

UNIVERSITÀ DEGLI STUDI DI ROMA TOR VERGATA

FACOLTÀ DI INGEGNERIA

CORSO DI LAUREA IN INGEGNERIA INFORMATICA

A.A. 2002-2003

Tesi di Laurea Triennale

Controllo di sistemi di produzione
con tempi di funzionamento e di guasto
non esponenziali

RELATORE

Ing. Francesco Martinelli

CANDIDATO

Marco Ciccotelli

A mio fratello Flavio

Ringraziamenti

Un sincero ringraziamento va all'Ing. Francesco Martinelli che mi ha seguito in questo lavoro con attenzione, disponibilità e cortesia.

Ringrazio i miei Genitori e mia sorella Alessia che con il loro affetto mi sono vicini ogni giorno e con i loro sacrifici permettono che la mia carriera universitaria prosegua senza preoccupazioni.

Meritano un grazie tutto particolare gli Amici più cari: Alessio, Matteo, Ramin, Emanuele, Fabrizio e Lorenzo, i miei “colleghi” Danilo e Donatello, che costituiscono sempre un punto di riferimento importante e con cui ho condiviso ogni piccola esperienza, soprattutto la più importante, il *Cammino di Santiago de Compostela*, in questo anno fonte inesauribile di determinazione e crescita personale.

INDICE

<i>INTRODUZIONE</i>	6
Struttura della tesi	7
<i>CAPITOLO 1 - DEFINIZIONE DEL PROBLEMA</i>	8
1.1 Controllo di sistemi di produzione	8
1.2 Tempo di funzionamento e di guasto con probabilità esponenziali	10
1.3 Tempo di funzionamento deterministico e tempo di guasto generale	15
<i>CAPITOLO 2 - TEMPO DI FUNZIONAMENTO E DI GUASTO DETERMINISTICI</i>	20
2.1 Analisi generale	20
2.2 Studio della funzione di costo $J(z)$	24
2.2.1 Stato 1	24
2.2.2 Stato 2	28
2.2.3 Stato 3	33
2.2.4 Studio della continuità e della derivabilità della $J(z)$	36
2.3 Grafico di $J(z)$	38
2.4 Simulazioni algoritmiche di sistemi D/D/1	40
Esempio 2.1	41
Esempio 2.2	43
Esempio 2.3	44
Esempio 2.4	46
Esempio 2.5	48
<i>CAPITOLO 3 - TEMPO DI FUNZIONAMENTO UNIFORME E TEMPO DI GUASTO DETERMINISTICO</i>	51
3.1 Analisi generale	51
3.2 Possibili andamenti di $J(z)$ nel tempo di funzionamento	52
3.3 Simulazioni algoritmiche di sistemi U/D/1	56
Esempio 3.1	57
Esempio 3.2	61
Esempio 3.3	64

3.4 Osservazioni sugli esempi	66
CONCLUSIONI E SVILUPPI FUTURI	67
BIBLIOGRAFIA	68
APPENDICE	69
Algoritmo per sistemi D/D/1	69
Algoritmo per sistemi U/D/1 con t^* fissato	73
Algoritmo per sistemi U/D/1 con t^* variabile	78

INTRODUZIONE

Il campo di ricerca in materia di automazione manifatturiera è in continuo sviluppo e tra i vari approfondimenti si propone di trovare nuove soluzioni per la determinazione di politiche ottime applicabili ai sistemi produttivi. Le caratteristiche di un sistema comprendono il tipo di flusso (continuo/discreto), il numero di macchinari e la quantità di pezzi diversi producibili, nonché le situazioni di funzionamento cicliche che determinano lo stato dell'impianto. Un'applicazione particolare sono i sistemi up-down in cui i cicli che si susseguono nel tempo sono semplicemente di funzionamento e guasto.

Uno dei principali settori nello studio di sistemi manifatturieri è l'analisi a partire dalla durata di questi cicli, con tempi di funzionamento e di guasto che abbiano densità di probabilità generiche. Nonostante numerose semplificazioni cui questi sistemi vengono sottoposti, la più importante delle quali è costituita da sistemi a macchina singola e con flusso continuo di produzione di un solo tipo di "parte", tale problema non è attualmente completamente esplorato e per molte scelte di tempi di distribuzione non è stato possibile determinare una politica ottima.

E' stata studiata finora approfonditamente la politica "hedging point" per sistemi con tempi di funzionamento e di guasto caratterizzati da densità di probabilità esponenziali. Nel campo di ricerca si lavora molto su questa situazione e su densità di probabilità deterministiche, mentre si trovano raramente analisi su altre situazioni, come possono essere quelle di tempi di funzionamento con probabilità gaussiana, normale o uniforme.

L'obiettivo di questo lavoro è di approfondire analiticamente esempi più comuni e, se possibile, intraprendere un nuovo argomento di studio. A questo scopo è stata necessaria una ricerca nella letteratura specializzata riguardante l'analisi di sistemi di produzione che ha delineato un quadro esaustivo dello studio attuale della materia.

Come notazione gli impianti di produzione up-down si indicano con tre caratteri: i primi due indicano la distribuzione di probabilità dei tempi di funzionamento e guasto, il terzo il numero dei server che la macchina utilizza (es. D/E/1 = tempo di funzionamento deterministico/tempo di guasto esponenziale/1 server).

Con questo lavoro si analizzeranno sistemi di produzione D/D/1, cioè con entrambi i tempi considerati deterministici, e si simuleranno sistemi di produzione U/D/1, in cui il tempo di funzionamento avrà una densità di probabilità uniforme. Per queste due tipologie di impianto si proporranno euristiche e risultati simulativi per trovare un valore di scorta ottima a partire dai parametri del sistema (tasso di produzione massimo, domanda costante, tempo di funzionamento, tempo di guasto, costo di scorta positiva e negativa).

Importante risultato a cui perverrà tale lavoro nel caso di sistemi D/D/1 è lo studio di una classe di politiche e della funzione di costo corrispondente fino alla dimostrazione dell'esistenza ed unicità di un valore di scorta ottimo in tale classe di politiche. Si proporranno in questo ambito delle semplificazioni del problema e delle analisi qualitative della funzione di costo.

Per sistemi U/D/1, invece, si proporrà un'euristica per la determinazione della scorta ottima a partire dalla simulazione del sistema caratterizzato da parametri costanti di tasso di produzione e costo. Verranno utilizzati algoritmi implementati in Matlab 6.1 e in C++ per il calcolo del costo al variare della scelta di scorta ottima e del parametro temporale che caratterizza tale euristica. I risultati mostrati sono puramente simulativi e si proporranno in merito alcune considerazioni di carattere intuitivo.

Struttura della tesi

Nel capitolo 1 verrà presentato un richiamo delle teorie di sistemi di produzione con tempi di setup non trascurabili e con densità di probabilità per i tempi di funzionamento e guasto generali. Ci sarà la descrizione del problema analizzato per sistemi D/G/1 e una relazione sulla ricerca in rete degli studi recenti in materia.

Il capitolo 2 sarà dedicato ai sistemi D/D/1, con la proposizione di un'euristica e l'analisi della funzione di costo relativa ad essa. Si studierà la ricerca del minimo di tale funzione e se ne dimostrerà l'esistenza e l'unicità. Infine si mostrerà la perfetta analogia di tali risultati con quelli derivanti da simulazioni algoritmiche di sistemi esemplificativi.

Nel capitolo 3 si simuleranno sistemi U/D/1 con l'euristica presentata nel capitolo 2 e con una classe di politiche più generali cercando di ricavare a livello intuitivo considerazioni di carattere generale per la determinazione della scorta ottima.

Nell'appendice verranno inseriti gli algoritmi di calcolo della funzione di costo utilizzati per i risultati mostrati nel capitolo 2 e 3.

CAPITOLO 1 - Definizione del problema

1.1 Controllo di sistemi di produzione

Il problema del controllo dei sistemi di produzione, l'argomento approfondito e sviluppato in questa trattazione, consiste nell'analisi dello stato attuale e a regime del sistema, nella determinazione di una politica di produzione in base a tale stato e nello studio del numero ottimo di pezzi da lavorare in ogni macchina per minimizzare la determinata funzione di costo.

Il problema dello scheduling, cioè la pianificazione della produzione, è perciò affrontato da un punto di vista dinamico, studiando il sistema a catena chiusa e determinando in base allo stato attuale il tasso di produzione corrispondente.

Si lavorerà nel caso di macchine soggette a guasti per sistemi del tipo "pull", in cui i buffer sono posti a valle della macchina; il caso generale è limitato ad una singola macchina (vedi figura 1.1) che produce p tipi di parti, ognuna delle quali con un tasso di produzione massimo μ_i e un livello costante di domanda d_i , che genera un livello di buffer $x_i(t)$, con contenuto positivo o negativo:

- $x_i(t) > 0$: la macchina ha prodotto un surplus di beni pari a $x_i(t)$, pronti ad essere consegnati alla domanda in arrivo;
- $x_i(t) < 0$: c'è una quantità di domanda arretrata da soddisfare pari a $-x_i(t)$.

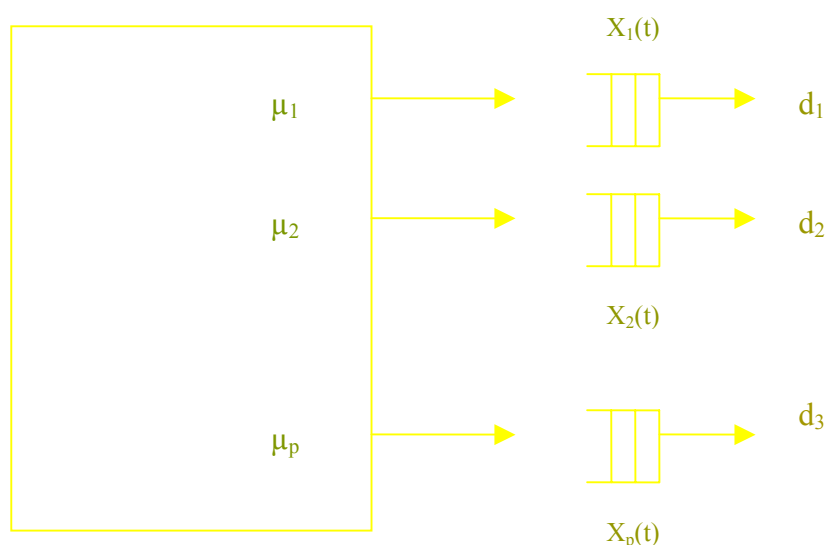


Figura 1.1 – Sistema di produzione a macchina singola

La prima semplificazione di questo tipo di sistemi che consideriamo è quella di un sistema che produca $P=1$ tipi di parti (vedi figura 1.2), considerando come significativi i risultati ottenuti ed estendibili al caso più generale di p parti.

