CyberPhysicalSystems

一、Finine State Automata

static: 没有memory, 没有状态变量

deterministic: 同样的输入只有确定的下一个状态

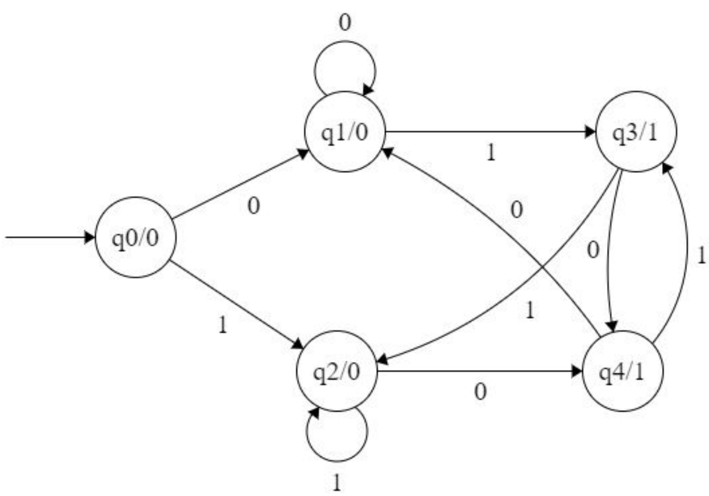
输入事件，输出事件：$IE_i^{(n)}, OE_i^{(n)}$， n为大类，i为子类。

unique 输入输出事件：单独事件的组合，因为状态机只允许一个状态出现。unique输入事件要利用 $IE_*^{(n)}$ 设定优先级。

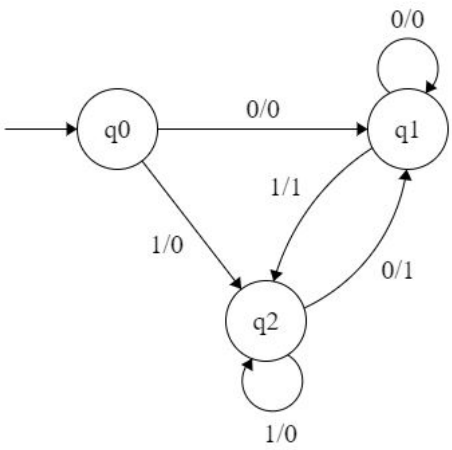
$$\begin{aligned} \tilde{u}_1 &= \{IE_1^{(2)}, IE_*^{(1)}, IE_*^{(3)}, IE_*^{(4)}\} \\ \tilde{u}_2 &= \{IE_1^{(4)}, IE_*^{(1)}, IE_*^{(3)}\} \\ \tilde{u}_3 &= \{IE_2^{(4)}, IE_*^{(1)}, IE_*^{(3)}\} \\ \tilde{u}_4 &= \{IE_1^{(3)}, IE_*^{(1)}\} \\ &\vdots \\ \tilde{u}_7 &= \{IE_1^{(1)}\} \end{aligned}$$

*代表忽略，*越多则中断等级高。*越少的，优先级越低。unique输出事件直接组合OE即可。 $\tilde{y}_1 = \{OE_1^{(1)}, OE_1^{(2)}, OE_1^{(3)}\}$

Moore型与Mealy型状态机：Moore型输出只取决于当前状态，输出写在状态结点上;



Mealy型取决于当前状态与输入，输出写在输入Transition上。



不管是Moore型还是Mealy型，遇到没有定义的Transition时一般回到自己 (Self transition) 并输出空事件ϵ

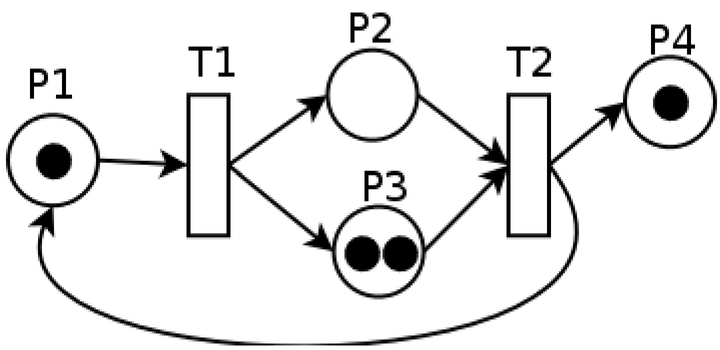
这样就可以写出State transition Table

$z(k)/y(k)$	$\tilde{u}_1$	$\tilde{u}_2$	$\tilde{u}_3$	$\tilde{u}_4$	$\tilde{u}_5$	$\tilde{u}_6$	$\tilde{u}_7$
$z_1/\tilde{y}_1$	z_2	z_1	z_1	z_1	z_5	z_1	z_4
$z_2/\tilde{y}_3$	z_3	z_3	z_2	z_2	z_2	z_2	z_2
$z_3/\tilde{y}_2$	z_3	z_3	z_1	z_3	z_3	z_3	z_3
$z_4/\tilde{y}_2$	z_4	z_4	z_4	z_1	z_4	z_4	z_4
$z_5/\tilde{y}_4$	z_5	z_5	z_5	z_5	z_5	z_2	z_5

