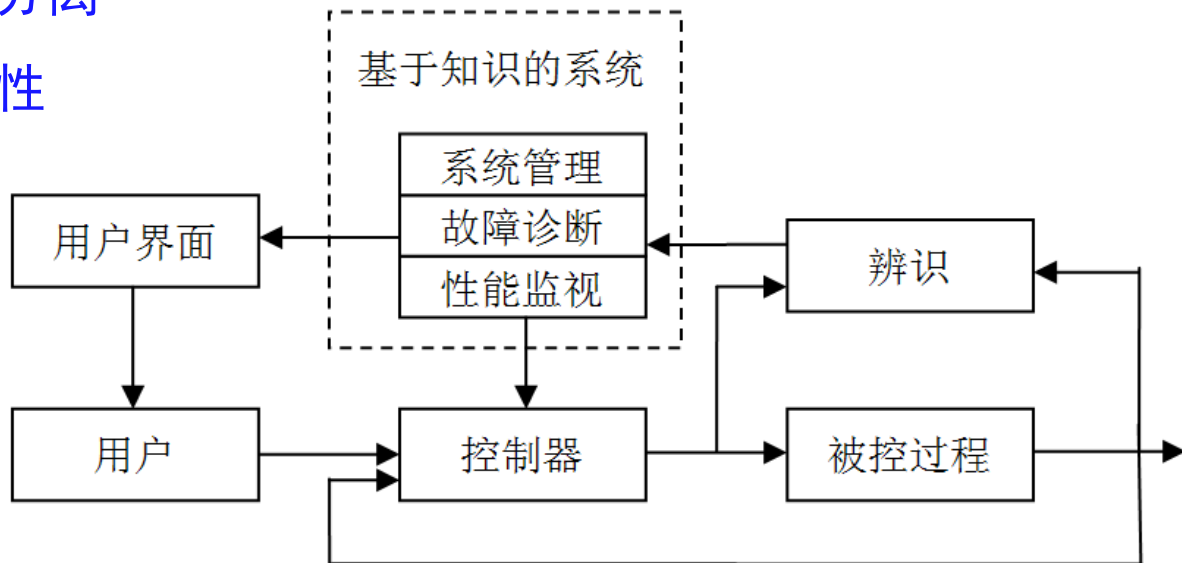




4.2.4 专家控制系统的设计

➤ 实时专家控制系统

- 为传统控制的底层加了一个**监督管理层**
- 是**基于知识的专家决策系统**
- 保留了专家系统的特色
 - ✓ 知识和数据的分离
 - ✓ 结构上的并行性





4.2.4 专家控制系统的设计

➤ 系统组织

■ 基于规则的组织方法

- 采用规则表示法来组织系统，实现实时应用
- 由外部事件所触发并驱动，一般没有明确的目标和计划
- 一旦开始推理，其过程就不能被中断和挂起，直至推理结束

■ 黑板法

- 设置一个全局的公共工作区
- 可接收推理机、知识库、数据库等知识源的访问和修改
- 复杂度低，方便维护，实时性和灵活性好

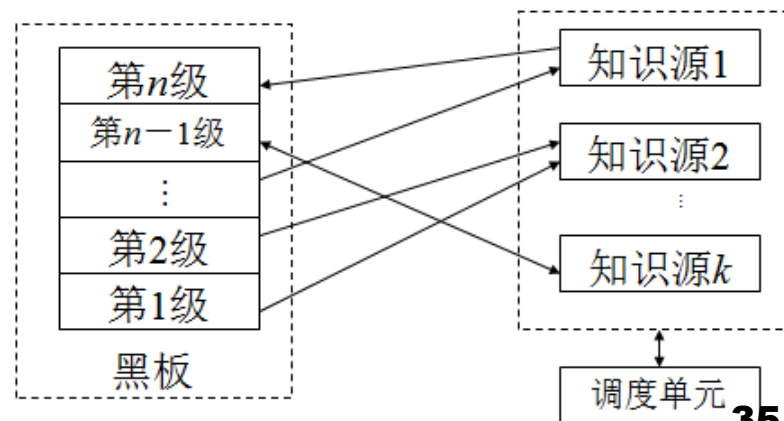


4.2.4 专家控制系统的设计

➤ 黑板法

- 由一组称为**知识资源(KS)**的独立模块和一块黑板组成求解系统
- 黑板：**全局数据库**，是一个可由一切KS访问的公用数据结构
- 知识资源：**子知识库**，有系统中专门领域的知识
- 同一个知识库，根据知识内涵不同，划分为不同知识源
- 黑板划分为不同层次，分别处理相关知识源的信息

