### 4.2.3模糊化和清晰化

量化因子是模糊控制器的输入接口，是对输入清晰量进行的放大或缩小变换。为了使这些清晰值能与语言表述的模糊规则相适配，进行近似推理，必须把它们变换成模糊量，即模糊子集。把这些输入的清晰值映射成模糊子集及其隶属函数的变换过程，称为模糊化。

在经过模糊逻辑推理之后输出的结论是模糊量，用它们不能直接推动执行机构进行控制，需要变换成清晰量。把模糊量变换成清晰量的过程，称为清晰化。

模糊化和清晰化模块是模糊控制器的两个重要的部件，尤其模糊化模块是绝不可缺少的。

1. 模糊化

通常经过采样得到的输入量都是清晰值，经过量化因子处理相当于进行一次比例变换，映射成模糊论域N上的某个实数值。这个实数值可能同时与N上的几个模糊子集有关，求出这个实数值隶属于各个相关模糊子集的隶属度，称为把清晰值模糊化。例如，在图3-1中，数0.4属于F集合“正好”的隶属度为0.6，而属于F集合“高”的隶属度为0.4……把实数值0.4属于各个模糊子集的隶属度都找出来，就称把0.4模糊化了。

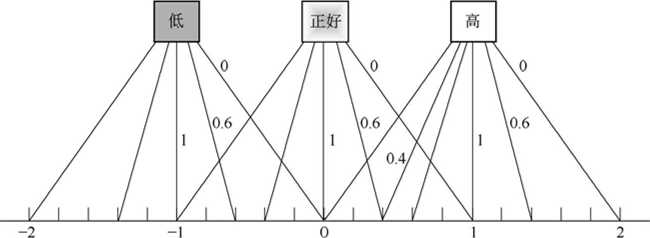


图 **错误!文档中没有指定样式的文字。**‑1 自然语言与论域（-2,2）之间的关系

为使每个实时输入变量的清晰值都能模糊化，首先需要确定覆盖在模糊论域N上模糊子集的数目，然后确定出各个模糊子集的隶属函数，这个过程称为进行模糊分布。

1. 模糊子集个数的选定

覆盖模糊论域的F子集数目应该适当，较多时虽可提高控制精度，但模糊规则数目相应地会增加得更快，致使运算量大幅增加。例如，一个二维F控制器，若覆盖偏差e论域的F子集个数为4，覆盖偏差变化率ec论域的F子集个数为3，二者搭配的结果模糊规则的数目为条；当覆盖e的F子集个数増为7，覆盖ec的F子集个数増为5时，则模糊规则数目就增加为条，较前增加了一倍还多。

因此，覆盖模糊论域的F子集数目应当取得恰当，根据经验和运算的方便性，一般覆盖整个模糊论域的F子集数目以3〜10个为宜，这样既可以确保一定的控制精度，又可以避免模糊规则数目的过多。

1. 模糊子集的分布

在模糊子集的个数确定之后，就要考虑模糊子集的分布，即模糊子集在模糊论域上的散布方式和情况。覆盖整个论域的F子集分布，应该具有三个基本特性：

完备性，即论域上任意一个元素至少得与一个F子集对应；

一致性，即论域上任意一个元素不得同时是两个F子集的核；

交互性，即论域上任何一个元素不能仅属于一个F集合。

此外，两个F子集的隶属函数交叉时，交叉点的隶属度值应取得合适，通常取为好。值大时，虽然能使系统的控制稳定性变好，变化平缓，但同时也会使系统的灵敏度变差；反之，值小时虽使控制灵敏度变好，但却使系统的稳定性变差，容易引起系统波动，若值过小将会使控制器的模糊性变差。

图4-8画出了几种在模糊论域[-3，3]上不可取的模糊分布，其原因是：

论域中[-1，-0.5]间的数值没被模糊子集涵盖；

相邻的模糊子集B、C和D相互交叠部分过多，致使模糊论域中数值1，同时以不小的隶属度隶属于B、C、D三个模糊子集；

模糊子集C和D隶属函数交点的隶属度是0.8，略显大了点。

为了避免上述不合理情况的发生，通常使一个模糊子集的“核”元素属于紧相邻那个模糊子集的隶属度尽量小，最好接近于零。

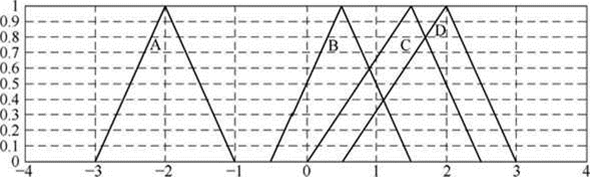
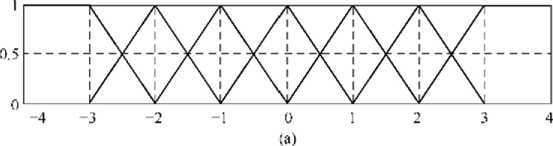
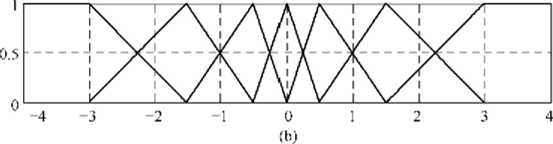