二、Pretri-Net

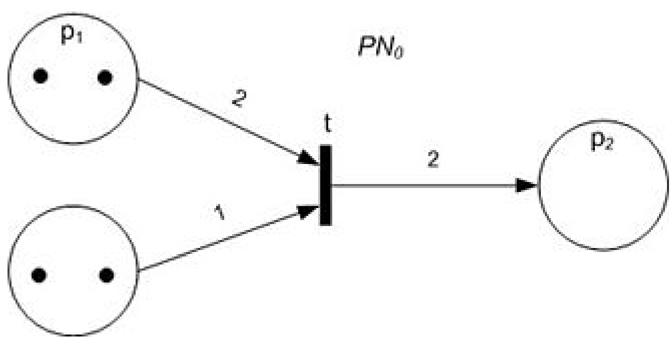
适用于cocurrent事件，即多个状态同时发生且需要同步。

> 数学概念：FSA是Normal Graph，只有一种结点可以随便连。Petri-Net是Bipartite Graph，有两种结点、有方向。



这两种结点是Place与Transition。$\mathcal{PRE}: P \to T, \mathcal{POST}: T \to P$

Token 令牌： 当且仅当一个T的所有Pre-place令牌都到位时才会下一步，由此实现同步，即wait for each other



Transition激活称为Fire，每次Fire后k+1时刻p位置令牌数量m为：

$$m(p,k+1)=\begin{cases}\text{不变} & ,\text{和}T_1\text{无关}\\ -1 & ,\text{在}T_1\text{上家}\\ +1 & ,\text{在}T_1\text{下家}\\ \text{不变} & ,\text{自循环}\end{cases}$$

向量化表示所有Transition就要用到incidence matrix，

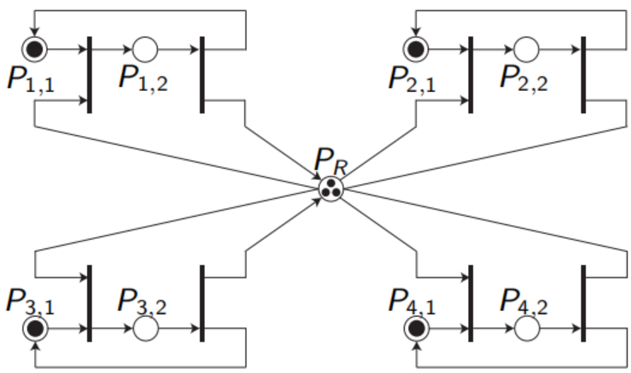
$$N=\left.\begin{bmatrix}-1 & 0 & \\ 1 & 0 & \\ 1 & -1 & \cdots \\ 0 & -1 & \\ 0 & 1 & \end{bmatrix}\right\} places$$

$$\text{初状态}\, m_0+N\cdot\delta_0=\text{次状态}$$

$$\begin{bmatrix}1\\0\\0\\1\\0\end{bmatrix}+N\cdot\begin{bmatrix}1\\0\\0\end{bmatrix}=\begin{bmatrix}0\\1\\1\\1\\0\end{bmatrix}$$

存在三种Petri-Net：

- Place/Transition: 每个Place上可以有任意个令牌
- Condition/Event： 每个Place上有且仅有一个令牌
- Signal-Interpreted：Condition/Event加上输入输出

> P/E Petri-Net用法：共用资源
>
> 
>

Non-Deterministic：

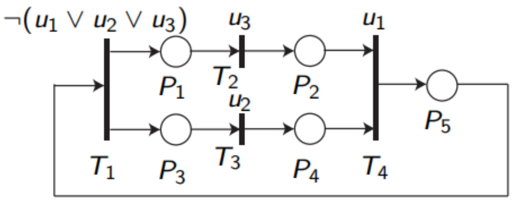
- 一P对多T：任何网络都可能遇到，不知道接下来会触发哪个T
- 多T对一P：C/E网络中，由于一P最多就一个令牌，多个T也不能同时发射

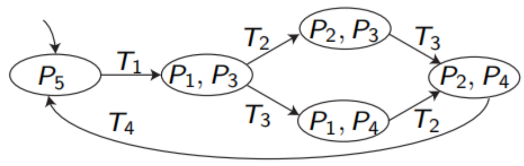
Synchronization Graph：严格按照$T\rightarrow P\rightarrow T$的单一连接顺序

Signal-Interpreted Petri-Net

在C/E Petri-Net的基础上：

- Trainsition 加上输入开关u
- Place 加上输出y





PN

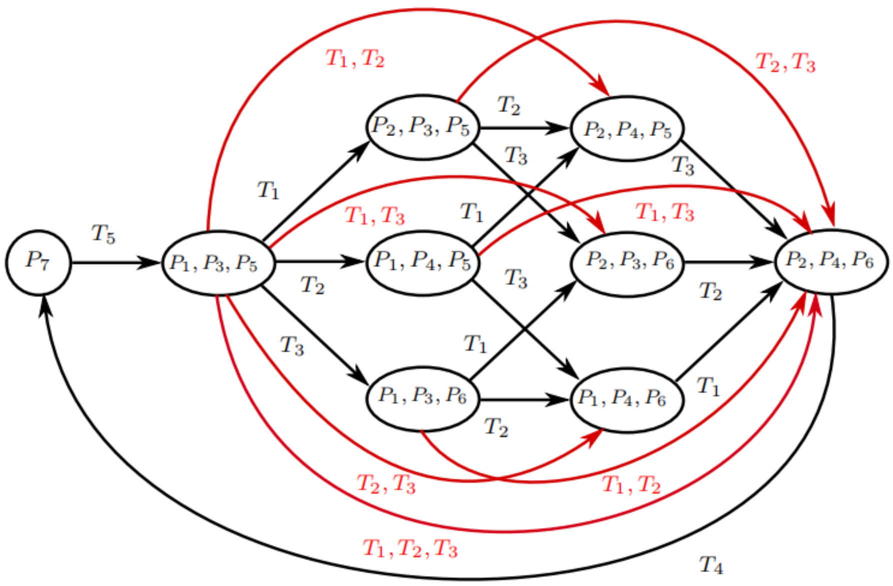
比起之前两种，又多了两条规则：

1. maximum step: 所有已满足条件（输入开关+令牌）的Transition均可发射
2. Iterative Firing：连环消消乐，遇到没有开关或者开关已满足的的Transition直接跳过，这些Transition永远不会被触发