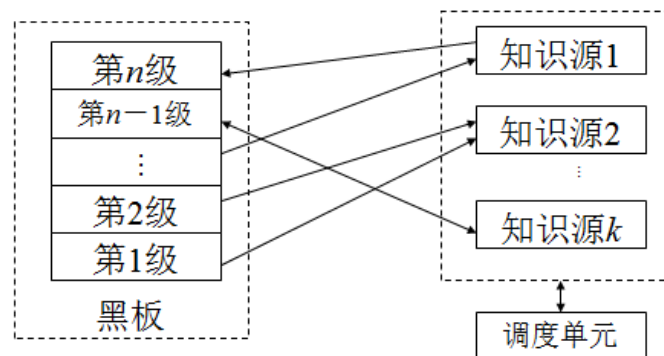
- ✓ 减少信息管理的复杂度
- ✓ 减轻通信的负担
- ✓ 提高知识库的可维护性
- ✓ 增强系统的实时性和灵活性





4.2.4 专家控制系统的设计

- 黑板的信息根据抽象性分层
 - ✓ 最底层是传感器的数据
 - ✓ 其次是提取的特征信息
 - ✓ 进一步是运行状态的假设、可行的控制策略等
- 系统的运作可由事件驱动
- 也可由调度单元根据启发性知识来调度





4.2.4 专家控制系统的设计

➤ 稳定性分析

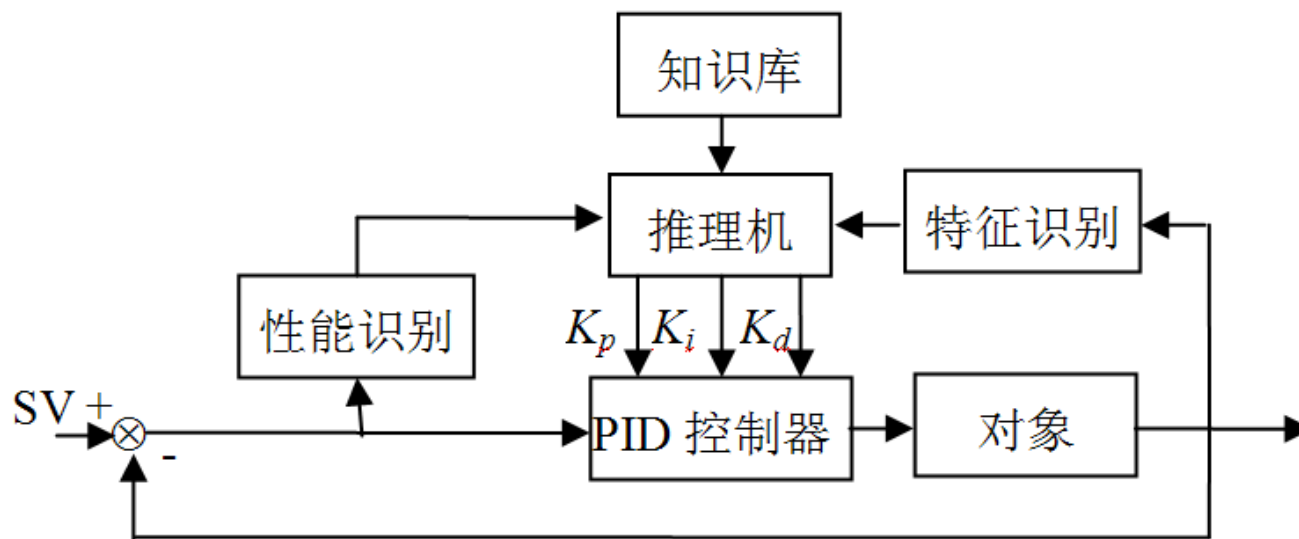
- 由于无法获得对象的模型，或无法获得控制算法的解析解，一般不能如经典算法一样地进行分析
- 主要依赖于以下的工作
 - ✓ **稳定性监控**：根据被控系统状态和输出，判断是否处于稳定状态
 - ✓ **控制策略的选择**：通过运行状态的识别，根据控制理论和经验，选择可以保证稳定的控制策略