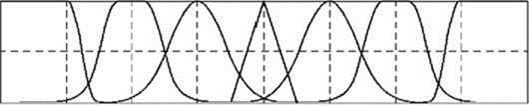
图4-8不太合理的模糊分布示例

图4-9画出了在论域[-4，4]上模糊子集合理分布的几个典型情况。

这些模糊子集的分布有的是均匀的，如图4-9(a)所示；有的是不均匀的，如图4-9（b）所示。不均匀分布中“零点”左右的模糊子集划分得较细，每个模糊子集占用的论域区段较小，这是因为零点对应于控制系统的工作点。这种分布可使控制器在零点附近的控制动作精确、细腻。







(C)

图4-9模糊子集分布合理的示例

由图4-9可见，模糊子集的隶属函数类型可以取成相同的，如图4-9(a)和（b)中除两个边缘外，所有隶属函数都是三角形的；也可以取成不全相同的，如图4-9(c)中从左到右分别为Z形、钟形、高斯型、三角形、高斯型、钟形和S形。

1. 隶属函数类型的选取

确定模糊子集就是确定模糊子集的隶属函数，根据模糊论域的离散性或连续性，隶属函数可以取成分离的或连续函数的形式。

用扎德表示法，模糊论域取离散值时，隶属函数形式为：



模糊论域为连续值时，隶属函数形式为：



隶属函数类型的选取没有统一的标准，取什么类型的隶属函数，完全取决于控制对象的不同情况加上设计者的习惯、运算简便和处理方便。一般模糊子集的隶属函数应该是连续函数，除在论域边界处外，都应该是对称的凸F集，这是人们思维渐变性的客观反映和实际需要。陡峭的隶属函数将使控制具有较高的分辨率，使控制灵敏度增高；反之，平缓的隶属函数会使控制的灵敏度降低。实用中，远离系统状态平衡点，偏差较大时可用低分辨率隶属函数；接近平衡点，偏差很小时，可采用高分辨率隶属函数，如图4-9(b)和图4-9(c)中零点附近就是如此。不过，最终都需要根据控制效果进行调整。

隶属函数的选取，包含着许多技术甚至是艺术成分，如何选取得更好，必须在实践中反复试验、探索方可确定，它不是一个纯技术问题，也不是一成不变的。

例如，世界上第一台利用模糊逻辑推理的模糊控制器——Mamdani等人的锅炉-蒸汽机模糊控制系统中，用了8个模糊子集涵盖变量的取值范围，他们选用的模糊子集及隶属函数不全一样。模糊子集的核，有的是平台（远离平衡点处），有的几乎是个点（平衡点附近）。现把它们选用的隶属函数都列在表4-1中，以供学习时参考。

表4-1 Mamdani蒸汽机模糊控制系统中选用的隶属函数

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 模糊子集名称 | 子集意义 | 隶属函数表达式 | 模糊子集名称 | 子集意义 | 隶属函数表达式 |
| PB | 正大 |  | NO | 负零 |  |
| PM | 正中 |  | NS | 负小 |  |
| PS | 正小 |  | NM | 负中 |  |
| PO | 正零 |  | NB | 负大 |  |

这些模糊子集的隶属函数形状如图4-10所示。

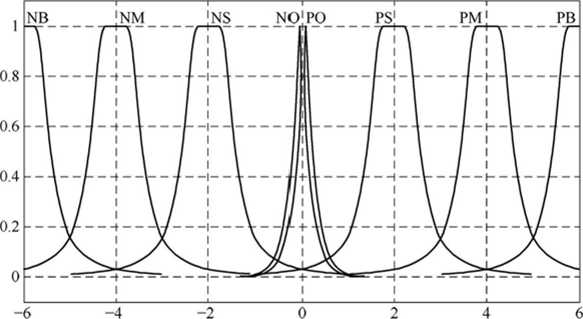


图4-10 Mamdani在蒸汽机模糊控制屮选用的隶属函数分布

特别应该注意的是，涵盖变量论域上的模糊子集分布是不均匀的：平衡点附近两个模糊子集隶属函数斜率很大，函数曲线相当陡峭，两个子集的“核”几乎是两个点，而且相当接近；而远离平衡点的模糊子集则大不相同，它们的“核”几乎是平台，彼此相距甚远。这样的分布可以提高平衡点附近的控制精度，又可使系统从远离平衡点处迅速趋向平衡点附近。

1. 清晰化

经过模糊逻辑推理后，输出的是模糊集合，由于它是多条模糊控制规则所得结论的综合，其隶属函数多数是分段、不规则的形状。清晰化的目的就是把它们等效成一个清晰值，即映射（变换）到一个代表性的数值上，这个任务由清晰化模块（F/D)完成。

清晰化的方法就是按照“言之有理、计算方便和具有连续性”的原则，在模糊集合的论域中找一个清晰数值来代表它。在2.5.3节已经介绍过具体的清晰化方法，常用的有面积平分法、面积中心法和最大隶属度法，由于最大隶属度法具有直观合理和计算方便的优点，实用中用得较多。

### 4.2.4模糊控制规则

模糊控制规则是模糊控制器的核心，它相当于传统控制系统中的校正装置或补偿器（如工业中经常使用的P1D控制器），是设计控制系统的主要内容。模糊控制规则的生成方法大体上有两种：一种是根据操作人员或专家对系统进行控制的实际操作经验和知识，归纳总结得出的；另一种是对系统进行测试实验，从分析系统的输入-输出数据中，归纳总结出来的。无论用什么方法生成的F控制规则，都可以用以下几种形式进行表述。