SIPN有如下指标：

Reachable Set $\epsilon(m_0)$：从一个初态开始，所有可能的系统状态集合。

Reachability Graph：类似FSA，见下图。黑线是普通Petri-Net，一次只能触发一个Transition；红线是SI-Petri-Net，在前者的基础上，还可以同时激发多个Transition。



Conflict-Free Outputs：多个Place指向同一个输出时，输出不同（一个出0，一个出1）就会产生冲突。

	0	1	−	c
0	0	c	0	c
1	·	1	1	c
−	·	·	−	c
c	·	·	·	c

Stability：由于IterativeFiring，如果一个输入会使得一个循环中所有开关都激活，就会导致无限循环。

检验方法：该循环中所有FireCondition并集为0。（比如$\neg u_2 \wedge (u_2 \wedge u_3) \implies 0$）

Liveness：所有结点均可到达，所有Transition均有用

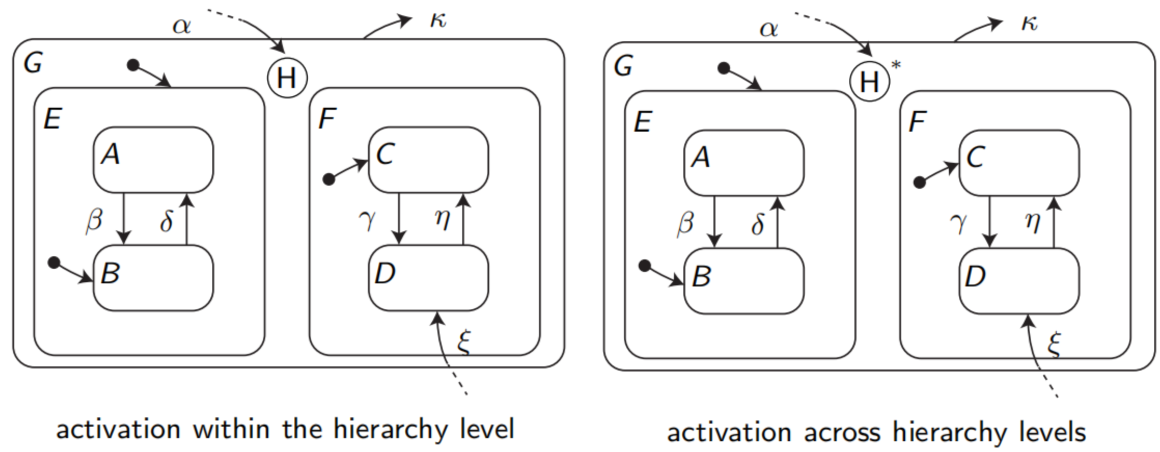
Deadlock：卡住或部分卡住

Well-Defined：Stable+Conflict-Free+Complete-Output+Deterministic Firing，此时可以与FSA互相转化。

三、Statecharts

有层级关系的FSA，主要特征是Superstate。比如紧急避险状态可能包含不少子状态，就不需要为它们一个个设置状态。

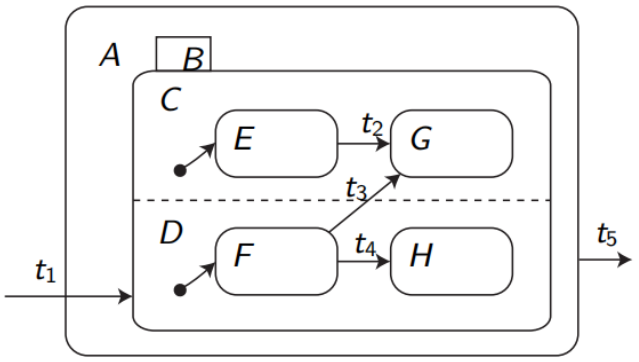
初始化：即使有Superstate存在，初状态仍要具体到某些子状态（Basic State）



每个层级都可以有一个H结点，用于保存退出时在这一层的结点信息。若没有则这一层存档丢失，要从默认结点开始。

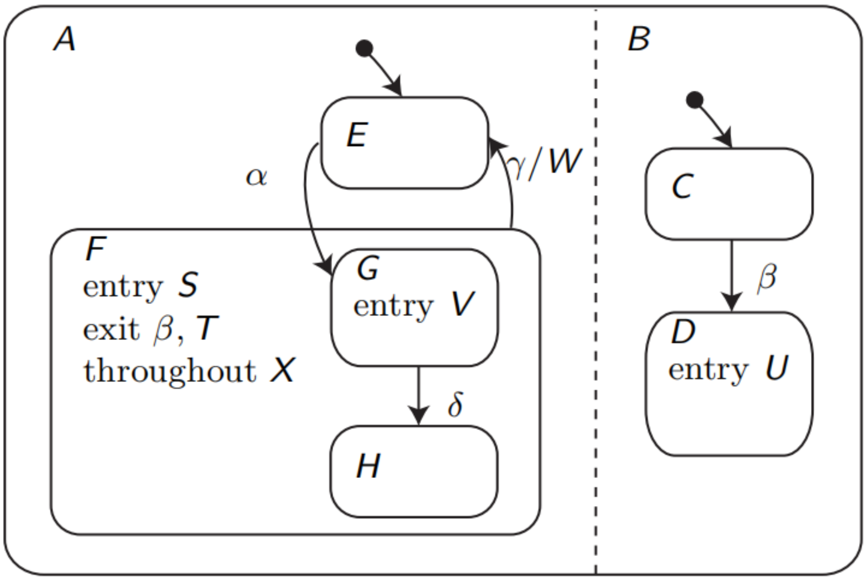
XOR-State：这个状态中能有一个子状态激活

AND-State：同步状态用虚线分隔，被同步状态绑定的两个子状态激活联动。



active basic states: E, F
active transition: t_3
scope: A
exited states: B, C, D, E, F
entered states: B, C, D, F, G