4.3 专家控制应用举例

➤ 专家PID控制系统设计

- PID的智能控制算法包括参数整定和结构优化两个方面
- 基于专家系统的参数整定
 - ✓ K_p 、 K_i 、 K_d 的设置，对PID控制算法的控制效果至关重要





4.3 专家控制应用举例

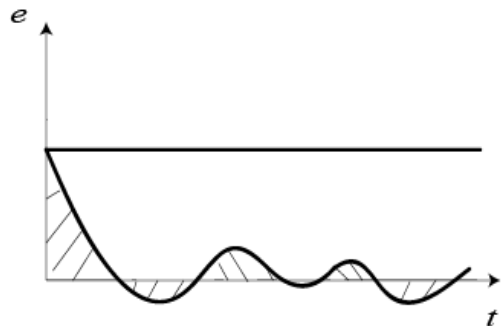
- 基于受控对象和控制规律的各种知识
- 利用专家经验来设计PID参数
- 无须知道被控对象的精确模型
- 专家PID控制是一种直接型专家控制器



4.3 专家控制应用举例

■ 参数整定的规则

- ✓ 当输出无波峰时，说明比例作用和积分作用偏小，可适当增加 K_p 和 K_I
- ✓ 当波形震荡发散时，说明比例作用过强或积分作用太大，可适当减小 K_p 和 K_I
- ✓ 超调量偏大时，则说明比例作用或积分作用偏大，可适当减小 K_p 和 K_I





4.3 专家控制应用举例

➤ 举例说明

- 令 $e(k)$ 表示离散化的当前采样时刻的误差值
- $e(k-1)$, $e(k-2)$ 分别表示前一个和前两个时刻的误差

$$\Delta e(k) = e(k) - e(k-1)$$

$$\Delta e(k-1) = e(k-1) - e(k-2)$$

- 根据误差及其变化, 对二阶系统单位阶跃响应误差曲线进行定性分析
 - ✓ 设定偏差的一个极大值, 记为 $M_{\max}$;
 - ✓ 设定一个偏差较大的中间值, 记为 M_{mid} ;
 - ✓ 设定一个偏差的极小值, 记为 $M_{\min}$

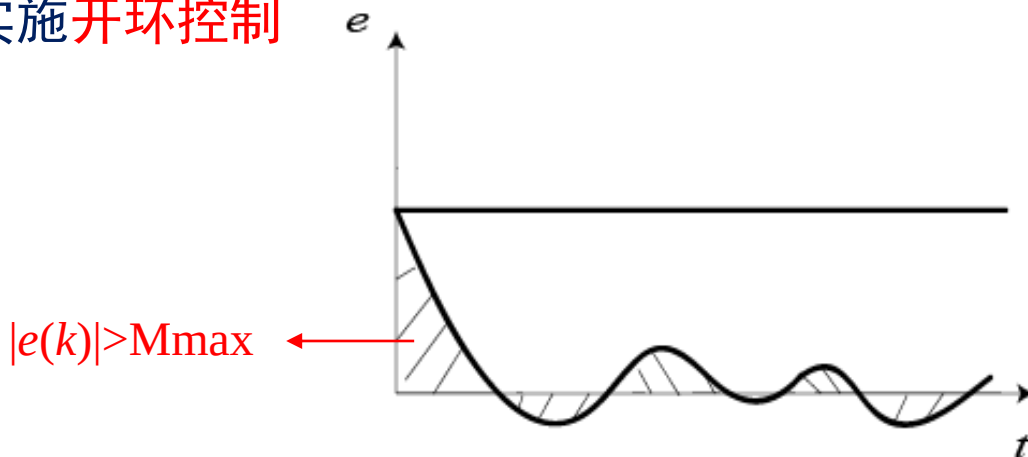


4.3 专家控制应用举例

➤ 规则设计

□ 如果 $|e(k)| > M_{\max}$

- ✓ 说明偏差的绝对值已经很大了，不论偏差变化趋势如何，都应该考虑控制器的输入应按最大（或最小）输出
- ✓ 以达到迅速调整偏差的效果，使偏差绝对值以最大的速度减小。
- ✓ 此时，它相当于实施开环控制