1. 语言型模糊规则

语言型模糊规则是由一系列的模糊条件语句组成的，即由许多模糊蕴涵关系“若……则……”（if…then…)构成。这些F条件语句是大量实验、观测和操作经验的归纳总结，在近似推理中认为它们是可靠的依据，是推理的出发点和得出正确结论的根据和基础，是“三段论”逻辑进行近似推理的大前提。把模糊控制规则用模糊条件语句表述的Mamdani控制器，称为语言型F控制器。

每条模糊条件语句都给出一个F蕴涵关系，即一条控制规则。若有条规则，就把它们表达的个F蕴涵关系做并运算，构成系统总的模糊蕴涵关系：



当采样得出的输入变量，经过模糊化后映射成模糊量X，则按近似推理合成法则，可以得到输出的模糊量为：



由于X未必能激活每条模糊规则，所以中可能有的项为零，求并时不予考虑。

由计算U的公式可知，构建语言型模糊控制器的关键，是根据经验总结出模糊规则，离线得出条F条件语句，从而求出系统的总模糊蕴涵关系，这是进行模糊推理的大前提。当采样得到输入量(小前提）时，将它模糊化并送入模糊推理机，经过近似推理，就能输出模糊量U。

二维语言型F控制器的结构，如图4-11所示。

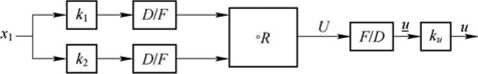


图4-11 二维语言型F控制器结构框图

例如，Mamdani等人的锅炉-蒸汽机模糊控制系统，用了两个双输入-单输出F控制器，以保持锅炉压力和活塞速度的恒定。一个F控制器根据蒸汽压力及其变化率，用输出控制供汽锅炉的加热能源；另一个F控制器根据蒸汽机转速及其变化率，用输出量去控制向蒸汽机供气阀门的开度。这里仅对后一个语言型F控制器的模糊控制规则作较详细的介绍。

用SE表示蒸汽机转速的偏差，用CSE表示转速偏差的变化率，用TC表示供气阀门的开度。每隔10s进行一次采样，测得输入量偏差SE和偏差变化率CSE，设输出模糊量为TC，它们的模糊分布归纳如下：

1. 用8个模糊子集涵盖转速偏差SE：正大(PB)、正中(PM)、正小(PS)、正零(PO)、负零(NO)、负小(NS)、负中(NM)和负大(NB)；
2. 用7个模糊子集涵盖转速偏差变化率CSE：正大(PB)、正中(PM)、正小(PS)、零(O)、负小(NS)、负中(NM)和负大(NB)；
3. 用5个模糊子集涵盖用于控制供气阀门开度的输出量TC：正大(PB)、正小(PS)、正好(O)、负小(NS)和负大(NB)。

根据操作经验总结出如下的语言型模糊控制规则：

1. 若偏差是负大且变化率不是负大或负中，则开度为正大；

（if SE = NB and CSE=not（NB or NM）then TC = PB）

1. 若转速偏差是负中且转速变化率是正小、正中或正大，则阀门开度为正小；

（if SE = NM and CSE=PS or PM or PB then TC=PS）

1. 若转速偏差是负小且转速变化率是正中或正大，则阀门开度为正小；

（if SE = NS and CSE=PM or PB then TC=PS）

1. 若转速偏差是负零且转速变化率是正大，则阀门开度为正小；

（if SE = NO and CSE=PB then TC=PS）

1. 若转速偏差是负零或正零且转速变化率是负小、零或正小，则阀门开度为负小；

(if SE = NO or PO and CSE= NS or NO or PS then TC = O)

1. 若转速偏差是正零且转速变化率是正大，则阀门开度为负小；

（if SE = PO and CSE = PB then TC=NS）

1. 若转速偏差是正小且转速变化率是正中或正大，则阀门开度为负小；

（if SE = PS and CSE = PM or PB then TC=NS）

1. 若转速偏差是正中且转速变化率是正小、正中或正大，则阀门开度为负小；

（if SE = PM and CSE=PS or PM or PB then TC=NS）

1. 若转速偏差是正大且转速变化率不是负中或负大，则阀门开度为负大。

（if SE = PB and CSE=not（NM or NB）then TC=NB）

这些语言型模糊规则，使用的都是两种典型的基本模糊条件句型：“if A then U”或“if A and B then U”。有的初看似乎不属于这两种之一，如①为“若偏差是负大且变化率不是负大或负中，则开度为正大”，但仔细分析它可以看成“若偏差是负大且变化率不是负大，则开度为正大”和“若偏差是负大且变化率不是负中，则开度为正大”两者的“析取”。9个句子中，有的比较简单，如、⑥就属于“if A and B then U”句型；虽然多数较为复杂，但都可以分解成多个基本模糊条件语句，如②、③、⑤、⑦、⑧、⑨即是。这些语句都具有模糊性，不够清晰，远没有传统控制的微分方程精确。但是Mamdani就是用这样的模糊语言规则，实现了锅炉-蒸汽机系统保持恒定转速的自动控制，完成了世界上第一台模糊逻辑控制系统这样的开创性工作。

1. 表格型模糊规则

由操作经验归纳总结出的模糊规则，用自然语言表述具有直观和易于理解的优点，但显得烦琐，在输入机器时并不方便，尤其是输入单片机时。不过，它们也可以转换成一个表格，使之具有直观简单、查算方便、快速简捷等优点，这类“表格型”控制规则尤其受到小型控制器的青睐。