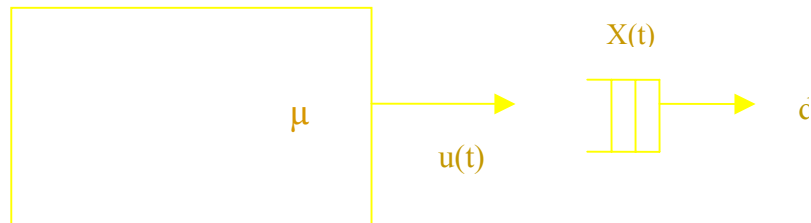


Figura 1.2 – Sistema di una singola macchina che produce un solo tipo di parti

Il tasso di produzione $u(t)$ è $[0, \mu]$ se la macchina è in funzione, invece $u(t)=0$ se la macchina è guasta.

Lo stato dinamico del buffer è

$$\frac{dx}{dt} = u(t) - d$$

mentre i tempi di funzionamento e di guasto, indicati con t_f e t_g , avranno una densità di probabilità di varia natura, la cui analisi sarà proprio l'argomento principale di questo lavoro.

Per ogni scelta di distribuzione di probabilità siano $\bar{t}_f$ e $\bar{t}_g$ i tempi medi di funzionamento e guasto.

La condizione di stabilità è

$$\frac{\bar{t}_f \mu}{\bar{t}_f + \bar{t}_g} > d \quad (1)$$

Nel caso non ci fossero guasti la macchina potrebbe portare e mantenere a 0 il contenuto $x(t)$ del buffer nell'ipotesi che $\mu > d$, ma ciò non è possibile a causa dei guasti.

Al sistema è possibile associare un funzionale di costo medio a regime del tipo:

$$J = \lim_{t \rightarrow \infty} \frac{1}{t} E \left[\int_0^t g(x(s)) ds \right] \quad (2)$$

in cui $g(x)$ è la funzione di costo e l'operatore E rappresenta la sua media statistica, che nel nostro caso sarà calcolabile in forma chiusa con notevoli semplificazioni algebriche e computazionali.

Lo scopo dello studio di questi sistemi consiste proprio nello sviluppare politiche ottime di produzione che minimizzino la funzione J .

1.2 Tempo di funzionamento e di guasto con probabilità esponenziali

Il caso in cui i tempi di funzionamento e guasto abbiano distribuzioni di probabilità di tipo esponenziale è quello più studiato e per questo tipo di sistemi si dimostra l'ottimalità della politica "hedging point" (definita più avanti).

I tempi di funzionamento e guasto sono distribuiti esponenzialmente se:

$$\text{Prob} \{ t_f \leq t \} = \int_0^t q_d e^{-q_d s} ds = 1 - e^{-q_d t} \quad (3a)$$

$$\text{Prob} \{ t_g \leq t \} = \int_0^t q_u e^{-q_u s} ds = 1 - e^{-q_u t} \quad (3b)$$

E' facile verificare che $\bar{t}_f = 1/q_d$ e $\bar{t}_g = 1/q_u$. In base a ciò la percentuale di tempo in cui la macchina è in servizio è data da $\frac{q_u}{q_d + q_u} \leq 1$. Pertanto, il tasso medio di produzione sarà $\frac{\mu q_u}{q_u + q_d}$ e la condizione di stabilità è data da:

$$\mu \frac{q_u}{q_d + q_u} > d \quad (4)$$

La (4) è quindi la condizione di stabilità di sistemi con probabilità di funzionamento e guasto esponenziali, che garantisce a regime un contenuto $x(t)$ finito per il buffer con il corrispondente margine di stabilità per sistemi E/E/1:

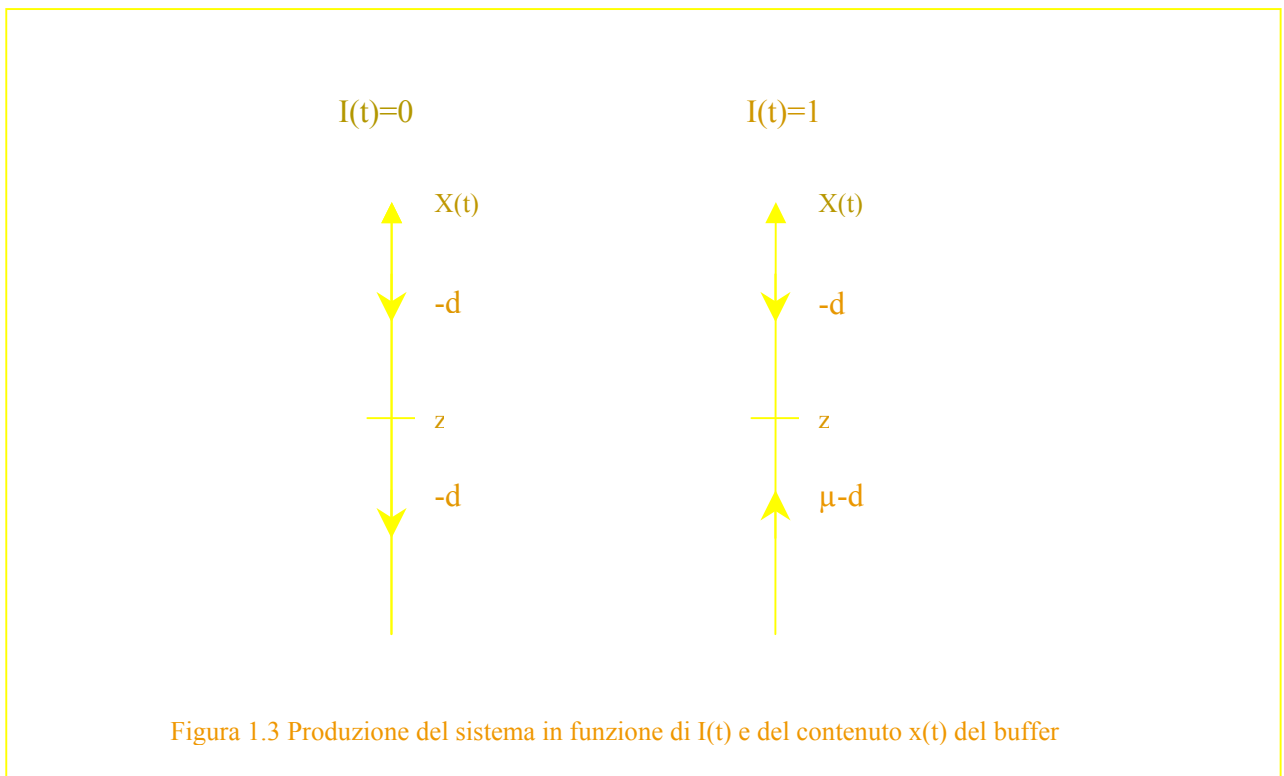
$$S := \mu q_u - d(q_d + q_u)$$

La politica ottima per questo tipo di sistemi, detta hedging point, è quella definita, considerando z il livello di scorta fissato, da:

$$u(t) = \begin{cases} 0 & x(t) > z \\ d & x(t) = z \\ \mu & x(t) < z \end{cases} \quad (5)$$

se la macchina è in funzione, altrimenti $u(t)=0$. L'ottimalità di tale politica è dimostrata per una funzione $g(x)$ di costo lineare a tratti, del tipo $g(x)=c_p x^+ + c_m x^-$. E' comunque possibile determinare il valore ottimo di scorta z anche per funzioni di costo convesse e positive come ad esempio $g(x)=cx^2$.

Se definiamo $I(t)$ come stato del sistema (che nel nostro caso può assumere due valori, 0 quando è in funzione e 1 se la macchina è guasta) possiamo verificare per la politica scelta che il sistema si avvicina a z come in figura 1.3:



Sviluppando i calcoli per la minimizzazione del funzionale J si ottiene che z minimizza J se e solo se

$$E\left[\frac{dg}{dx}\right] = 0 \quad (6)$$

Nel caso della $g(x)=c_p x^+ + c_m x^-$ da noi scelta, con $c_p, c_m > 0$, otteniamo

$$E\left[\frac{dg}{dx}\right] = -c_m P(x < 0) + c_p P(x \geq 0) = -(c_m + c_p)P(x < 0) + c_p$$

che combinata con la (6) diventa

$$P(x < 0) = \frac{c_p}{c_p + c_m} \quad (7)$$

Se consideriamo il caso in cui sia $z=0$,

$$P(x < 0) = 1 - \gamma \quad (8)$$

In generale, intendendo con γ la probabilità che a regime $x(t)=z$, dobbiamo distinguere due casi. Se

$$\frac{c_p}{c_p + c_m} < 1 - \gamma \quad \text{occorre uno } z > 0 \text{ che è calcolabile in forma chiusa:}$$

$$z^* = \frac{1}{\lambda} \ln \left(\frac{c_p + c_m}{c_p} (1 - \gamma) \right) \quad (9)$$

dove

$$\lambda = \frac{q_u}{d} - \frac{q_d}{\mu - d} \quad (10)$$

$$\gamma = \frac{\mu \cdot q_u - d(q_d + q_u)}{(\mu - d)(q_d + q_u)} \quad (11)$$

sicuramente positivi nel caso in cui sia soddisfatta la (4) e quindi il sistema sia stabile.

Se, invece, $\frac{c_p}{c_p + c_m} \geq 1 - \gamma$ allora $z^*=0$, cioè il sistema lavora con costi minimi senza scorte e si parla di politica “Just in Time” (JIT).

Punto di riferimento per gli studi in materia è l’analisi fatta nel 1988 da Bielecki e Kumar in [2] i cui risultati sono ripresi a livello didattico e come fonte per numerosi articoli tra cui [3] di S.P.Sethi, H.Yan e Q.Zhang.

Allo stesso risultato pervengono Y.Feng e H.Yan, in [4], in cui viene analizzato l’utilizzo della politica Hedging Point su sistemi analoghi a quello presentato precedentemente, con una funzione di costo

$$J^u(x, i) = E \left(\int_0^{\infty} e^{-\gamma t} [c^+ x_u^+(t) + c^- x_u^-(t)] dt \right) \quad (12)$$

$$i = 0, 1$$

la cui analisi avviene attraverso le equazioni HJB nel caso in cui non sia calcolabile un risultato in forma chiusa. Si dimostra che è ottima tale politica e si calcolano differenti soluzioni di z^* .

Si perviene al risultato di $z^*=0$ (caso JIT) se

$$\frac{\gamma(\gamma + d + q_0 + q_1)}{\mu(\gamma + q_0 + d)} \geq \frac{c^-}{c^+ + c^-}(y_1 - 1) \quad (13)$$

mentre nel caso in cui non fosse soddisfatta questa disuguaglianza z^* è il più piccolo degli interi che soddisfano

$$h(z) \leq -(y_1 - 1) \frac{c^+}{c^+ + c^-} \cdot \frac{\mu(\gamma + d + q_0)(y_2 - y_3)}{\gamma(\gamma + q_0 + q_1)} \quad (14)$$

dove

$$h(z) = \left(\frac{\gamma + d + q_0 + q_1}{\gamma + q_0 + q_1} - \frac{1}{1 - y_2} \right) y_2^{z+1} - \left(\frac{\gamma + d + q_0 + q_1}{\gamma + q_0 + q_1} - \frac{1}{1 - y_3} \right) y_3^{z+1} \quad (15)$$

considerando che y_1, y_2, y_3 sono parametri ricavati nell'analisi del sistema di equazioni differenziali nello studio della funzione di costo.

