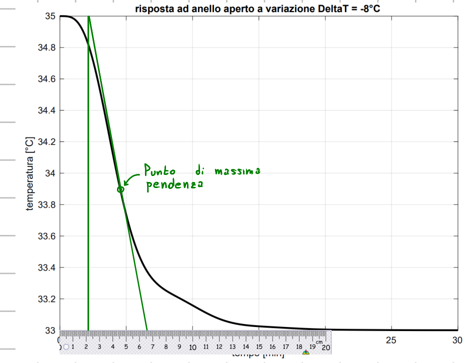
Es 1) Si puo' applicare il 1° metodo Z-N:



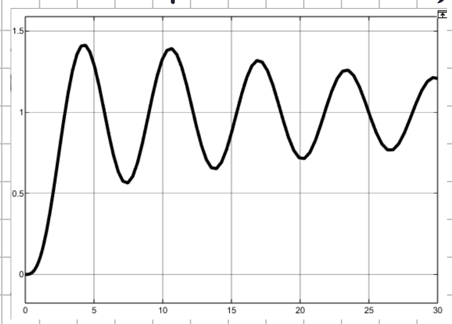
$K \cdot \Delta U = -2 \Rightarrow K \cdot (-8) = -2 \Rightarrow \frac{1}{4}$ Scelgo un regolatore PID

$K_p = 4 \cdot 1{,}2 \cdot \left(\frac{2.1}{4.5}\right)^{-1} = 10{,}27$

$T_i = 2 \cdot 2.1 = 4.2 \Rightarrow K_i = 2.44$ $\theta = 2.1$ $\tau = 4.5$

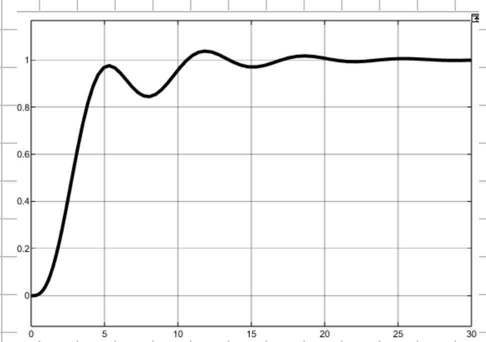
$T_d = \frac{1}{2} \cdot 2.1 = 1.05 \Rightarrow K_d = 10.78$

Con i parametri scelti, la risposta indiciale e' la seguente:



La risposta e' terribile, ed il sistema non e' lontano dall'essere instabile.

Provo a scalare di un fattore $\frac{1}{2}$ i parametri ottenuti col metodo Z-N:

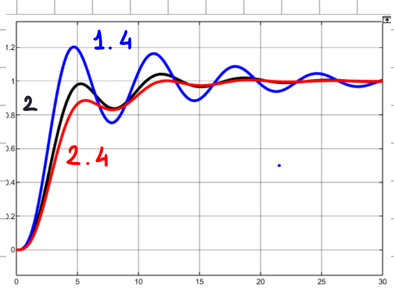


Con i seguenti guadagni

$K_p = 4.89$ $K_I = 1.16$ $K_D = 5.13$ $N = 20$ (filtraggio der.)

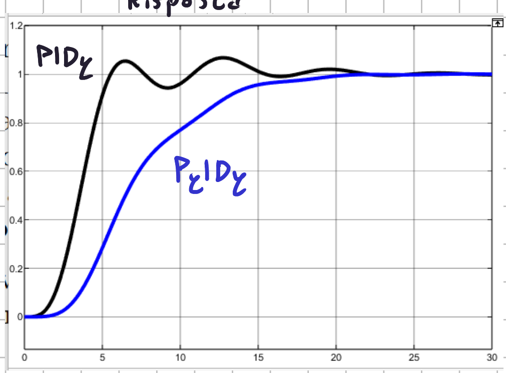
La risposta e' decisamente migliore, la sovraelongazione e' quasi nulla, ed il tempo di salita e' di circa 3/4 secondi con oscillazioni limitate.