1. StateCharts 可以在状态转移（类似Moore型）或收到输入（类似Mealy型）或保持状态时有输出。分别对应Entry, Exit, Throughout
2. 外部输入信号（希腊字母α, β等）、内部的状态信号（$[in(A)], [NOT\ in(B)]$）、以及内部输出信号（大写拉丁字母ABC）都可以激发状态转移
3. 如果是内部信号激发的转移，则不消耗时间（$Still\ t_1$）
4. FSA输出信号只限于某个时间点，Petri-Net输出信号可以是连续时间，而Statecharts开关信号也可以连续。



	$\mathbf{t_0}$	$\mathbf{t_1}$	$\mathbf{t_2}$	$\mathbf{t_3}$	**still** $\mathbf{t_3}$
external event	–	α	δ	γ	–
generated event	–	S, V	–	T, W, β	U
active basic states	(E, C)	(G, C)	(H, C)	(E, C)	(E, D)
$X(t),\ t \in [t_k, t_{k+1}]$	inactive	active	active	inactive	inactive

Scope： 一条Transition的Scope为囊括它起点与终点的**最低一层XOR-Fashion**

当输入信号相同的两条Transition冲突时，执行Scope大的那条

四、Discrete Control

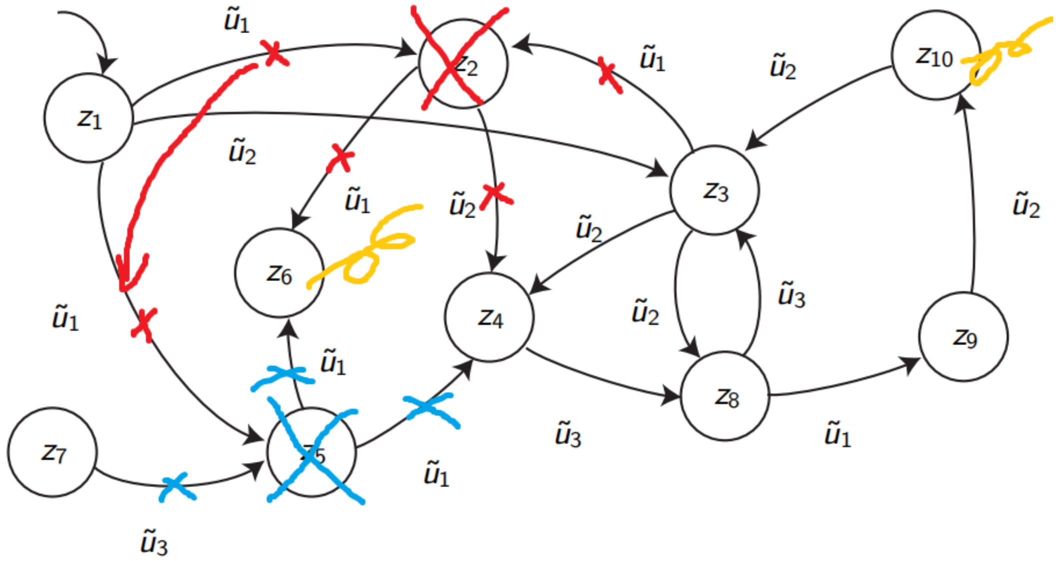
控制器Controller $\Longleftrightarrow$ 工厂Plant

Controller For Non-deterministic FSA

前提是工厂的状态图已知

- 工厂彻底禁止某些事件 Z_f ，连带和它们相连的所有Transition
- 对某些状态，只禁止某些输入 $\Omega_f = \{(Z_x, U_x), \cdots\}$
- 最后要到达目标事件 Z_g goal

Plant model:

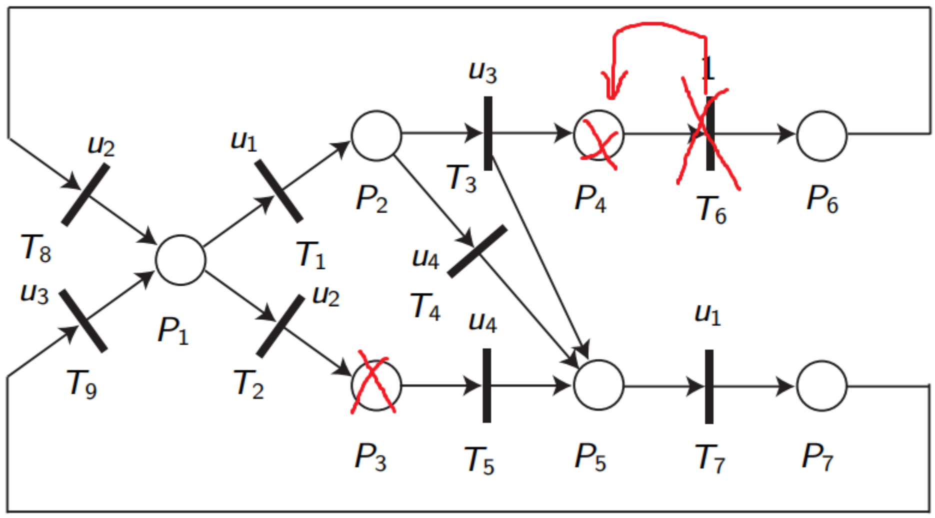


1. 禁止状态：Z_2
2. 删除Transition：只删除z_2的所有输入输出还不够。由于Non-deterministic，为了完全杜绝进入Z_2的可能性，必须ban掉所有z_1出发的$\tilde{u}_1$输入线，但只能禁止状态、没法禁止输入，因此z_5不幸也顺带被禁
3. 最后找一条起点至终点最短路径 $Z_1 \to Z_3 \to Z_8 \to Z_9$ 或$Z_1 \to Z_3 \to Z_4 \to Z_8 \to Z_9$ (Z_3输入$\tilde{u}_2$时非Deterministic)

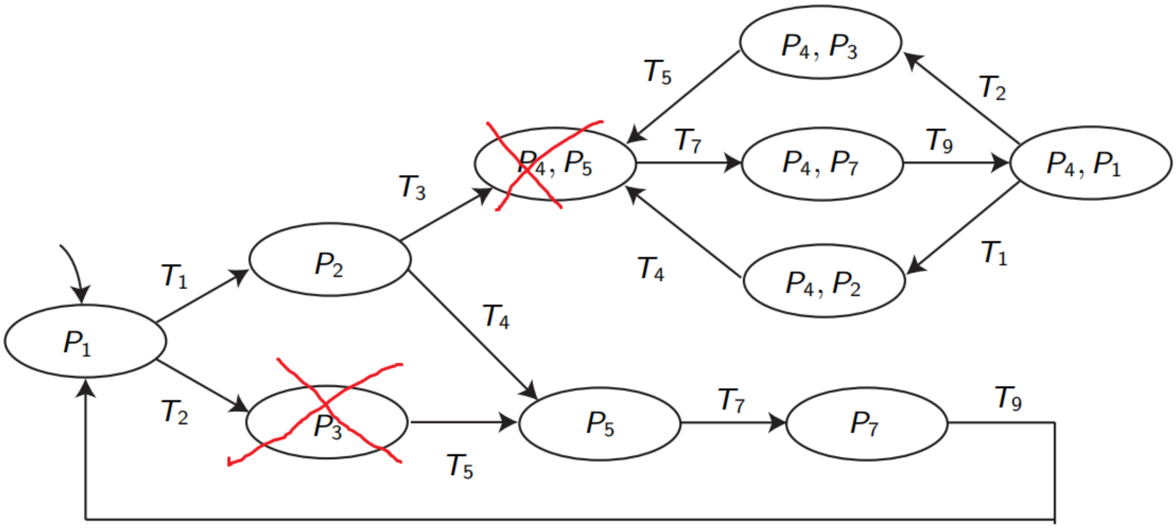
Controller For Petri-Net

- 禁止某些状态组合 Marking$\mathcal{M}_f$
- 禁止某些转移$\mathcal{T}_f$, 若不可控Transition（开关永远为1）被禁，还必须删除其Pre-Place
- 最后要到达某个 Marking $\mathbf{m}_g$

Plant model:



1. 题设$\mathcal{T}_f = \{T_6\}.\mathcal{M}_f = [* * 1 * * * *]$
2. $\mathcal{M}_f = [* * 1 * * * *] \cup [* * *1 * **]$
3. 转化为Reachability Graph



五、Discrete System Verification

六、连续系统建模与仿真

非线性ODE解析解 Seperation of variables

$\frac{d\nu}{d\xi} = \tilde{g}(\xi)\tilde{f}(\nu) \implies \int \frac{1}{\tilde{f}(\nu)} d\nu = \int \tilde{g}(\xi) d\xi$

> e.g. $\dot{h} = -k\sqrt{h}$，此时$\nu = h; \xi = t$（时间）
>
> $\therefore \tilde{g}(\xi) = 1, \tilde{g}(\nu) = -k\sqrt{\nu} = -k\nu^{0.5}$
>
> $\implies \int -\frac{1}{k}\nu^{-0.5} d\nu = \int 1 \cdot d\xi \implies -\frac{2}{k}\nu^{0.5} + C_1 = \xi + C_2$
>
> $\therefore \nu = \left[-\frac{k}{2}(\xi + C)\right]^2$
>
> 求初值：代入$\xi = 0, ;v_0 = (-\frac{k}{2}C)^2 = h_0$

线性时不变系统LTI 存在解析解

齐次解 $x^{(h)}(t) = e^{a(t-t_0)}x_0$

特解 $x^{(p)}(t) = \int_{t_0}^{t} e^{a(t-\tau)}u(\tau)d\tau$

由于是线性系统，解析解就是两个解叠加

一维情况下：$x(t) = x^{(h)}(t) + x^{(p)}(t)$

数字解法

- 欧拉法 $\dot{x}(t) = f[x(t)]$，$x(t_{k+1}) = x(t_k) + \underbrace{(t_{k+1} - t_k)}_{\text{步长}h}\underbrace{f(x(t_k))}_{\text{梯度}}$
- Heun Method ${\color{red}x(t_{k+1})} = x(t_k) + (t_{k+1} - t_k) \cdot \frac{1}{2}[f(x(t_k)) + f({\color{red}x(t_{k+1})})]$