Lo stesso Y.Feng assieme a B.Xiao in [5], come anche J.R.Perkins e R.Srikant nell'articolo [6] analizzano il caso in cui si debba lavorare in condizioni di domanda non costante, del tipo

$$D(t) = \sum_{i=1}^{N(t)} d_i \quad (16)$$

con $N(t)$ variabile con distribuzione di Poisson con tasso λ e d_i parametri esponenziali di media $1/\lambda_d$.

Per questo sistema la stabilità è assicurata da

$$\frac{q_u}{q_u + q_d} \mu > \frac{\lambda}{\lambda_d} \quad (17)$$

Utilizzando la politica hedging point si calcola in forma chiusa nel caso in cui

$$\frac{c^-}{c^+ + c^-} > \frac{\mu \cdot q_u - d(q_u + q_d)}{(\mu - d)(q_u + q_d)} \quad (18)$$

che il valore $z^*>0$ è

$$z^* = \frac{1}{\lambda^{BK}} \ln \left[(1 - \gamma^{BK}) \left(1 + \frac{c^-}{c^+} \right) \right] \quad (19)$$

con

$$\lambda^{BK} = \frac{q_u}{d} - \frac{q_d}{\mu - d} \quad (20)$$

$$\gamma^{BK} = \frac{\mu \cdot q_u - d(q_u + q_d)}{(\mu - d)(q_u + q_d)} \quad (21)$$

il cui costo corrispondente è

$$J^* = c^+ \left[z^* + \frac{d}{q_d + q_u} \right] \quad (22)$$

mentre se la disuguaglianza (18) non è soddisfatta si rientra nel caso JIT con $z^*=0$.

J.Hu, P.Vakili e G.Yu dimostrano in [7] che condizione necessaria e sufficiente affinché sia ottima la politica hedging point è che la funzione di guasto sia funzione lineare del parametro di produzione u .

J.R.Perkins e R.Srikant allargano il problema a sistemi che producono due tipi di parti (figura 1.1 con $P=2$) in [8], caso poi generalizzato a n parti dagli stessi autori in [9] nel caso in cui sia

$$J = \lim_{t \rightarrow \infty} \frac{1}{T} E \left(\int_0^T \sum_{i=1}^n [c^+ x_u^+(t) + c^- x_u^-(t)] dt \right) \quad (23)$$

Per due soli tipi di parti si fissa come politica di produzione per i due buffer

$$u_1(t) = \begin{cases} 0 & f(x_1 - z_1, x_2 - z_2) > 0, (x_1, x_2) \neq (z_1, z_2) \\ \mu & f(x_1 - z_1, x_2 - z_2) < 0, (x_1, x_2) \neq (z_1, z_2) \\ \mu_1 & f(x_1 - z_1, x_2 - z_2) = 0, (x_1, x_2) \neq (z_1, z_2) \\ 0 & (x_1, x_2) = (z_1, z_2) \end{cases} \quad (24a)$$

$$u_2(t) = \begin{cases} \mu - u_1(t) & (x_1, x_2) \neq (z_1, z_2) \\ 0 & (x_1, x_2) = (z_1, z_2) \end{cases} \quad (24b)$$

dimostrando l'ottimalità della JIT sotto le condizioni:

$$\gamma_1 \geq \frac{c_1^-}{c_1^- + c_1^+} \quad (25a)$$

$$\gamma \geq \frac{c_2^-}{c_2^- + c_2^+} \quad (25b)$$

con

$$\gamma_1 = \frac{\lambda_1 d_1}{q_u + q_d} \text{ e } \gamma = \frac{\lambda(d_1 + d_2)}{q_u + q_d} \text{ (se } \lambda = \frac{q_d}{d_1 + d_2} - \frac{q_u}{\mu - (d_1 + d_2)} \text{)}$$

Non sono presenti dimostrazioni di una coppia di valori di hedging point (z^*_i) che minimizzino il costo nel caso non vengano soddisfatte le condizioni (25a) e (25b) mentre si dimostra l'esistenza di limiti superiori per i due buffer.

Anche M.Veatch e M.Caramanis in [10] propongono euristiche per il caso generale di macchina singola a n parti per poi raggiungere gli stessi risultati di Perkins-Srikant nel caso di 2 parti.

1.3 Tempo di funzionamento deterministico e tempo di guasto generale

E' quindi evidente che la letteratura nel caso di tempi di guasto e funzionamento con probabilità esponenziali è ricca di numerosi articoli, sia per la valenza di possibili applicazioni pratiche, sia per la semplicità di analisi rispetto a distribuzioni più complicate.

Un caso relativamente semplice è quello in cui il tempo di funzionamento sia fissato costante, cioè il caso di sistemi D/G/1. Tuttavia non sono stati realizzati studi teorici molto approfonditi su questo caso, se non quello del 1995 elaborato da J.Hu e D.Xiang: “*Optimal Control for Systems with Deterministic Production Cycles*”.

In questo lavoro i due studiosi analizzano a ciclo aperto il sistema D/G/1 proponendo una semplice euristica di controllo della scorta ottima da inseguire, non curandosi della distribuzione di probabilità dei tempi di guasto.

Si considera un sistema di produzione con una singola macchina che produce un solo tipo di parti, con una domanda costante d e due stati possibili α_t per la macchina, 0 se è guasta e 1 se è in funzione.

La dinamica del sistema è caratterizzata da:

$$\frac{dX_t}{dt} = u_t - d \quad (26)$$

dove $0 \leq u_t \leq \alpha_t \mu$.

Per ogni politica π si può definire una funzione di costo

$$J_{\pi} = \lim_{T \rightarrow \infty} \frac{1}{T} E \int_0^T (C^+ X_t^+ + C^- X_t^-) dt \quad (27)$$

se $X_t^+ = \max(X_t, 0)$ e $X_t^- = \max(-X_t, 0)$, e C^+, C^- costanti non negative.

L'obiettivo come nei casi precedenti è trovare una politica π^* che minimizzi la (27).

Per ottenere tale risultato si propone di mantenere una scorta non negativa alla fine del periodo di funzionamento D, al fine di bilanciare la perdita nel periodo di guasto susseguente.

La politica di produzione ottima è

$$u_t^* = \begin{cases} = \mu & X_t^* < 0, \alpha_t = 1 \\ \geq d & X_t^* = 0, \alpha_t = 1 \end{cases} \quad (28)$$

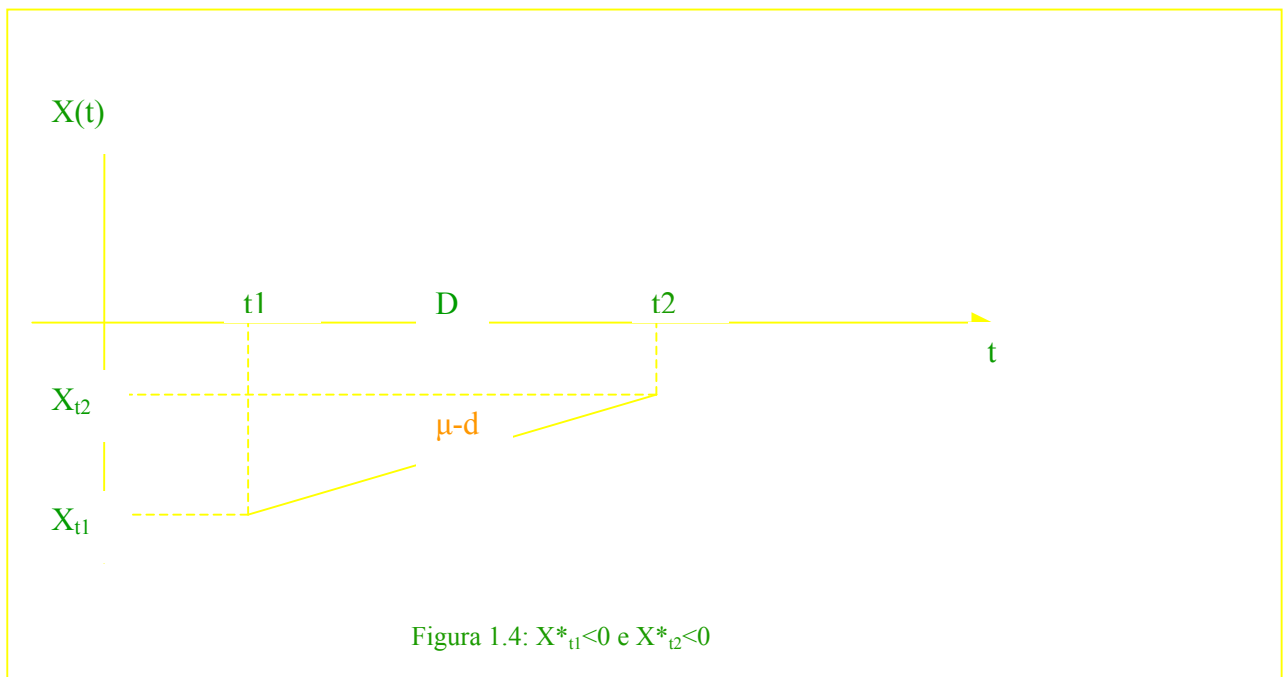
cioè quando la scorta è negativa il sistema cerca di portarla il più rapidamente possibile a 0, dopodiché, se possibile, si mantiene scorta nulla finché non è necessario portarsi al valore prefissato di scorta z^* .