此外，模糊规则表还可以直接由归纳总结测试时的操作数据得出，这种方法花费时间少，客观性强，生成过程简单，对于控制规则较多或很难用语言表述的控制规则更为实用，在工程上是一种非常有效的方法，自从1992年提出之后，已经广为流传。

1）把语言型模糊规则转换成模糊规则表

用语言表述的模糊规则，可以换成表格的形式。例如，上面的锅炉-蒸汽机的9项模糊规则，就可以表示成表4-2的形式，它包含着28条模糊规则：表头“列”的内容是覆盖偏差SE的模糊子集；表头“行”的内容是覆盖偏差变化率CSE的模糊子集；表芯所列的模糊子集是与“列”和“行”输入模糊子集对应的输出模糊子集TC。例如，当偏差SE=NS且偏差变化率CSE=PM时，从表中可查得与这组输入对应的输出量TC为PS，即SE中NS一行和CSE中PM一列交叉点的取值，它就表示此时输出量TC应为PS（正小）。

表4-2蒸汽机模糊规则表

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| CSE  SE | NB | NM | NS | N0 | PS | PM | PB |
| NB |  |  | PB | PB | PB | PB | PB |
| NM |  |  |  |  | PS | PS | PS |
| NS |  |  |  |  |  | PS | PS |
| NO |  |  | NS | NS | NS |  | PS |
| PO |  |  | NS | NS | NS |  | NS |
| PS |  |  |  |  |  | NS | NS |
| PM |  |  |  |  | NS | NS | NS |
| PB |  |  | NB | NB | NB | NB | NB |

每个控制系统的模糊规则表，都是事先做好的，相当于“语言控制规则”的模糊条件语句。只要将该表存入机器，使用中通过查表即可得出应有的输出，不必在控制过程中进行推理运算，非常简便。

2)用测试数据生成模糊控制规则表

模糊规则表4-2是从语言控制规则转换来的，建立它首先需要把操作经验总结成“if…then…”语言型模糊规则，这类工作往往需要有经验的技术工人或专家经过长时间的操作、观察和积累经验方可得出。这种积累、总结有下述困难：虽然技术人员能够很好地操作被控系统，但却未必能用清晰的语言表述成合格的模糊规则；②由于语言表述的控制规则具有模糊性，往往不够完整和充分，致使其他技术工人按文操作却达不到技术要求；③如果模糊规则太多，靠人工很难总结出来；使用语言型模糊规则时，每个采样周期内都要进行近似推理合成运算，对于实时性很强的系统很难完成。于是人们又开发出一些新的产生模糊规则的方法，王立新等人在1992年提出根据输入-输出数据组，经过分析、归纳、总结生成F控制表的方法，这种方法的实用性很强，下面简单介绍其原理和方法。

让有经验的专家对系统进行实际操作，对操作过程的输入-输出数据进行测试和记录，得出的大量输入-输出数据组中隐含着专家的经验、知识，可以用它们总结出控制规则来。

仍以双输入-单输出系统为例，设输入量为和，输出量为，故已测试出个数据组：



为测量的批次，，和组成第次测试得出的一个数据组。根据这个数据组，可以按下述步骤生成该系统的模糊控制规则。

（1）根据测量数据的取值范围确定变量的模糊论域，并选定覆盖模糊论域的模糊子集及其隶属函数

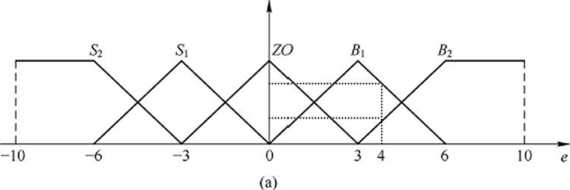
假设某系统的输入量为和，输出量为，已知它们的取值范围，即论域分别为：

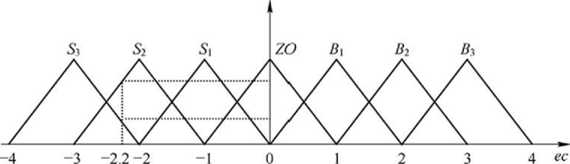
，，

对于每个论域，都用个完备模糊子集覆盖它，正整数可以因论域而异。为简单起见，其隶属函数都取成三角形。

按下述方法选取后，做出如图4-12所示的模糊子集及隶属函数分布。

1. 对于变量，取，即用5个F子集覆盖，F子集分别为（负大）、（负小）、ZO（零）、（正小）和（正大）。如图4-12**(**a**)**所示。
2. 对于变量，取，即用7个模糊子集涵盖，F子集分别为（负大）、（负中）、（负小）、ZO（零）、（正小）、（正中）和（正大）。如图4-12(b)所示。
3. 对于变量，取，用7个模糊子集涵盖，F子集分别为（负大）、（负中）、（负小）、ZO（零）、（正小）、（正中）和（正大）。如图4-12(c)所示。





(b)