4.3 专家控制应用举例

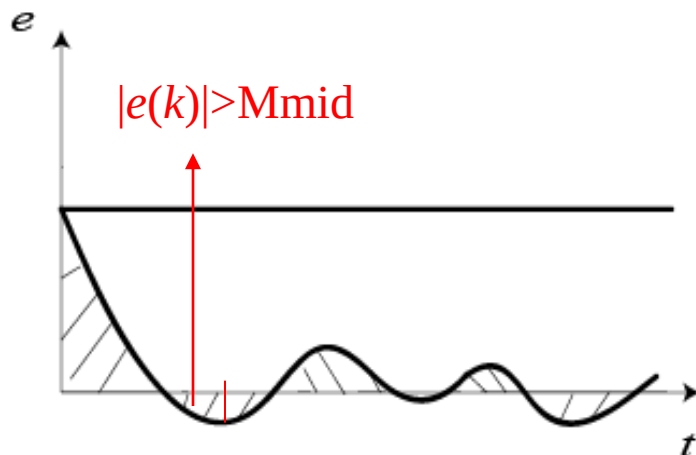
□ 当 $e(k) \cdot \Delta e(k) > 0$ 或者 $\Delta e(k) = 0$ 时

- 说明偏差在朝向偏差绝对值增大的方向变化，或为某一固定值，再判断偏差的绝对值与偏差的中间值 M_{mid} 之间的关系

✓ 此时如果 $|e(k)| > M_{mid}$ ，说明偏差也较大，可考虑由控制器实施较强的控制作用，控制器输出为

$$u(k) = u(k-1) + k\{k_p[e(k) - e(k-1)] + k_i e(k) + k_d[e(k) - 2e(k-1) + e(k-2)]\}$$

□ 其他情况请同学自行判断





4.3 专家控制应用举例

■ 下面给出部分程序代码

```

for k=1:1:500
    time(k)=k*ts; r(k)=1;
    u(k)=kp*x(1)+kd*x(2)+ki*x(3); % PID Controller
    % Expert control rule
    if abs(x(1))>0.8 % Rule1: Unclosed control rule
        u(k)=0.45;
    elseif abs(x(1))>0.40
        u(k)=0.40;
    elseif abs(x(1))>0.20
        u(k)=0.12;
    elseif abs(x(1))>0.01
        u(k)=0.10;
    end
    if x(1)*x(2)>0 && (x(2)==0) % Rule2
        if abs(x(1))>0.05
            u(k)=u_1+2*kp*x(1);
        else

```

```

        if u(k)>=10
            u(k)=10;
        end
        if u(k)<=-10
            u(k)=-10;
        end
        % Linear model
        y(k)=-den(2)*y_1-den(3)*y_2-den(4)*y_3+num(1)*u(k)...
            +num(2)*u_1+num(3)*u_2+num(4)*u_3;
        error(k)=r(k)-y(k);
        % -----Return of parameters-----
        u_3=u_2;u_2=u_1;u_1=u(k);
        y_3=y_2;y_2=y_1;y_1=y(k);
        x(1)=error(k); % Calculating P
        x2_1=x(2);
        x(2)=(error(k)-error_1)/ts; % Calculating D
        x(3)=x(3)+error(k)*ts; % Calculating I
        error_1=error(k);
    end
end

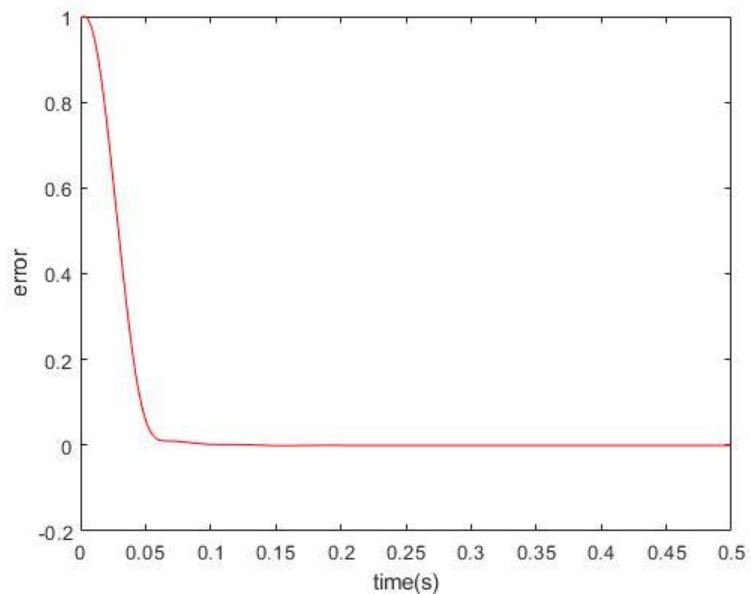
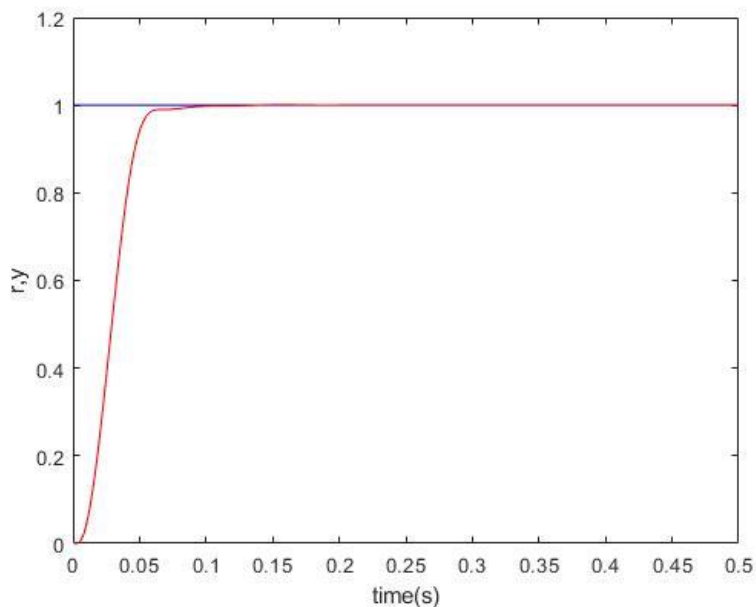
```