Applicando tale politica possiamo distinguere per ogni ciclo generico di funzionamento tre casi distinti:

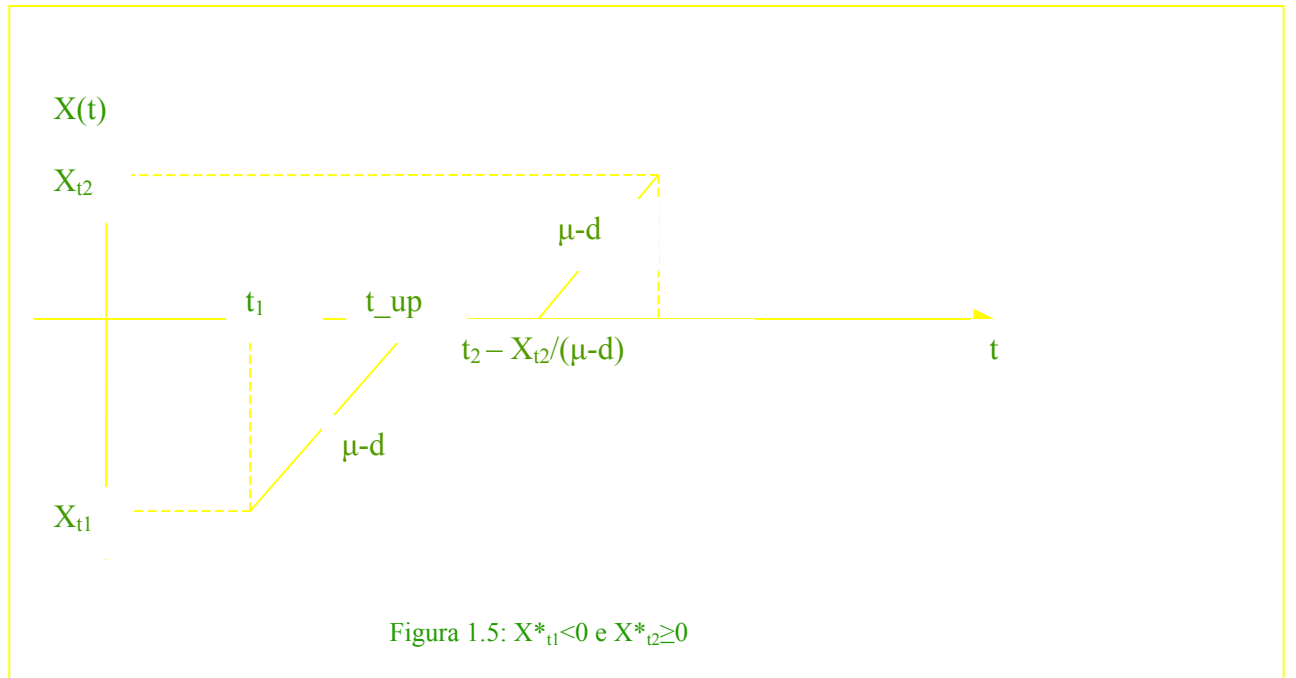
1) $X_{t1}^* < 0$ e $X_{t2}^* < 0$

In questo caso è dimostrato che $X_{t2}^* > X_{t1}^*$ (figura 1.4); ciò implica che

$$u_t^* = \mu \quad \forall t \in [t_1, t_2] \quad (29)$$



2) $X_{t1}^* < 0$ e $X_{t2}^* \geq 0$



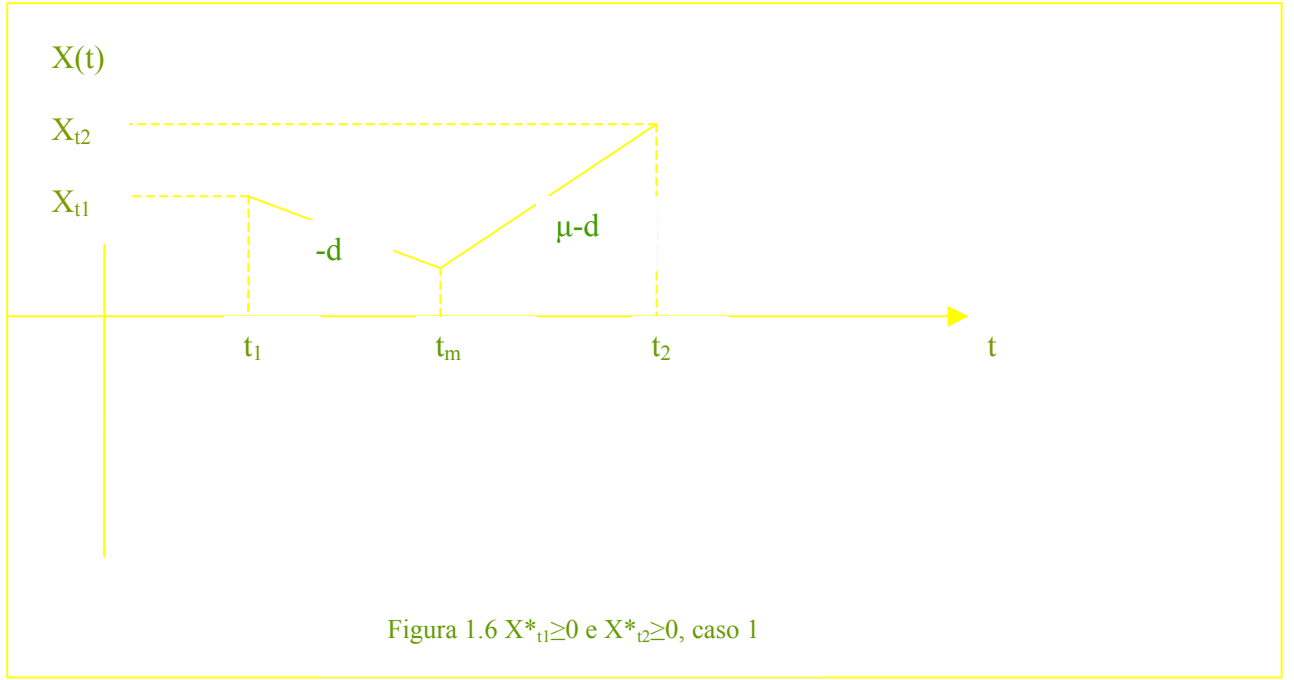
Se chiamiamo $t_{up} = |X_{t1}^*|/(\mu - d)$ il tempo che impiega il buffer a portarsi al valore nullo, abbiamo che $u_t^* = \mu \quad \forall t \in [t_1, t_{up}]$ con $\underline{X_t^*} < 0$, e segue che $\underline{X_t^*} \geq 0 \quad \forall t \in [t_{up}, t_2]$. Dobbiamo perciò costruire un nuovo controllo u_t (mostrato in figura 1.5) ampliamento di u_t^* :

$$u_t = \begin{cases} d & t \in [t_{up}, t_2 - X_{t2}^*/(\mu - d)] \\ \mu & t \in [t_2 - X_{t2}^*/(\mu - d), t_2] \\ u_t^* & \text{altrimenti} \end{cases} \quad (30)$$

3) $\underline{X_{t1}^*} \geq 0$ e $\underline{X_{t2}^*} \geq 0$

Abbiamo che $\underline{X_t} \geq 0 \quad \forall t$. Per questo caso però dobbiamo distinguere due possibili comportamenti del buffer:

a) $X_{t2}^*/(\mu - d) + X_{t1}^*/d \geq D$



E' questo il caso in cui il buffer non fa in tempo ad arrivare a zero prima che sia il momento di produrre al massimo per raggiungere il valore di scorta scelto (vedi figura 1.6). Perciò

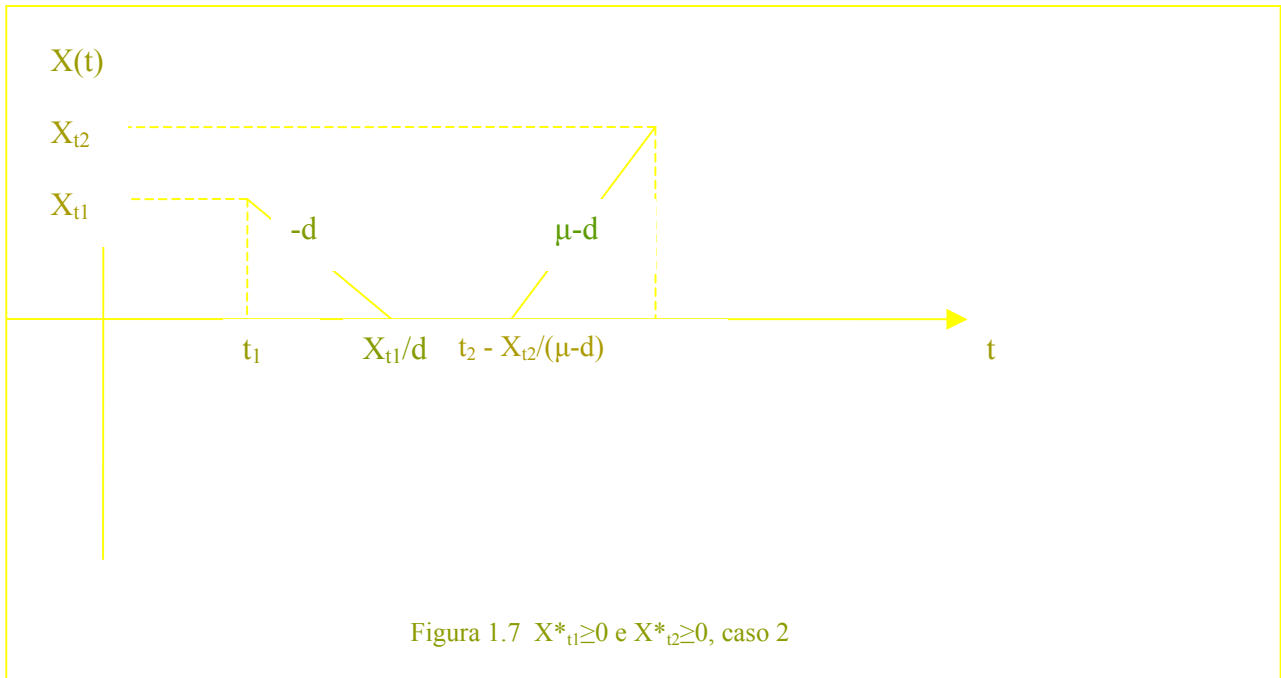
$$u_t = \begin{cases} 0 & t \in [t_1, t_1 + t_m] \\ \mu & t \in [t_1 + t_m, t_2] \\ u_t^* & \text{altrimenti} \end{cases} \quad (31)$$

con $t_m = (X_{t1}^* + (\mu - d) D - X_{t2}^*) / \mu$ che rappresenta il tempo impiegato dal buffer a raggiungere il punto dove necessariamente comincia a lavorare a potenza massima.

b) $X_{t2}^* / (\mu - d) + X_{t1}^* / d < D$

Il caso in cui il buffer riesca a mantenersi a zero per un certo tempo (vedi figura 1.7) viene definito dalla politica

$$u_t = \begin{cases} 0 & t \in [t_1, t_1 + X_{t1}^* / d] \\ d & t \in [t_1 + X_{t1}^* / d, t_2 - X_{t2}^* / (\mu - d)] \\ \mu & t \in [t_2 - X_{t2}^* / (\mu - d), t_2] \\ u_t^* & \text{altrimenti} \end{cases} \quad (32)$$



CAPITOLO 2 - Tempo di funzionamento e di guasto deterministici

2.1 Analisi generale

Prendendo spunto dalla politica appena enunciata abbiamo sviluppato il caso di sistemi D/D/1, cioè definendo anche il tempo di guasto del sistema come costante (con probabilità deterministica). Attraverso questa scelta, che non fa perdere di generalità i casi illustrati nel paragrafo 1.3, possiamo studiare il comportamento a ciclo chiuso, considerando che a seconda dei parametri caratteristici del sistema si rientrerà a regime in uno dei casi dello studio di sistemi D/G/1.

Analizziamo perciò un sistema di produzione a macchina singola che produce un solo tipo di parti, con domanda d costante e tasso di produzione $0 \leq u_t \leq \alpha_t \mu$ con α_t stato del sistema (0 se è guasta, 1 se è in funzione).