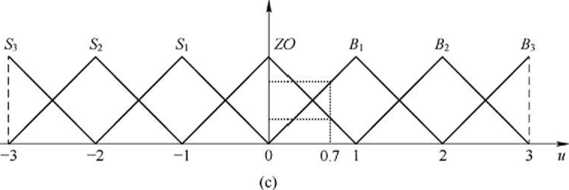


图4-12模糊子集分布及其隶属函数图

根据图4-12，很容易写出各隶属函数的数学表达式。例如，根据图4-12(b)，可以写出其中的表达式；根据图4-12(c)，可以写出其中的表达式，它们分别为：

变量的一个F子集 

变量的一个F子集 

利用图4-12和各模糊子集的隶属函数表达式，可以很容易地求出一个变量属于某个模糊子集的隶属度，这在下面将会用到。

（2）用每组数据构成一条模糊规则

对于每组数据，求出它们与之对应F子集的隶属度，选出隶属度最大的模糊子集，用于构成模糊控制规则。

假设已知、和分别为，，，如图4-12中竖直虚线所示，用这组数据可按下述方法构成一条模糊规则。

1. 对于变量，代入：

， 

可以算出。选其中大者，即用去构成模糊规则。

1. 对于变量，代入：

，

可以算出。选其中大者，即用去构成模糊规则。

1. 对于变量，代入：

，

可以算出。选其中大者，即用去构成模糊规则。

通过上述计算、选择，用数据组，，构成一条模糊规则：“if is ** and is  then is **”。

按此办法，每组数据都可构成一条模糊规则，然后再进行筛选。

（3）赋予每条规则一个强度，根据强度大小按“去小留大”原则决定矛盾规则的去留每组数据都能构成一条模糊规则，但在构成的许多条模糊规则中，可能由于测量误差等原因会出现矛盾现象，即有些规则前件和一样，而后件却不同。为了对矛盾的模糊规则进行筛选、取舍，对每条规则定义一个强度，筛选中按其强度决定取舍，如下所述。

把构成规则的每个数据属于其模糊子集的隶属度相乘，定义为该条规则的“强度”，括弧中的为规则的序号。遇到矛盾规则出现时，则根据其强度大小，按“去小留大”原则决定取舍。

例如，假设刚才得出的第5条规则为“if is ** and is  then is **”，按照刚才定义的强度，有。若代入、和的取值，有：



则认为该条规则的强度G(5)=0.373。

如果另有一组数据，，。据此可以构成第8条规则“if is ** and is  then is **”，它的前件与第5条规则的一样，显然第8条规则与第5条规则是矛盾的。而第8条规则的强度为：



于是按“去小留大”原则，保留第5条规则“if is ** and is  then is **”。

为了强调专家意见，有时在每条规则的“强度”计算中，可以乘以由专家提出的“认定强度”。

（4）确定模糊规则表

由于覆盖变量有5个F子集、覆盖变量有7个F子集，测得的数据经过以上处理后最多可得出35条控制规则，据此可以制出表4-3。

表4-3待填充的模糊规则表

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| ec  u  e |  |  |  | ZO |  |  |  |
|  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |
| ZO |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |

表4-3左侧的列表头表示覆盖偏差e的模糊子集；表格上面的行表头表示覆盖偏差变化率的模糊子集；表芯的每个方格则填入与所在行和列对应输出量的模糊子集。例如，我们前面确定的一条控制规则是“if is ** and is  then is **”，则在表4-3中找出变量为一行和变量为一列交汇的方格里填上**，……。如果数据足够多，就可以填满表4-3中所有空格。如此把所有得出的控制规则都填人表中，就得到该系统的完整模 糊规则表。

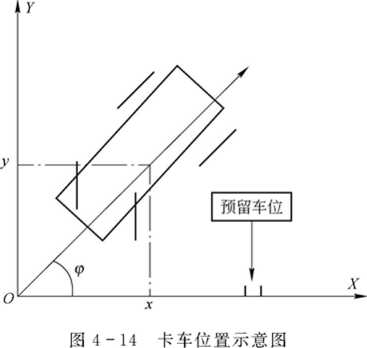
这样的模糊规则表，包含着操作人员的智慧、经验、技巧和推理，它替代了语言型模糊规则和近似推理过程。由数据归纳的模糊规则表，比人工总结的模糊条件语句更客观、公正，更方便于他人和机器使用。

一个二维表格型模糊控制器的结构，如图4-13所示。



图4-13表格型F控制器结构框图

由结构框图可知，实时采样得到的数据经过模糊化处理后输入机器，通过查询模糊规则表便可得到应有的输出模糊量，从而避免了近似推理过程。实际应用中，特别是在控制系统较为简单而采用单片机控制时，常常采用这种查表法。例如，设某时刻测得的输入量为和，经过模糊化处理后分别映射成模糊量**和，计算机只需根据输入数据，从表4-5中查出相应的输出量为**，就可把**输入到下一级清晰化模块（F/D)，用于去操作被控对象。

**例4-2** 如图4-14所示，某卡车（图中对“卡车”做了放大）所处位置由三个状态变量确定：卡车位置的坐标、和卡车前行方向与轴的夹角。已知，。为使它倒入预留车位，即达到状态，某熟练司机用“匀速后退”行驶，仅靠“控制方向盘”的操作完成了这一任务，方向盘的转角。