Per completezza, sono riportate alcune varianti con fattori di scala 2.4 e 1.4

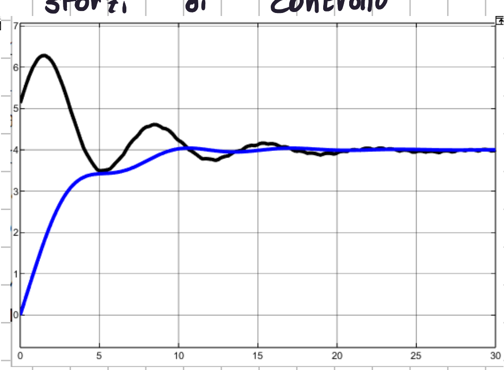


Essendo lo sforzo di controllo troppo alto, si considerano le azioni PD o P solo sull'uscita:

Risposta



sforzi di controllo

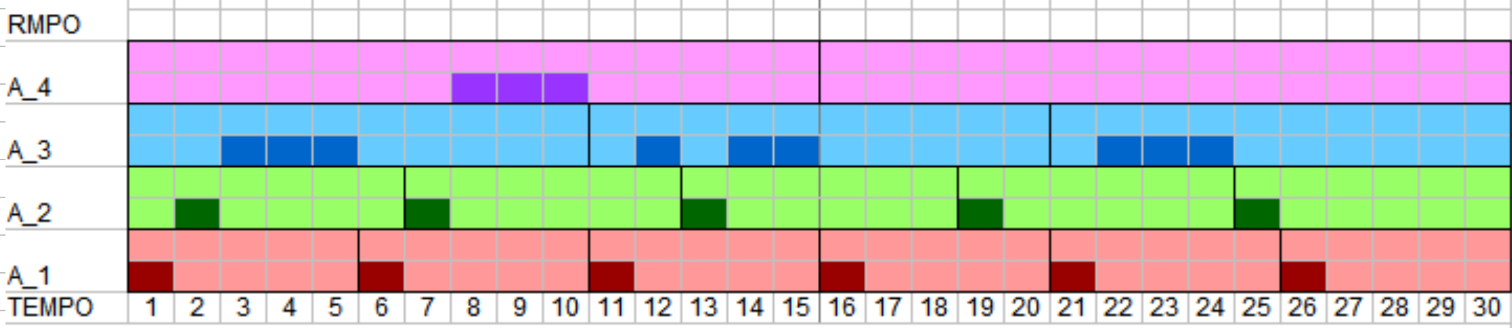


In definitiva, lo sforzo di controllo del regolatore PIDy è accettabile, nonostante quello del PyIDy sia nettamente inferiore, è conveniente il primo regolatore in quanto ha un tempo di salita molto più rapido, nel caso il sistema dovesse avere dei limiti di saturazione stringenti, sarebbe più appropriato il regolatore PyIDy.

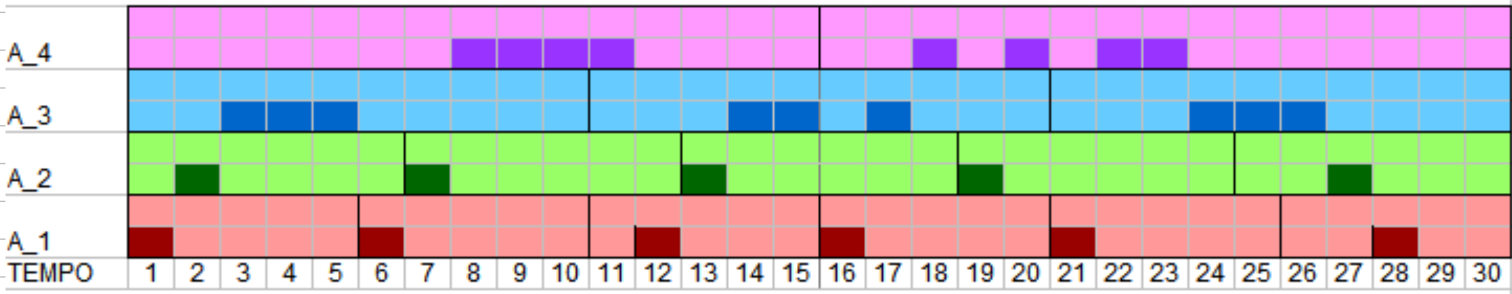
Es 2) Task periodici equivalenti:

	T_i	C_i
A_1	5	1
A_2	6	1
A_3	10	3
A_4	15	4

$$\Rightarrow U = \frac{8+9+5+6}{30} = \frac{28}{30} \begin{cases} U > U_{lsn} \\ U > \ln(2) \\ \text{non ci sono rel. armoniche} \end{cases}$$

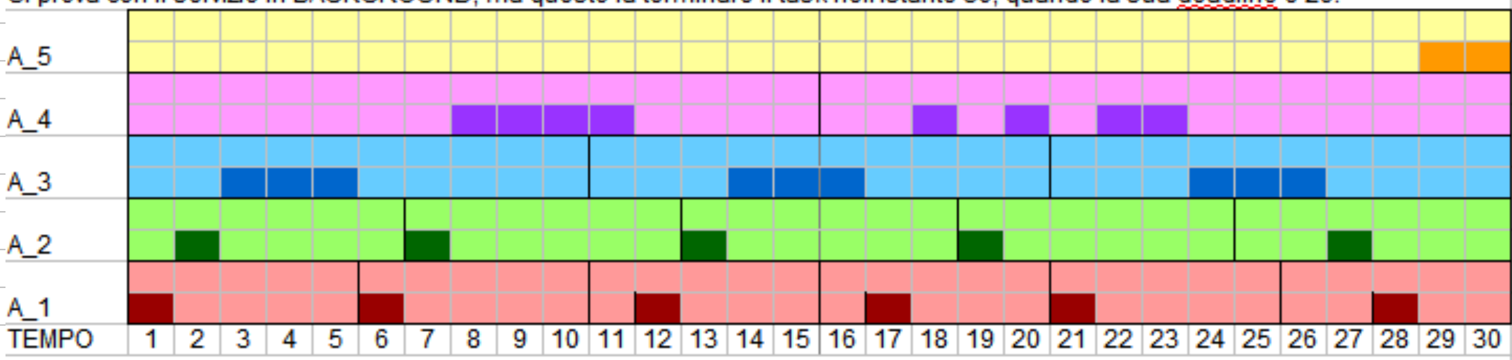


RMPO non è soluzione, si prova con EDF

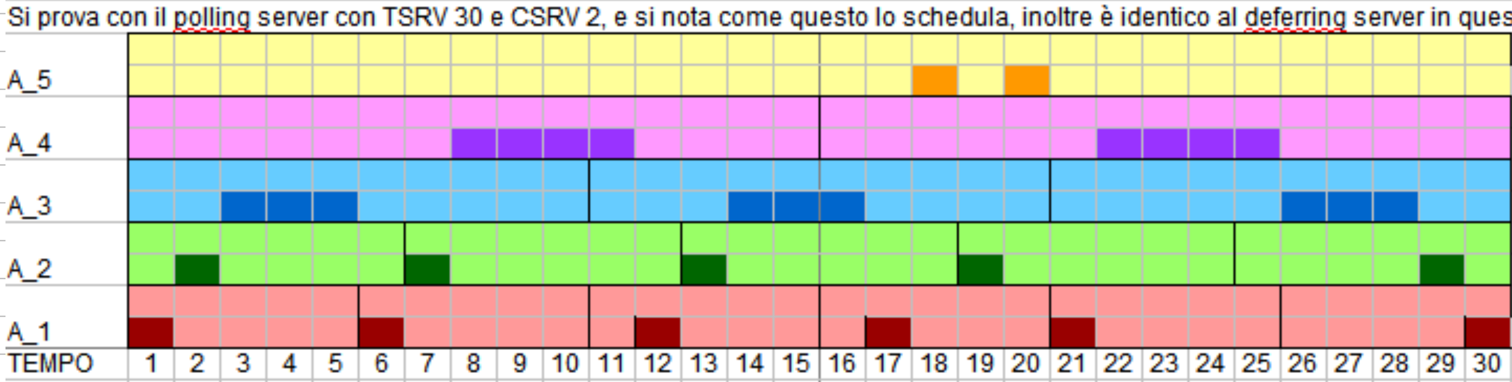


Ovviamente EDF è soluzione in quanto risolve qualsiasi insieme di task periodici con U<=1