Il ciclo di produzione avviene a regime in maniera periodica con T_f tempo di funzionamento e T_g tempo di guasto, entrambi costanti. La funzione di costo è lineare a tratti e definita come $g(x) = c_p x^+ + c_m x^-$ con c_p e c_m costanti non negative corrispondenti rispettivamente a scorta positiva e negativa.

Perciò viene definita la funzione

$$J = \frac{1}{T} E \left[\int_0^T (c_p x^+ + c_m x^-) dt \right] \quad (33)$$

che sarà oggetto dello studio di politica ottima.

In questo tipo di sistemi la condizione necessaria per la stabilità è

$$(\mu - d) T_f - d T_g > 0 \quad (34)$$

Definiamo come t^* l'istante, misurato a partire dall'inizio dell'intervallo di funzionamento, in cui il sistema inizia a produrre a potenza massima per raggiungere il valore z di scorta:

$$t^* = T_f - z/(\mu - d) \quad (35)$$

Ne segue $0 \leq t^* \leq T_f$ che combinata con la (35) sviluppa il sistema

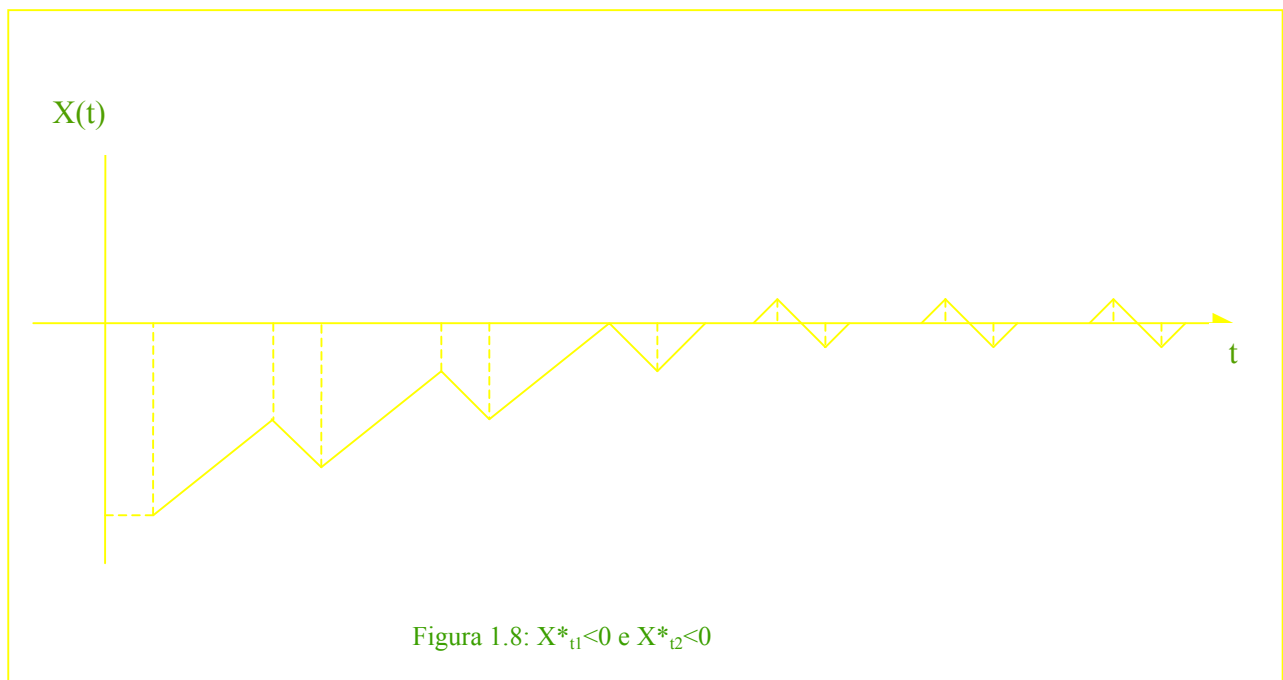
$$\begin{cases} T_f - z/(\mu - d) \geq 0 \\ T_f - z/(\mu - d) \leq T_f \end{cases} \quad (36)$$

da cui si ottiene

$$\begin{cases} z \leq T_f(\mu - d) \\ z \geq 0 \end{cases} \quad (37)$$

che sono i limiti superiore e inferiore della scorta z assegnabile al sistema.

A differenza dei 3 casi analizzati per sistemi D/G/1 a ciclo aperto, nei sistemi D/D/1 possiamo affermare con sicurezza la presenza di due soli casi a ciclo chiuso se viene soddisfatta la condizione (34) di stabilità poiché l'andamento del caso 1 dei sistemi D/G/1 a regime ricade nel caso 2 (vedi figura 1.8), ovvero il sistema a regime non può rimanere a buffer negativo:



Va considerato che la velocità in termini di cicli di funzionamento con cui il sistema si stabilizza su valori periodici dipende da quanto è stabile il sistema.

Le notazioni differiscono leggermente in quanto possiamo affermare a ciclo chiuso che:

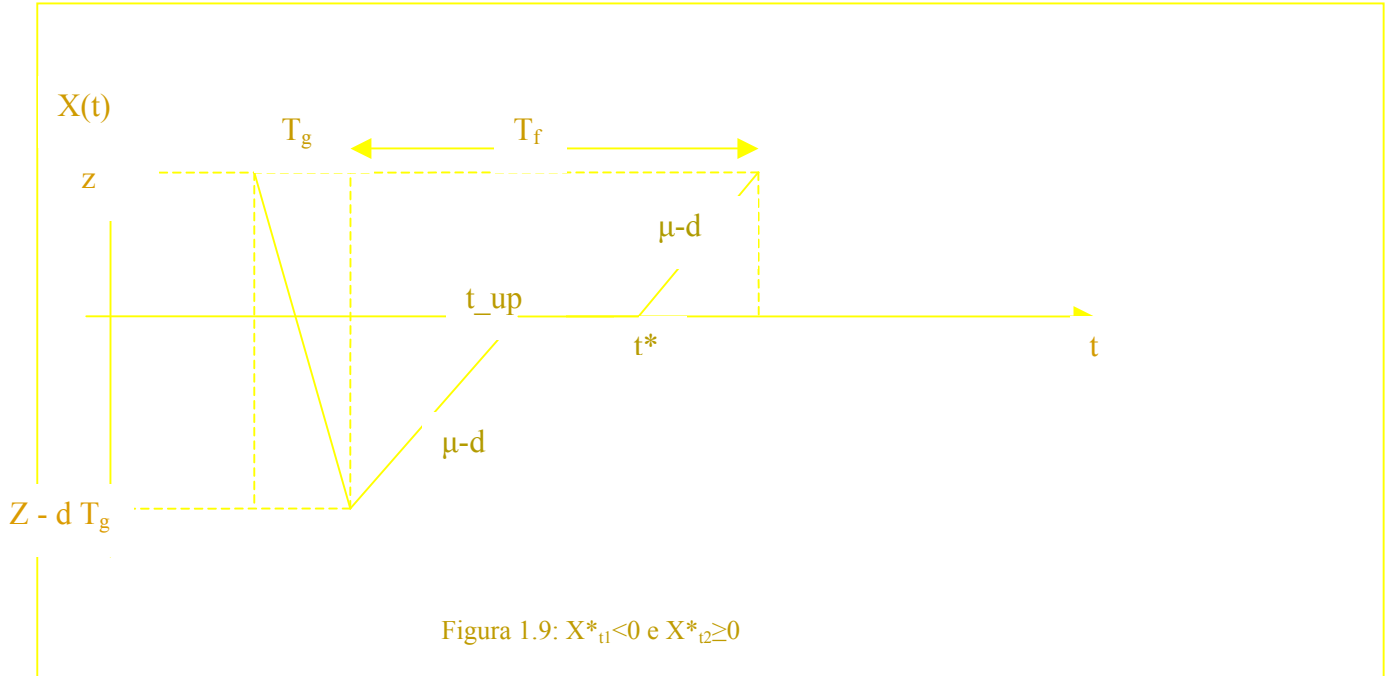
$$\begin{cases} X^*_{t1} = z - dT_g \\ X^*_{t2} = z \end{cases} \quad (38)$$

Analizziamo i casi corrispondenti alle possibili varianti del sistema:

1) $X^*_{t1} < 0$ e $X^*_{t2} \geq 0$

In questo caso la politica di produzione scelta è

$$u(t) = \begin{cases} 0 & t \in T_g \\ d & t \in [t_{up}, t^*] \\ \mu & \text{altrimenti} \end{cases} \quad (39)$$



2) $X^*_{t1} \geq 0$ e $X^*_{t2} \geq 0$

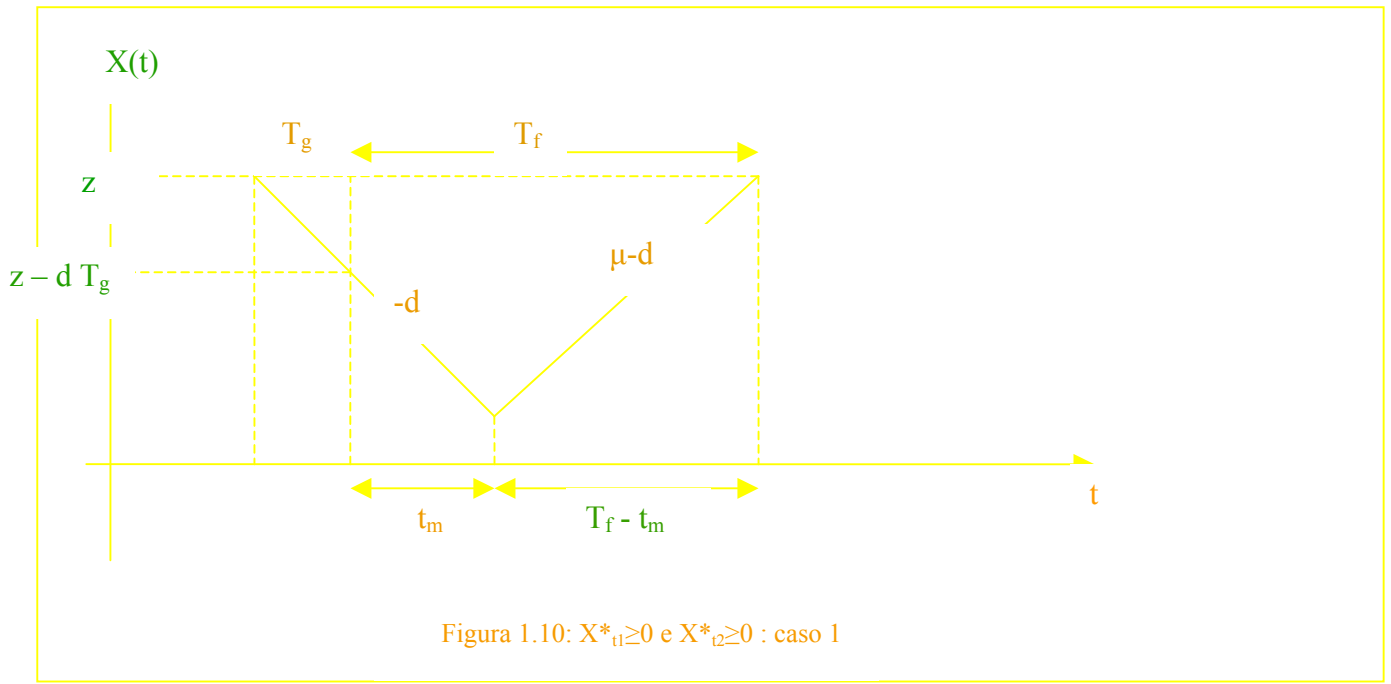
Si devono distinguere analogamente ai sistemi D/G/1 due possibilità:

a) $z/(\mu - d) + (z - dT_g)/d \geq T_f$

Il buffer non riesce a raggiungere lo zero prima di t_m , definito per i sistemi D/D/1

$$t_m = \frac{z - dT_g + T_f(\mu - d) - z}{\mu} \quad (40a)$$

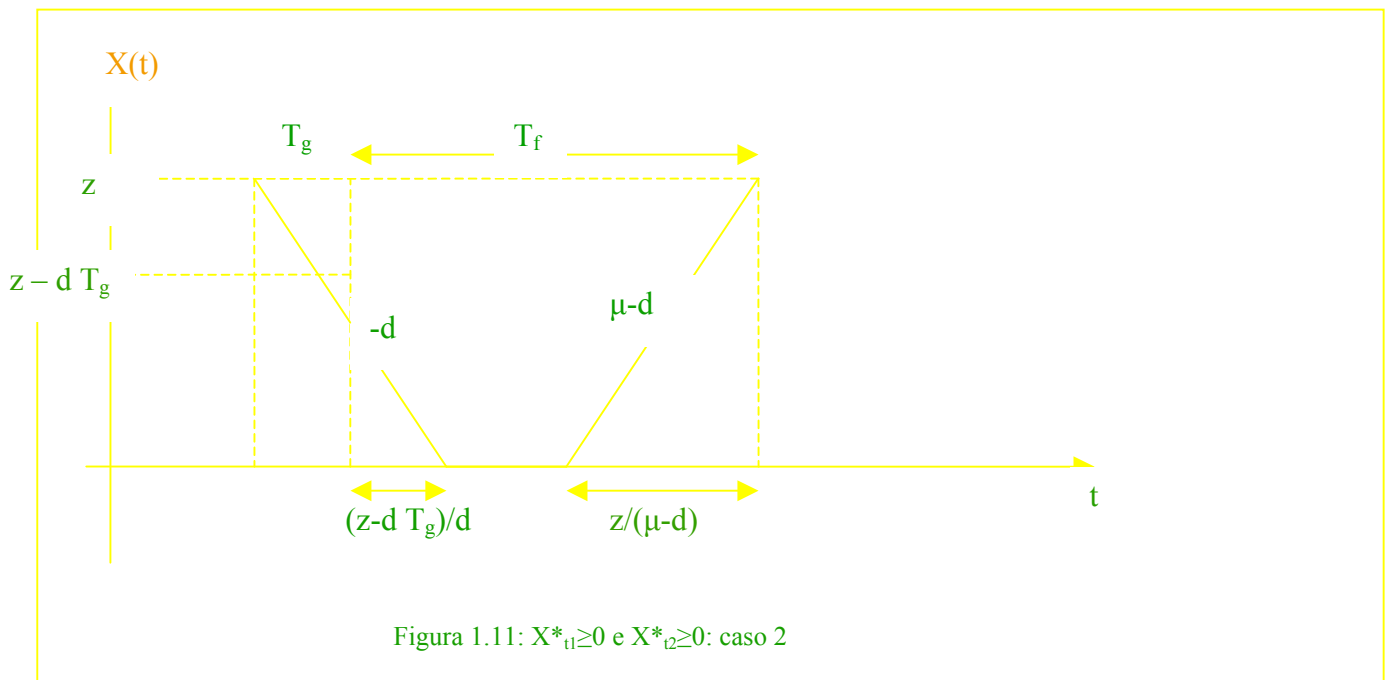
$$u(t) = \begin{cases} \mu & t > t_m \\ 0 & \text{altrimenti} \end{cases} \quad (40b)$$



b) $z/(\mu - d) + (z - dT_g)/d < T_f$

Nel caso in cui il buffer riesca ad arrivare a zero abbiamo la politica

$$u(t) = \begin{cases} 0 & t < (z - dT_g)/d \\ \mu & t > z/(\mu - d) \\ d & \text{altrimenti} \end{cases} \quad (41)$$



2.2 Studio della funzione di costo $J(z)$

Relativamente al caso di sistemi D/D/1 abbiamo illustrato nel precedente paragrafo i tre possibili stati nei quali il sistema potrebbe trovarsi a regime, determinati in base alle combinazioni plausibili del segno del buffer agli estremi del tempo di funzionamento. Avendo isolato la dinamica del sistema, che si presenta periodica dopo un tempo tanto minore quanto maggiore è il margine di stabilità definito come

$$S := T_f(\mu - d) - T_g d \quad (42)$$

possiamo anche determinare in forma chiusa la funzione di costo relativa ad ognuno dei tre casi.

2.2.1 Stato 1

Rifacendoci alla figura 1.8, possiamo dire che per la politica scelta a regime non avremo mai $z^* < 0$ in quanto proprio la condizione di stabilità (34) garantisce l'avvicinamento progressivo al valore nullo di z nel corso dei cicli di funzionamento/guasto. Lo stato 1 è infatti caratterizzato dalla presenza a regime di livelli di buffer negativi all'inizio del tempo di funzionamento.

Perciò l'estremo inferiore dello stato 1 è $z=0$ (vedi fig.1.12), corrispondente al minimo valore negativo di buffer accettabile ($x=-dT_g$) a regime.

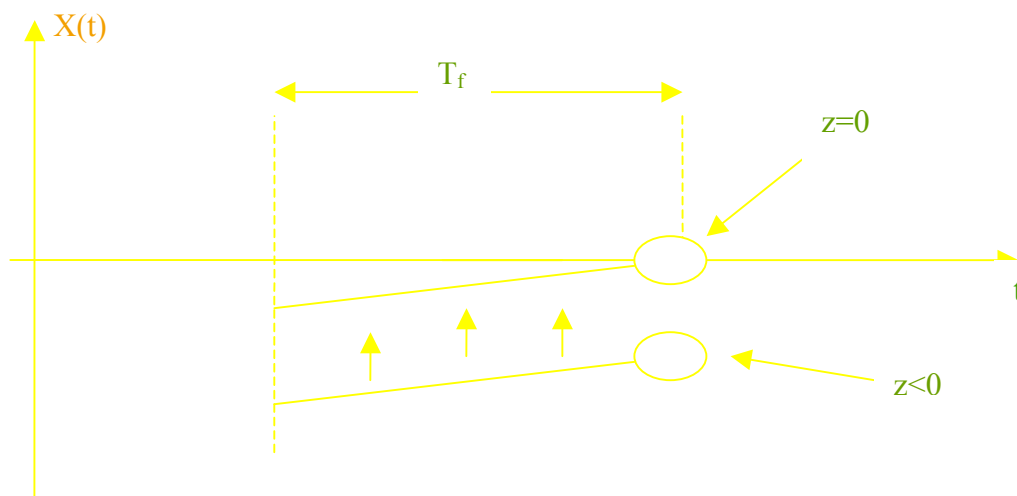


Figura 1.12 : Estremo inf. dello stato 1

L'estremo superiore dello stato è invece ricavato (vedi figura 1.13) dal passaggio dallo stato 1 ($z-d*T_g < 0$) allo stato 2 ($z-d*T_g > 0$) ed è perciò rappresentato proprio dal punto $z = d * T_g$, limite superiore della scorta con cui alla fine del tempo di guasto si ha un buffer negativo.

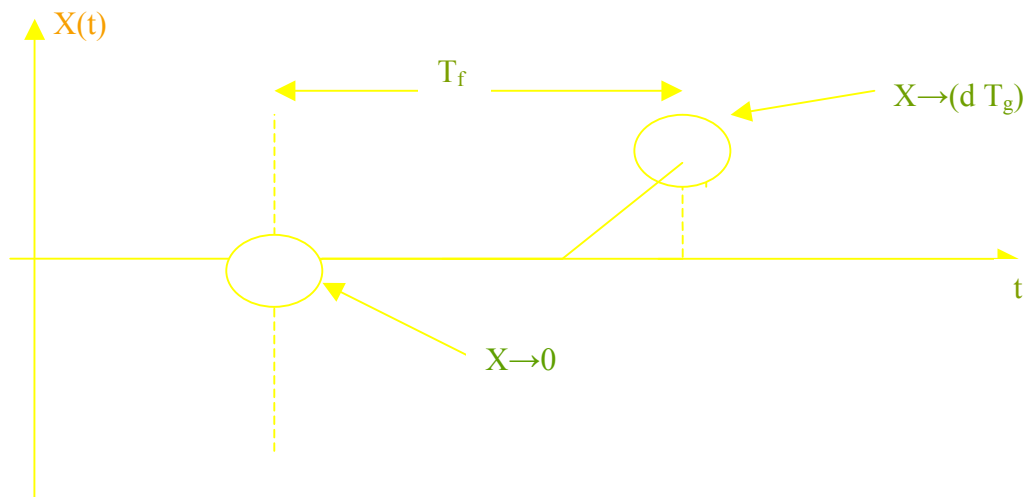


Figura 1.13 : Estremo sup. dello stato 1

Definito l'intervallo entro cui lavoriamo, cioè per $z \in [0, d * T_g)$ passiamo a visualizzare le aree di costo e a studiarle in forma chiusa (figura 1.14).