为了得到$x(t_{k+1})$需要它本身的信息，这是不可能的，所以等式右边用欧拉法代替

$K_1 = f(x(t_k)), K_2 = f[x(t_k) + hK_1]$

$x(t_{k+1}) = x(t_k) + h \cdot \frac{1}{2}(K_1 + K_2)$

- Runge-Kutta $x(t_{k+1}) = x(t_k) + h \cdot \frac{1}{6}(K_1 + 2K_2 + 2K_3 + K_4)$

$$
\begin{aligned}
K_1 &= f[x(t_k)] \\
K_2 &= f[x(t_k) + \tfrac{1}{2}hK_1] \\
K_3 &= f[x(t_k) + \tfrac{1}{2}hK_2] \\
K_4 &= f[x(t_k) + hK_3]
\end{aligned}
$$

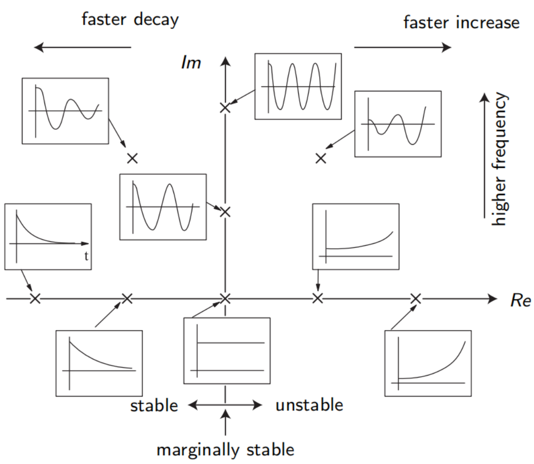
这三种方法的误差呈指数级下降

$\left|\frac{x^e(t_k+h)-x(t_k+h)}{h}\right| \leq C \cdot h^p$， 指数P越大，误差越小

欧拉p=1，Heun p=2，Runge p=4

七、分析连续系统

稳定条件：$\dot{x}(t) = A \cdot x(t)$，A所有特征值均$Re < 0$



$$
\begin{aligned}
\overrightarrow{\dot{x}(t)} &= \mathbf{A}\vec{x}(t) + \mathbf{B}\vec{u}(t) \\
\vec{y}(t) &= \mathbf{C}\vec{x}(t) + \mathbf{D}\vec{u}(t)
\end{aligned}
$$

Controllability： $R = \begin{pmatrix} A & AB & A^2B & \cdots & A^{n-1}B \end{pmatrix}$ 若满秩则该系统完全可控（从任意状态到任意状态）

Observability：$O = \begin{pmatrix} C^T & A^TC^T & (A^T)^2C^T & \cdots & (A^T)^{n-1}C^T \end{pmatrix}$, 若满秩说明只看输入输出就可知状态。（单输出只要检察$det(O) \neq 0$即可）

拉氏变化：

$$
\begin{aligned}
sX(s) = AX(s) + BU(s) &\implies X(s) = (sI-A)^{-1}BU(s) \\
Y(s) = CX(s) + DU(s) &\implies \underbrace{\left[C(sI-A)^{-1}B + D\right]U(s)}_{G(s)}
\end{aligned}
$$

传递函数为$G(s) = C(sI-A)^{-1}B + D$，一维情况下$G(s) = \frac{cb}{s-a} + d$

初值定理与终值定理：

$$
\begin{aligned}
\lim_{t\to 0} f(t) &= \lim_{s\to\infty} sF(s) \\
\lim_{t\to\infty} f(t) &= \lim_{s\to 0} sF(s)
\end{aligned}
$$

非线性系统稳定的条件

Lyapunov function $\dot{V}(x) \leq 0$

仍用线性系统举例：MSD系统，设$s_0 = 0, u = 0$

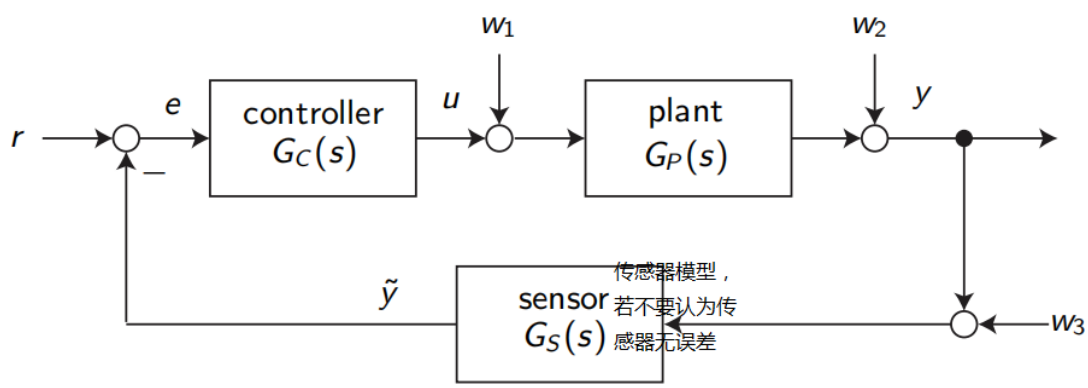
$$\dot{x}_1 = x_2$$
$$\dot{x}_2 = \frac{1}{m}(-cx_1 - d(x_2))$$

V 一般选取能量，此例中，动能$T = \frac{1}{2}mx_2^2$, 弹簧势能$P = \frac{1}{2}cx_1^2$

$V = T + P = \frac{1}{2}mx_2^2 + \frac{1}{2}cx_1^2$

$$\begin{aligned}\dot{V} &= \frac{\partial V}{\partial x} \cdot \frac{dx}{dt} \\ &= \begin{pmatrix} cx_1 & mx_2 \end{pmatrix} \begin{pmatrix} \dot{x}_1 \\ \dot{x}_2 \end{pmatrix} \\ &= cx_1x_2 + mx_2\frac{1}{m}(-cx_1 - dx_2) \\ &= -dx_2^2 < 0\end{aligned}$$