对司机倒车入位的操作过程进行了等时间间隔记录，把输入输出的数据经过反复试验筛选和整理，得出有代表性的部分数据，如表4\_4所示。

根据这些数据设计一个模糊控制器，使其功能可以代替司机的倒车入位操作。

表4-4 某司机倒车输入-输出数据表

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 次第 | 输入量 | | 输出量 | 次第 | 输入量 | | 输出量 |
|  |  |  |  |  |  |
| 0 | 1.00 | 0.00 | -19.00 | 9 | 8.72 | 65.99 | -9.55 |
| 1 | 1.95 | 9.37 | -14.95 | 10 | 9.01 | 70.85 | -8.50 |
| 2 | 2.88 | 18.23 | -16.90 | 11 | 9.28 | 74.98 | -7. 45 |
| 3 | 3.79 | 26.57 | -15.85 | 12 | 9.46 | 80. 70 | -6. 40 |
| 4 | 4.65 | 34.44 | -14.80 | 13 | 9.59 | 81.90 | -5.34 |
| 5 | 5.45 | 41.78 | -13.75 | 14 | 9.72 | 84. 57 | -4. 30 |
| 6 | 6.18 | 48.60 | -12.70 | 15 | 9.81 | 86.72 | -3.25 |
| 7 | 7.48 | 54.91 | -11.65 | 16 | 9.88 | 88. 34 | -2.20 |
| 8 | 7.99 | 60.71 | -10.60 | 17 | 9.91 | 89. 44 | 0. 00 |

解 根据这些数据，定义5个F子集覆盖输入量；定义7个F子集覆盖；定义7个F子集覆盖输出，假设这些F子集都取成三角形隶属函数，如图4-15所示。

由每组输入-输出数据产生一条控制规则，并根据前面介绍的方法算出相应的强度。例如，对于表4-4中第11批次测试数据为，按图4-15的模糊子集分布，可以算出 、、，据此可得出一条控制规则：“if  is  and  is  then  is ”。

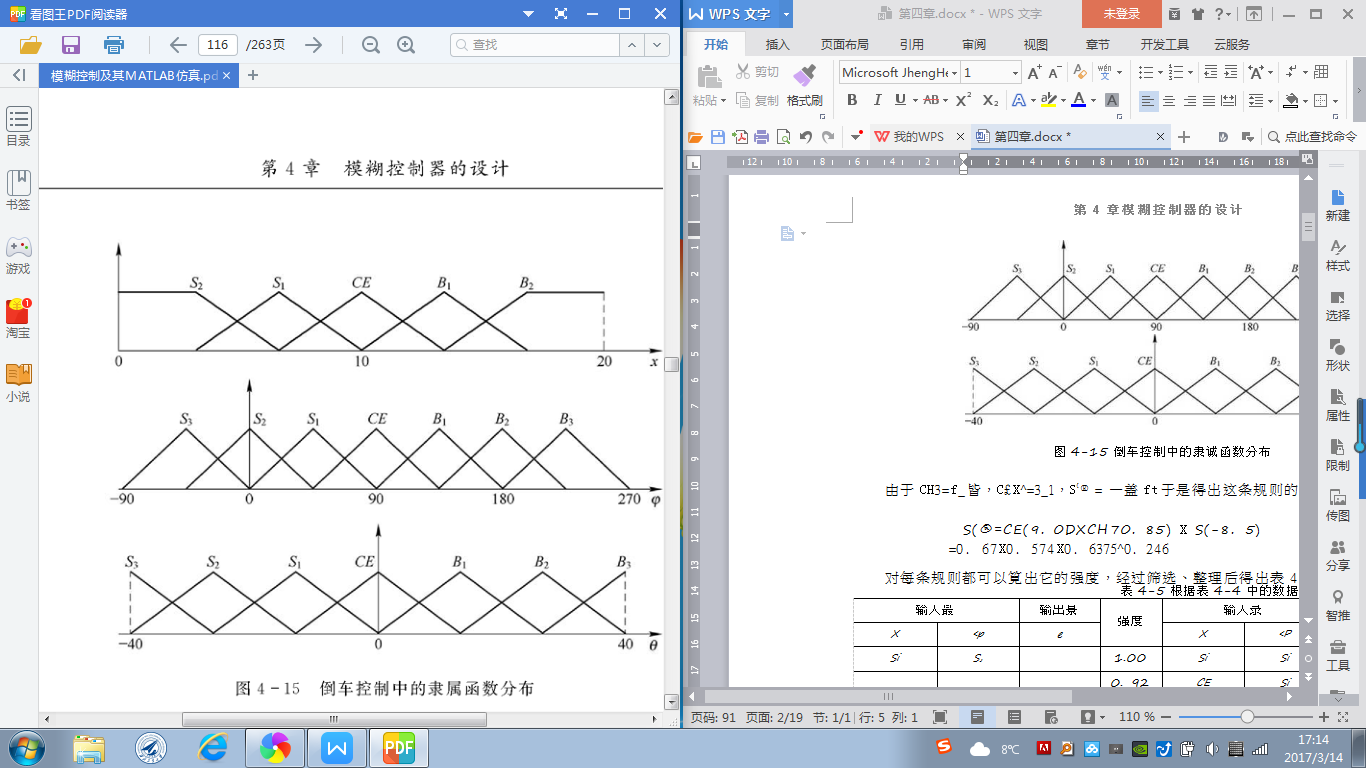


图 4-15 倒车控制中的隶属函数分布

由于，于是得出这条规则的强度为:



对每条规则都可以算出它的强度，经过筛选、整理后得出表4-5。

表4-5根据表4-4中的数据组产生的F控制语句及其强度

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 输入量 | | 输出量 | 强度 | 输入量 | | 输出量 | 强度 |
|  |  |  |  |  |  |
| S2 | S2 | S2 | 1.00 | S1 | S1 | S1 | 0.60 |
| S2 | S2 | S2 | 0.92 | CE | S1 | CE | 0.16 |
| S2 | S2 | S2 | 0.35 | CE | CE | CE | 0.32 |
| S2 | S2 | S2 | 0.12 | CE | CE | CE | 0.45 |
| S2 | S2 | S2 | 0.07 | CE | S1 | S1 | 0.35 |
| S1 | S2 | S1 | 0.08 | CE | S1 | S1 | 0.21 |
| S1 | S1 | S1 | 0.18 | CE | CE | CE | 0.54 |
| S1 | S1 | S1 | 0.53 | CE | CE | CE | 0.88 |
| S1 | S1 | S1 | 0.56 | CE | CE | CE | 0.92 |

根据上述数据表，可以得出六条模糊规则：“”、“”、“”、“”、“”、“”。综合更多的试验数据，可以总结出有35条在“倒车入位模糊规则表”，如表4-6所示，实际只有27条。表中加“”者为刚才得出的，抛弃了其中的第5条。

表 4-6 倒车入位模糊规则表

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  | CE |  |  |  |
|  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |
| CE |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |

把这张模糊控制规则表嵌入模糊控制器中，在MATLAB中进行仿真，可以得出图4-16所示的一个倒车入位轨迹示意图。

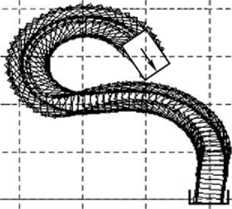


图 4-16 采用F控制仿真的倒车轨迹示意图

这里只介绍了一种根据输入-输出数据组设计F控制系统的方法，它是一种简单启发式方法，其中隶属函数是预先确定的，并没有根据输入—输出数据组进行优化。还有一些根据输入—输出数据组设计F控制系统的方法，比如梯度下降法、递推最小二乘法、聚类法等，可以根据输入—输出数据组对F控制系统中的部分参数进行优化、修正调整，诸如优选隶属函数参数、控制器结构、控制规则数目等。对此这里不作介绍，有兴趣的读者可查阅书后列出的“参考文献[14]”。不过现在更多的是利用神经网络方法，根据数据自动生成模糊规则。

**3.公式型模糊规则**

语言型和表格型模糊规则一旦存入模糊控制器，很难进行实时修改，这给环境和条件经常变化的实际生产将带来许多不便。于是有人提出用公式表述模糊规则，这种方法称公式型模糊规则表述法。设已知某双输入—单输出模糊规则如表4 -7所示。

表4-7 —个双输入-单输出F规则基础表

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| EC  U  E | NB | NS | ZE | PS | PB |
| NB | NB | NB | NS | NS | ZE |
| NS | NB | NS | NS | ZE | PS |
| ZE | NS | NS | ZE | PS | PS |
| PS | NS | ZE | PS | PS | PB |
| PB | ZE | PS | PS | PB | PB |

为了便于对模糊规则进行实时修改，把表中的“模糊子集”换成“模糊数”，模糊数也是模糊子集，所以这种替换可以看成模糊子集名称的变更。例如，把表4-7中的F子集“NB、NS、ZE、PS和PF”分别换成模糊数“-2、-1、0、1和2”，于是可以得出表4- 8。特别注意，表4-8中的“数值”并非是“整数”，而是模糊数，它们的隶属函数可以自己定义

表 4-8 一个双输入-单输出模糊数规则表

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| EC  U  E | -2 | -1 | 0 | 1 | 2 |
| -2 | -2 | -2 | - 1 | - 1 | 0 |
| -1 | -2 | -1 | - 1 | 0 | 1 |
| 0 | -1 | -1 | 0 | 1 | 1 |
| 1 | -1 | 0 | 1 | 1 | 2 |
| 2 | 0 | 1 | 1 | 2 | 2 |

仔细分析可以看出，表4 -8完全可以用下面的公式代替：



公式中的U、E和EC都代表某个“模糊数”； < >为取整运算符，表示将其中数值的绝对值四舍五入取整，正负号与< > 中数值的符号相同。比如，，则；若，则。 当赋予模糊数E和EC不同的数值时，由公式就可以算出对应的U。如，取E=-1，取EC 分别为-2、-1、0、1、2，代入公式就可以得出表芯第二行的内容：-2、-1、-1、0 和1。

为了扩大上述公式的使用范围，进行如下的变换：



再用代替0.5便可写成。把的取值由0.5扩充为0到1间的任何实数值，则变成一个普遍公式：



称为“修正因子”，这个公式被称为“F规则公式”。当赋予一个确定值时，就得出一张 F规则表。如取=0.2和=0.7，就分别得出表4-9和表4-10。

表4-9 取=0.2时，有得出的模糊数规则表

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| EC  U  E | -2 | -1 | 0 | 1 | 2 |
| -2 | -2 | -1 | 0 | 0 | 1 |
| -1 | -2 | -1 | 0 | 1 | 1 |
| 0 | -2 | -1 | 0 | 1 | 2 |
| 1 | -1 | -1 | 0 | 1 | 2 |
| 2 | -1 | 0 | 0 | 1 | 2 |