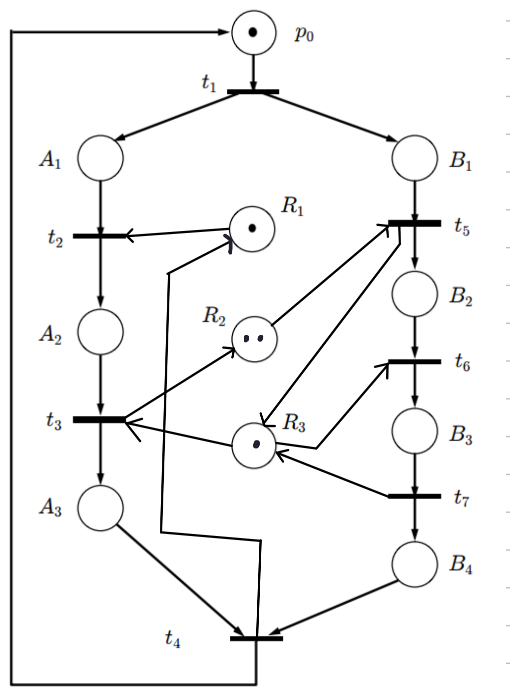
Si prova con il servizio in BACKGROUND, ma questo fa terminare il task nell'istante 30, quando la sua deadline è 29.



Si prova con il polling server con TSRV 30 e CSRV 2, e si nota come questo lo schedula, inoltre è identico al deferring server in questo



Es 3) e' necessario che $x_0(R_1) = x_0(R_3) = 1$ e $x_0(R_2) = 2$ per rendere la rete non bloccante. La matrice di incidenza e':



$$C = \begin{bmatrix} -1 & 0 & 0 & 1 & 0 & 0 & 0 \\ 1 & -1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & -1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & -1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & -1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & -1 \\ 0 & 0 & 0 & -1 & 0 & 0 & 1 \\ 0 & -1 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & -1 & 0 & 0 \\ 0 & 0 & -1 & 0 & 1 & -1 & 1 \end{bmatrix}$$

$Ker(C) = \{[x, x, x, x, x, x, x]^T ; x \in \mathbb{R}\}$

Il T-invariante canonico e' $\eta = [1\ 1\ 1\ 1\ 1\ 1\ 1]^T$.

Una sequenza di scatti può far tornare la rete allo stato iniziale se ogni transizione scatta lo stesso numero di volte.

$Ker(C^T) =$ $\{(a+b-c, a-c+d-e, a+d-e, a, b-d+e, b, b+e, b, c, d, e) : a, b, c, d, e \in \mathbb{R}\}$

Ci sono 5 P-invarianti canonici, il supporto ricopre P, la rete e' limitata.

$\gamma_1 = [1\ 1\ 1\ 1\ 0\ 0\ 0\ 0\ 0\ 0\ 0]$ $a = 1$

$\gamma_2 = [1\ 0\ 0\ 0\ 1\ 1\ 1\ 1\ 0\ 0\ 0]$ $b = 1$

$\gamma_3 = [0\ 0\ 1\ 1\ 0\ 0\ 0\ 0\ 1\ 0\ 0]$ $a = c = 1$

$\gamma_4 = [1\ 1\ 1\ 0\ 0\ 0\ 0\ 0\ 0\ 1\ 0]$ $b = d = 1$

$\gamma_5 = [1\ 0\ 0\ 1\ 1\ 0\ 1\ 0\ 0\ 0\ 1]$ $a = e = 1$