4.3 专家控制应用举例

■ 仿真结果

- 仿真系统为 $G_p(s) = \frac{523500}{s^3 + 87.35s^2 + 10470s}$





4.3 专家控制应用举例

➤ 过程专家控制系统

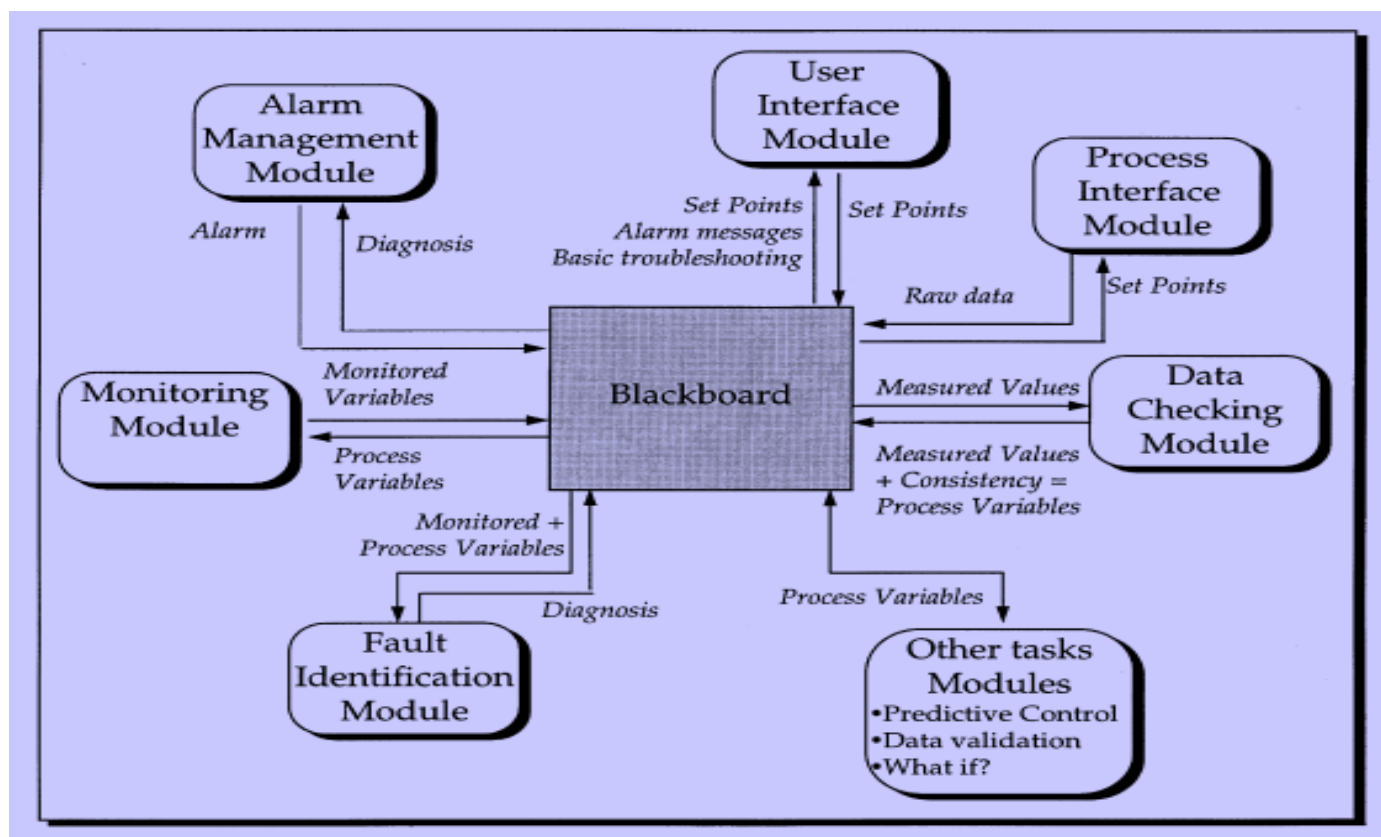
- 包括故障诊断、性能监视、生产调度、经营过程优化和重组等
- 需要对被控对象的工艺和工况具有深入的了解，来设计具有实时性的专家系统
- 已有公司开发出了专门的商业软件，其中最为著名的是美国Gensym公司的产品G2