八、 单输入单输出系统



$$G_{YR} = \frac{G_C \cdot G_P}{\underbrace{1+G_CG_PG_S}_{G_0}} = \frac{G_CG_P}{1+G_0}$$
$$G_{YW_1} = \frac{G_P}{1+G_0}$$
$$G_{YW_2} = \frac{1}{1+G_0}$$
$$G_{YW_3} = \frac{-G_0}{1+G_0}$$

PID 控制器

$U(s) = (K_p + sK_d + \frac{K_I}{s})E(s)$

PD控制器可以稳定系统，但存在稳态误差
$\lim_{t\to\infty} y(t) - r(t) = \lim_{s\to 0} s \cdot (Y(s) - R(s)) = \lim_{s\to 0} s \cdot (G_{YR} - 1)R(s)$

可以引入K_I来消除稳态误差：

$G(s) = \frac{K(1+b_1s+\cdots+b_ws^w)}{s^m(1+a_1s+\cdots+a_us^u)}$，如果本身有一个极点就不需要$K_I$，只有分母上没有S的0态系统需要额外加一个S在分母上。

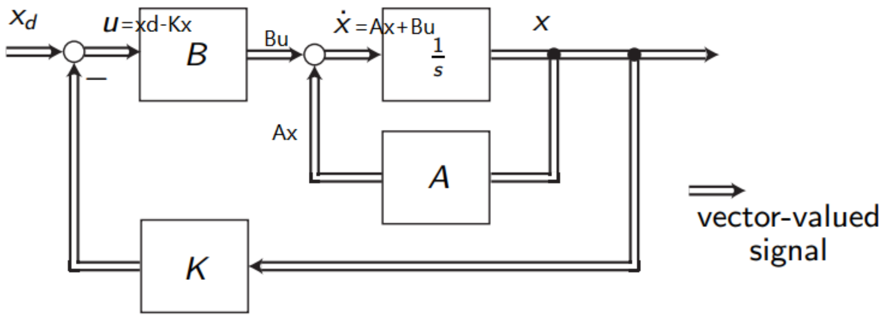
root-locus?

九、多输入多输出系统

State feedback control

适用于：有多个输入输出且状态变量之间有关联

$u = x_d - K \cdot x$



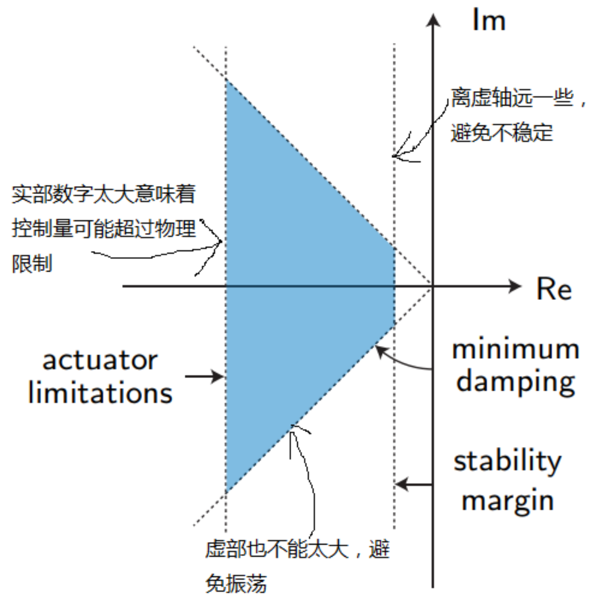
限制：状态量$\vec{x}$必须直接知道，不能再用传感器测量值

$$\begin{aligned}\dot{x} &= Ax + Bu \\ &= Ax + B(x_d - Kx) \\ &= (A - BK)x + B \cdot x_d \\ &= A_c x + u_c\end{aligned}$$

当系统完全可控时，可用这种方法得到任意响应（任意极点）的系统。

线性二次优化

实际上由于物理限制不可能任意取极点，



要求最优极点是一个二次项最优方程：$J = \int_0^\infty \underbrace{x^TQx}_{状态偏离惩罚} + \underbrace{u^TRu}_{能量消耗惩罚}\ dt$，设$x_d = 0$，所以$x$和$u$都是越小越好

> 要使这个二次项积分有上限，二次项的终值必须为0 → x，u终值为0

（证明略）最优解为$K = R^{-1}B^TP_+$，其中P_+为方程$PA + A^TP - PBR^{-1}B^TP + Q = 0$的**正定，对称**解

$$P_+ = \begin{bmatrix} p_{11} & p_{12} \\ p_{21} & p_{22} \end{bmatrix}$$

对称：$p_{12} = p_{21}$

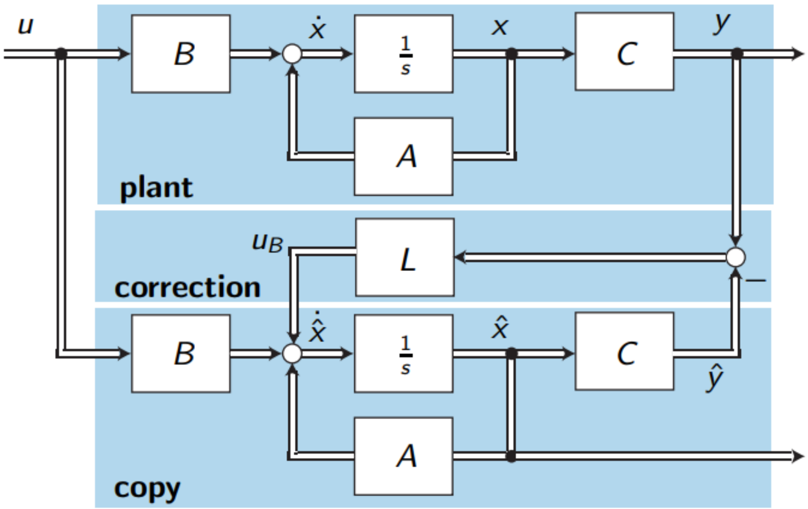
Sylvester`s criterion: $p_{11} = P_{+小矩阵} > 0, p_{11}p_{22} - p_{12}^2 = \det(P_+) > 0$