表4-10取=0.7时，由得出的模糊数规则表

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| EC  U  E | -2 | -1 | 0 | 1 | 2 |
| -2 | -2 | -2 | -1 | -1 | -1 |
| -1 | -1 | -1 | -1 | 0 | 0 |
| 0 | -1 | 0 | 0 | 0 | 1 |
| 1 | 0 | 0 | 1 | 1 | 1 |
| 2 | 1 | 1 | 1 | 2 | 2 |

由F规则公式可以看出，当修正因子=0.5时，偏差E和偏差变化率EC对输出U的权重相同，都为0.5。一旦偏离了0.5，则它们对输出U的影响就不再相等：>0.5时，偏差E的权重大于偏差变化率的权重，<0.5时，则正好相反。

通过仿真方法可以研究修正因子对系统性能的影响。一般说来，减小修正因子时，偏差变化率的权重增大，系统响应速度变慢；增大时系统的上升时间变短、超调量变大、调整时间变长。使用时，根据被控对象的特点及控制指标，选取合适的修正因子数值。

使用公式法，只需变更修正因子的大小，就可以改变整个控制器的模糊规则，给实际工作带来许多方便。有些系统要求在不同的工作状态下，偏差和偏差变化率的权重应该有所不同。比如偏差大时，控制器的主要任务是快速消除偏差影响，这时应取较大的值，以便加强偏差的权重，提高系统的响应速度。反之，偏差较小时，控制器的任务是使系统尽快趋于稳定，应取较小的修正因子值，以便增大偏差变化率的权重，提高系统的稳定性。可见，利用这些修正因子的变化，就能改善系统的动态特性。

为了使调整方便灵活，可引入多个修正因子。下面列出带两个和四个修正因子的模糊规则公式，以供参考。涵盖偏差论域的模糊子集用模糊数0、和表示，其中均为0到1中间的实数，通常取。





公式型二维F控制器的基本结构，如图4-17所示。它与表格型F控制器结构框图基本一样，只是把表格型F控制器中的“F控制表”模块换成了这里的F控制公式模块“”，特别需要注意，这里的偏差E和偏差变化率EC都是模糊数。

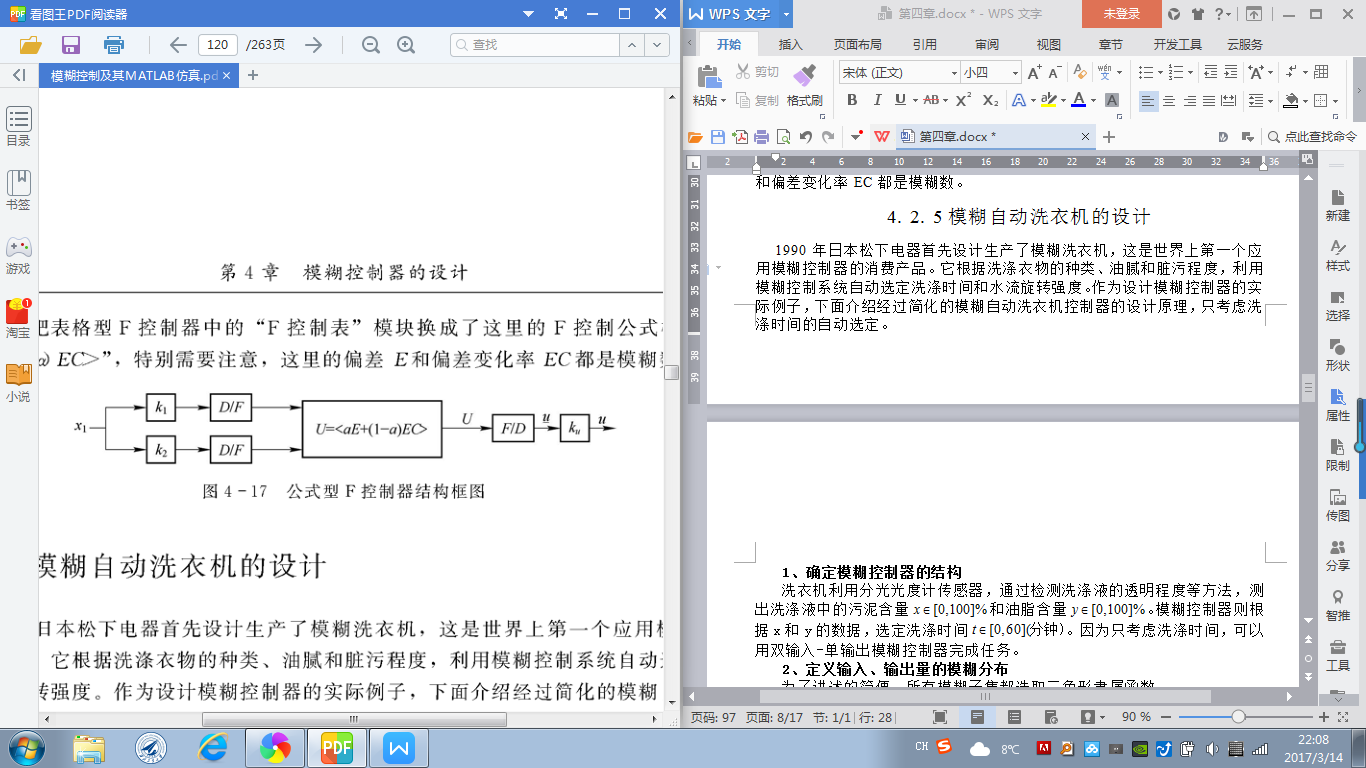


图 4-17 公式型F控制器结构图