4.3 专家控制应用举例

➤ 过程专家控制系统举例

■ 甜菜制糖工业监控系统





4.23 专家控制应用举例

➤ 模块举例说明

- 控制模块

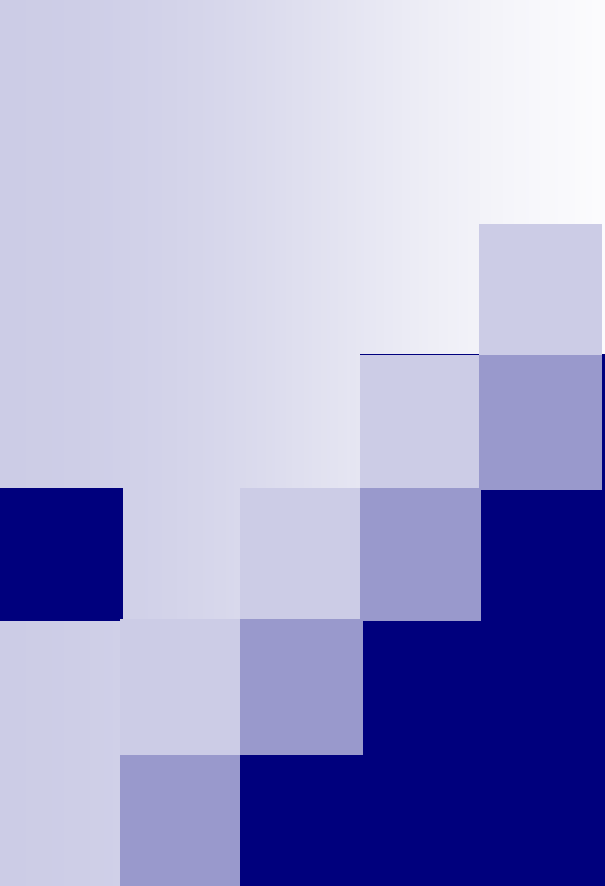
- ✓ 用户接口模块：提供原始数据、先验信息以及控制设定值等
- ✓ 数据校验模块：检验传感器数据一致性
- ✓ 其他模块：包括控制策略、数据有效性检验等内容

- 监督部分

- ✓ 监视模块：通过监视变量，监视系统运行的状态，发现故障
- ✓ 故障辨识模块：辨识故障的类型
- ✓ 报警管理模块：诊断故障的原因，并报警

- 人机部分

- ✓ 包括一个用户界面模块：用来提供控制要求和人机交互诊断



Thanks!