一般$Q = qI, R = I$，Q是必须的，R不是必须的

Q也可以是$\begin{pmatrix} \underbrace{10}_{惩罚重} & & \\ & \underbrace{1}_{惩罚轻} & \\ & & 10 \end{pmatrix}\begin{pmatrix} p \\ v \\ a \end{pmatrix}$，一般就设对角线元素就可以了

Luenberger Observer

一般情况下状态量还是隐藏的，只有测量值$y = Cx + Du$。要得到状态值可以用软件来模拟真实的物理系统，而软件中所有状态都是已知的。→用软件系统里的状态作为真实系统的状态返回值



$$
\begin{aligned}
u_B &= L(y - \hat{y}) \\
&= L[(Cx + Du) - (C\hat{x} + Du)] \\
&= LC(x - \hat{x}) \\
\dot{\hat{x}} &= A\hat{x} + Bu + LC(x - \hat{x}) \\
\text{误差}\dot{\tilde{x}} &= \dot{x} - \dot{\hat{x}} \\
&= (Ax + Bu) - [A\hat{x} + Bu + LC(x - \hat{x})] \\
&= A(x - \hat{x}) - LC(x - \hat{x}) \\
&= \underbrace{(A - LC)}_{A_0}\underbrace{(x - \hat{x})}_{\tilde{x}}
\end{aligned}
$$

即可以任意选定一个L，使得$A_0 = A - LC$观测状态量和实际值之间没有误差

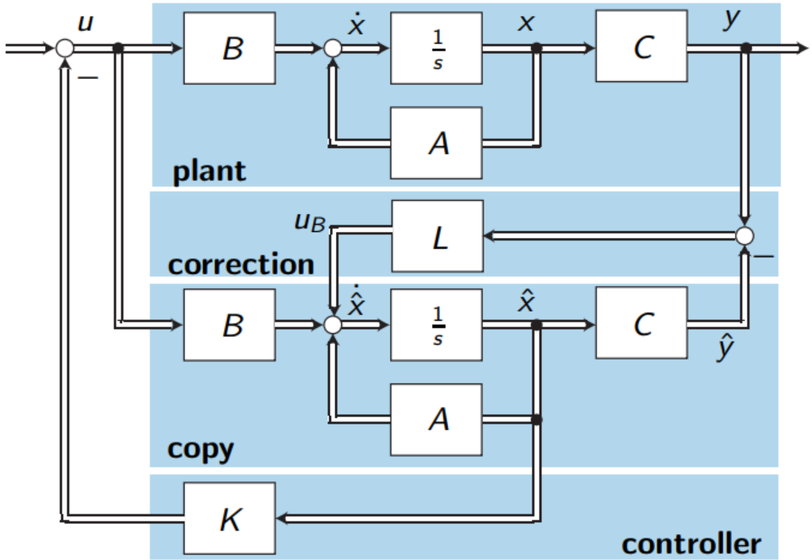
这也是一个优化问题：$J(\tilde{x}) = E(||\tilde{x}||)^2 = \lim_{T\to\infty} \frac{1}{2T}\int_{-T}^{T} ||\tilde{x}(t)||^2 dt$

$$
\begin{array}{c|l}
\dot{x} = Ax + Bu + v & process\ error \\
y = Cx + Du + w & \text{传感器误差}
\end{array}
$$

（证明略）最优解为$L = \Pi_+ C^T W$

其中，Π_+为等式$-A\Pi - \Pi A^T + \Pi C^T W^{-1} C\Pi = V$的**正定**解

合并任意控制与任意观测



$$
\begin{aligned}
\dot{x} &= Ax + B\overbrace{(x_d - K\hat{x})}^{\mathbf{u}} \\
&= Ax + Bx_d - BK\underbrace{\hat{x}}_{x-\tilde{x}} \\
&= (A - BK)x + Bx_d + BK\tilde{x}
\end{aligned}
$$

$\therefore [\dot{x}] = [\,A - BK \quad BK\,]\begin{bmatrix} x \\ \tilde{x} \end{bmatrix} + [B]x_d$ ①

再加上观测矩阵的公式：$\dot{\tilde{x}} = (A - LC)\tilde{x} \implies [\dot{\tilde{x}}] = [\,0 \quad A - LC\,]\begin{bmatrix} x \\ \tilde{x} \end{bmatrix}$②

合并①②式可得

$$
\begin{bmatrix} \dot{x} \\ \dot{\tilde{x}} \end{bmatrix} = \underbrace{\begin{bmatrix} \overbrace{A - BK}^{A_c} & BK \\ 0 & \underbrace{A - LC}_{A_0} \end{bmatrix}}_{A_{co}} \cdot \begin{bmatrix} x \\ \tilde{x} \end{bmatrix} + \begin{bmatrix} B \\ 0 \end{bmatrix} x_d
$$

计算A_{co}的特征值就是将A_c与A_o的特征值合并成一个集合

$\det(\lambda I - A_{co}) = \det[\lambda I - (A - BK)] \times \det[\lambda I - (A - LC)] = 0$