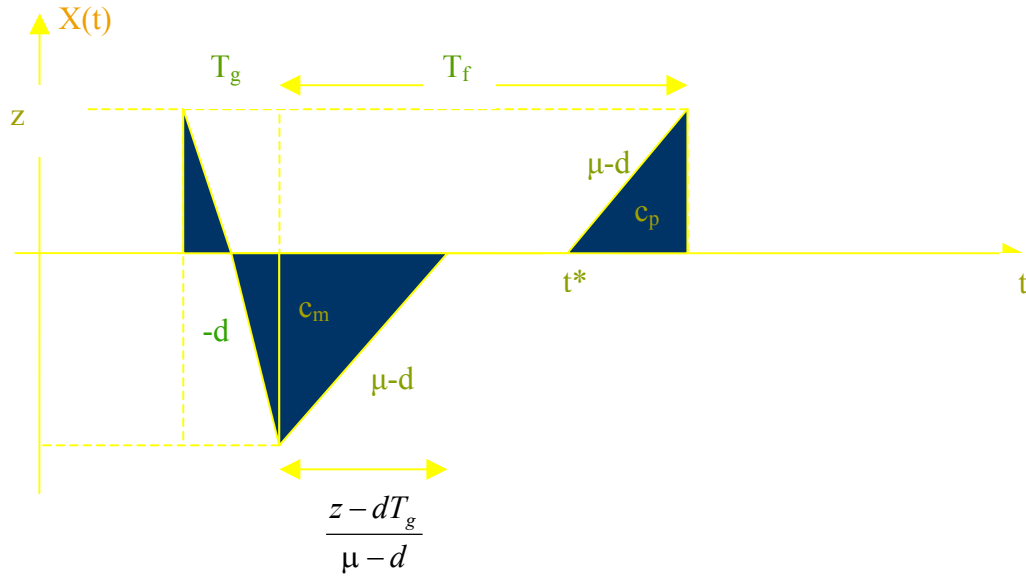


Figura 1.14: costo per $X_{t1}^* < 0$ e $X_{t2}^* \geq 0$

$$J(z) = \frac{1}{T} c_m \left[\frac{(z - dT_g)(z - dT_g)}{2(\mu - d)} + \frac{(z - dT_g)(z - dT_g)}{2d} \right] + \frac{1}{T} c_p \left[\frac{z \cdot z}{2(\mu - d)} + \frac{z \cdot z}{2d} \right] =$$

$$= \frac{1}{T} c_m \left[\frac{(z - dT_g)^2 d + (z - dT_g)^2 (\mu - d)}{2(\mu - d)d} \right] + \frac{1}{T} c_p \left[\frac{z^2 d + z^2 (\mu - d)}{2(\mu - d)d} \right]$$

$$J(z) = \frac{1}{T} c_m \left[\frac{(z - dT_g)^2 \mu}{2(\mu - d)d} \right] + \frac{1}{T} c_p \left[\frac{z^2 \mu}{2(\mu - d)d} \right] \quad (43)$$

Calcoliamo il valore della funzione di costo negli estremi dell'intervallo

$$J(z=0) = \frac{c_m d \mu T_g^2}{2(\mu - d)T} \quad (44a)$$

$$J(z \rightarrow dT_g) = \frac{c_p d \mu T_g^2}{2(\mu - d)T} \quad (44b)$$

Calcoliamo ora la derivata di $J(z)$ per capire la pendenza in questo tratto:

$$\frac{\partial J}{\partial z} = \frac{c_m \mu (z - d T_g)}{T d (\mu - d)} + \frac{c_p \mu z}{T d (\mu - d)} = \frac{(c_m + c_p) z \mu - c_m d T_g \mu}{T d (\mu - d)} \quad (45)$$

Per trovare il minimo della funzione poniamo la (45) uguale a 0 e troviamo

$$z_{stato1}^* = \frac{T_g d c_m}{c_m + c_p} \quad (46)$$

valore ottimo di scorta per minimizzare la funzione di costo relativa all'intervallo $z \in [0, d^* T_g]$.

Questo è dimostrato dal confronto di $J(z^*)$ con gli estremi dell'intervallo, che ha un unico punto di derivata nulla, e che scopriremo essere di minimo. Sostituendo la (46) nella (43) infatti otteniamo:

$$\begin{aligned} J(z^*) &= \frac{c_m \mu}{2 T d (\mu - d)} \left[\frac{T_g d c_m}{c_m + c_p} + T_g d \right]^2 + \frac{c_p \mu}{2 T d (\mu - d)} \left[\frac{T_g d c_m}{c_m + c_p} \right]^2 = \\ &= \frac{c_m \mu}{2 T d (\mu - d)} \left[\frac{T_g d c_p}{c_m + c_p} \right]^2 + \frac{c_p \mu}{2 T d (\mu - d)} \left[\frac{T_g d c_m}{c_m + c_p} \right]^2 = \\ &= \frac{c_m \mu T_g^2 d c_p^2}{2 T (\mu - d) (c_m + c_p)^2} + \frac{c_p \mu T_g^2 d c_m^2}{2 T (\mu - d) (c_m + c_p)^2} = \\ &= \frac{c_m c_p \mu T_g^2 d (c_m + c_p)}{2 T (\mu - d) (c_m + c_p)^2} = \\ &= \frac{c_m c_p \mu T_g^2 d}{2 T (\mu - d) (c_m + c_p)} \end{aligned} \quad (47)$$

Confrontando la (47) con le (44a) e (44b) è ovvio che

$$J(z^*) < J(0) \quad \forall z$$

$$J(z^*) < J(z \rightarrow d T_g) \quad \forall z$$

poiché i termini delle disuguaglianze sono uguali a meno di un numero sicuramente minore di 1.

Inoltre analizzando la derivata seconda di $J(z)$:

$$\frac{\partial^2 J}{\partial z^2} = \frac{c_m \mu}{Td(\mu - d)} + \frac{c_p \mu}{Td(\mu - d)} = \frac{(c_m + c_p) \mu}{Td(\mu - d)} \quad (48)$$

che è positiva per ogni valore dei parametri di sistema e indica la convessità di $J(z)$. Costruiamo così in maniera qualitativa il grafico relativo allo stato 1 di $J(z)$ (figura 1.15) comunque orientativo in quanto anche la differenza tra i due estremi dipende dal valore dei parametri di costo c_m e c_p .

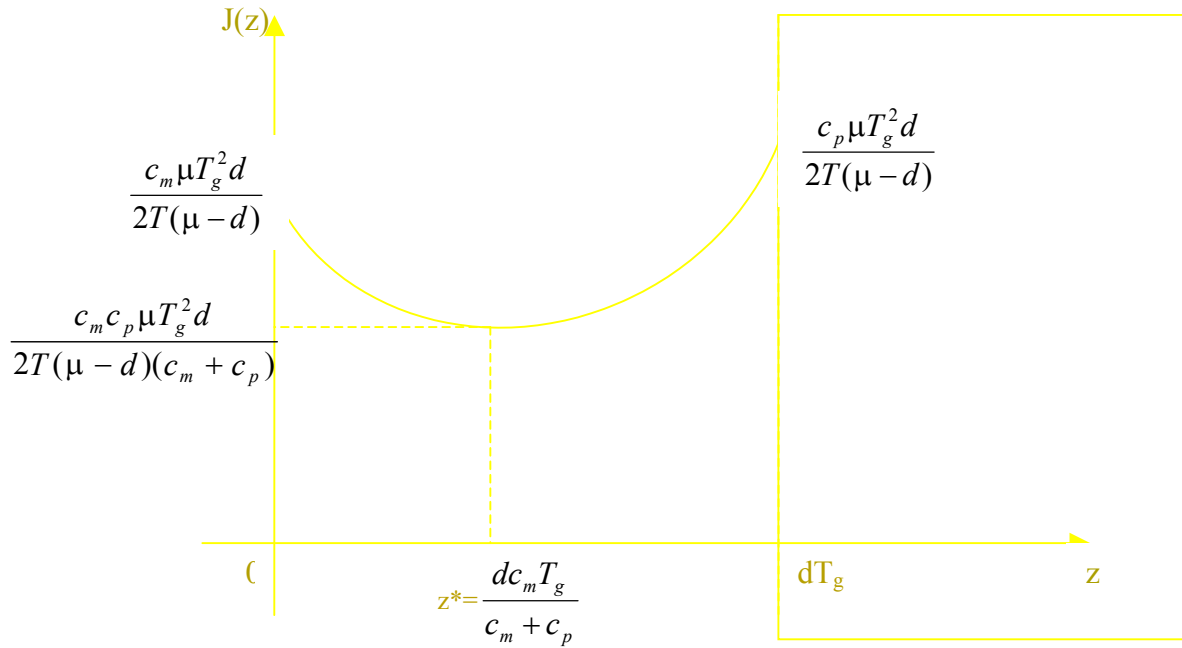


figura 1.15: grafico di $J(z)$ relativo allo stato 1

2.2.2 Stato 2

Definiamo lo stato 2 come quello caratterizzato a regime da

$$X(t) \geq 0 \quad (49)$$

a prescindere dalle scelte sui parametri del sistema con la condizione che ci sia almeno un istante δt in cui valga $X(t)=0$.

L'estremo inferiore dei valori di z di interesse è infatti la minima scorta per garantire a regime che la condizione (49) valga per l'inizio del tempo di funzionamento ($z - dT_g \geq 0$):

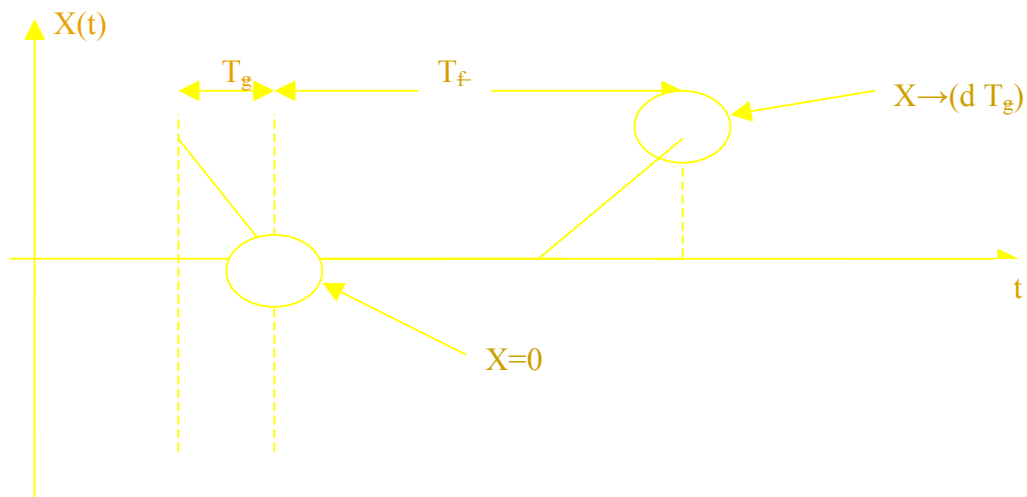


Figura 1.16 : Estremo inf. dello stato 2

Considerando la (49) e la condizione che deve esserci almeno un istante in cui $X(t)=0$, l'estremo superiore è proprio il caso in cui si abbia un solo istante con scorta nulla:

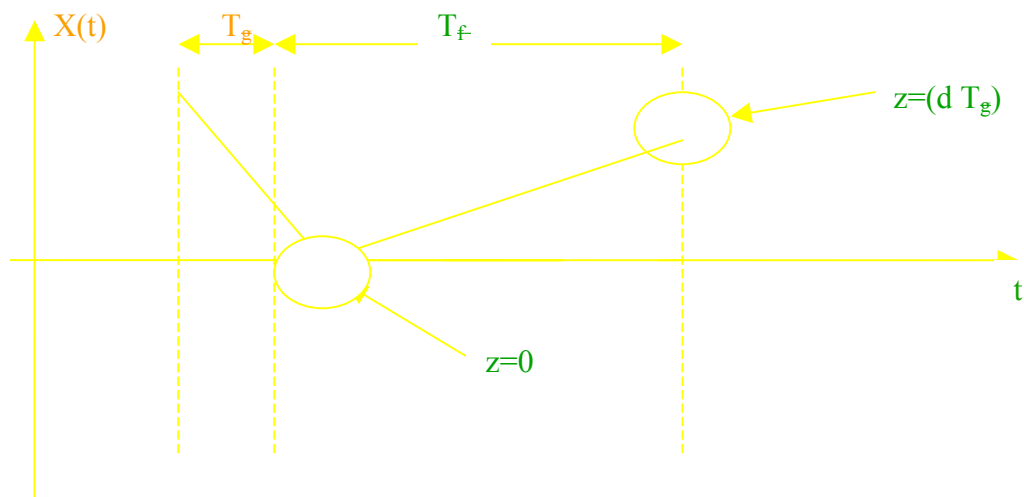


Figura 1.17 : Estremo sup. dello stato 2

L'espressione che quindi determina tale valore è

$$z/(\mu - d) + (z - dT_g)/d = T_f \quad (50)$$

da cui si ricava

$$z \frac{\mu}{d(\mu - d)} = T_f + T_g$$

$$z = \frac{d}{\mu} (\mu - d)(T_f + T_g) \quad (51)$$

Perciò per $z \in [d * T_g, \frac{d(\mu - d)(T_f + T_g)}{\mu}]$ studiamo in forma chiusa l'integrale della funzione di costo:

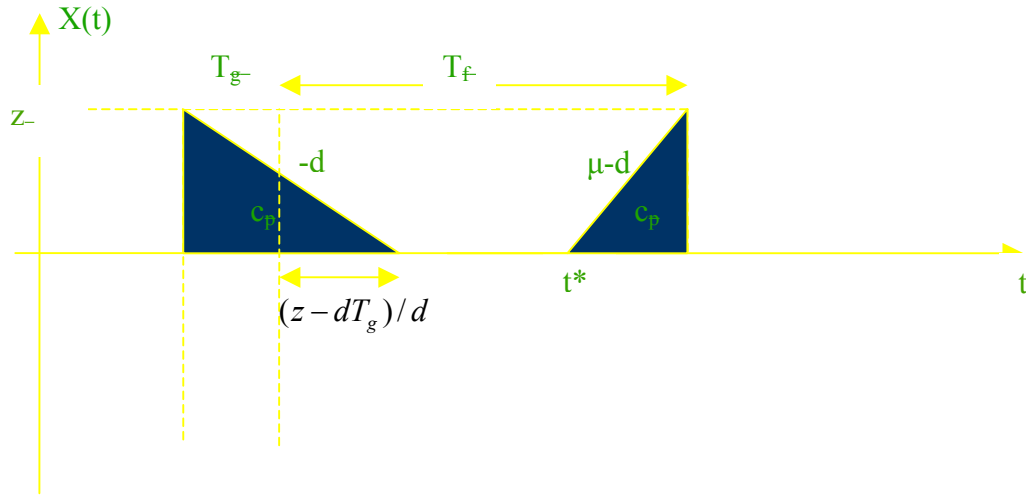


Figura 1.18: costo per $X^*_{t1} \geq 0$ e $X^*_{t2} \geq 0$

$$\begin{aligned}
 J(z) &= \frac{c_p}{T} \left[\frac{(z - dT_g)^2}{2d} + \frac{z^2}{2(\mu - d)} + \frac{T_g(z + z - dT_g)}{2} \right] = \\
 &= \frac{c_p}{T} \left[\frac{(z - dT_g)^2}{2d} + \frac{z^2}{2(\mu - d)} + \frac{T_g(2z - dT_g)}{2} \right]
 \end{aligned} \tag{52}$$

Gli estremi dell'intervallo dello stato 2 corrispondono a

$$J(z = dT_g) = \frac{c_p}{T} \left(\frac{T_g^2 d^2}{2(\mu - d)} + \frac{T_g - dT_g}{2} \right) = \frac{c_p}{T} \left(\frac{T_g^2 d^2 + T_g^2 d\mu - T_g^2 d^2}{2(\mu - d)} \right) = \frac{c_p T_g^2 d\mu}{2T(\mu - d)} \tag{53}$$

$$\begin{aligned}
J(z \rightarrow \frac{d(\mu-d)T}{\mu}) &= \frac{c_p}{T} \left[\frac{\left(\frac{d(\mu-d)T}{\mu} - dT_g \right)^2}{2d} + \frac{\left(\frac{d(\mu-d)T}{\mu} \right)^2}{2(\mu-d)} + \frac{T_g \left(2 \frac{d(\mu-d)T}{\mu} - dT_g \right)}{2} \right] = \\
&= \frac{c_p}{T} \left[\frac{d}{2\mu^2} ((\mu-d)T - T_g \mu)^2 + \frac{d}{2\mu^2} (\mu-d)T^2 + \frac{dT_g}{2\mu} (2T(\mu-d) - \mu T_g) \right] = \\
&= \frac{c_p d}{2T\mu} \left[\frac{(T(\mu-d) - \mu T_g)^2}{\mu} + \frac{dT^2(\mu-d)}{\mu} + T_g (2T(\mu-d) - \mu T_g) \right] = \\
&= \frac{c_p d}{2T\mu^2} \left[(T(\mu-d) - \mu T_g)^2 + dT^2(\mu-d) + \mu T_g (2T(\mu-d) - \mu T_g) \right] = \\
&= \frac{c_p d}{2T\mu^2} \left[(T^2(\mu-d)^2 + \mu^2 T_g^2 - 2\mu T T_g (\mu-d)) + dT^2(\mu-d) + 2\mu T_g T(\mu-d) - \mu^2 T_g^2 \right] = \\
&= \frac{c_p d}{2T\mu^2} (T^2(\mu-d)(\mu-d+d)) = \frac{c_p d T^2 \mu (\mu-d)}{2T\mu^2} = \frac{c_p d T (\mu-d)}{2\mu}
\end{aligned} \tag{54}$$

La (54) è sempre maggiore della (53) in condizioni di stabilità.

Calcoliamo la derivata per capire l'andamento del grafico di questo tratto di J(z):

$$\frac{\partial J}{\partial z} = \frac{c_p}{T} \left[\frac{z - dT_g}{d} + \frac{z}{\mu - d} + T_g \right] = \frac{c_p (z\mu - zd - T_g d\mu + d^2 T_g + zd)}{Td(\mu-d)} + \frac{T_g c_p}{T} \tag{55}$$

Ponendo la (55) uguale a zero troviamo il minimo della funzione

$$\begin{aligned}
z\mu - T_g d(\mu-d) &= -T_g d(\mu-d) \\
z^*_{stato2} &= 0
\end{aligned} \tag{56}$$

e calcolando la derivata seconda di J(z)

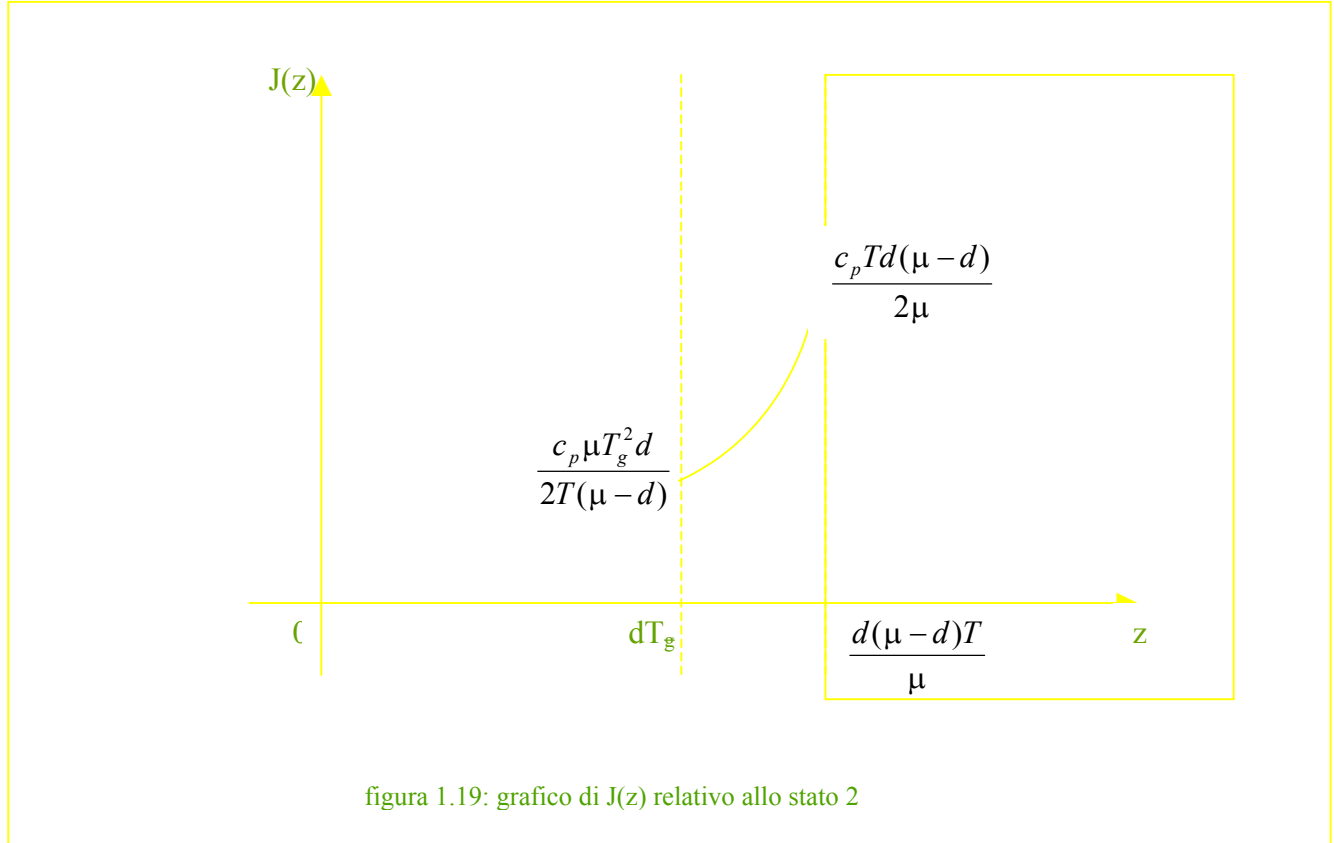
$$\frac{\partial^2 J}{\partial z^2} = \frac{c_p \mu}{dT(\mu-d)} \tag{57}$$

sempre positiva per ogni scelta dei parametri del sistema. Quindi il valore z^* della (56) è un valore di minimo della (52) (considerando anche che $J(0)=0$). Confrontando però la (56) con la (46), che esprime il valore minimo per la funzione J(z) espressa dallo stato 1, scopriamo che

$$z^*_{\text{stato2}} < z^*_{\text{stato1}}$$

(58)

perciò il minimo relativo allo stato 2 cade all'interno dello stato 1 e dando luogo così ad un grafico del tipo



2.2.3 Stato 3

Analizziamo ora il caso in cui troviamo in tutto il ciclo di funzionamento e guasto a regime

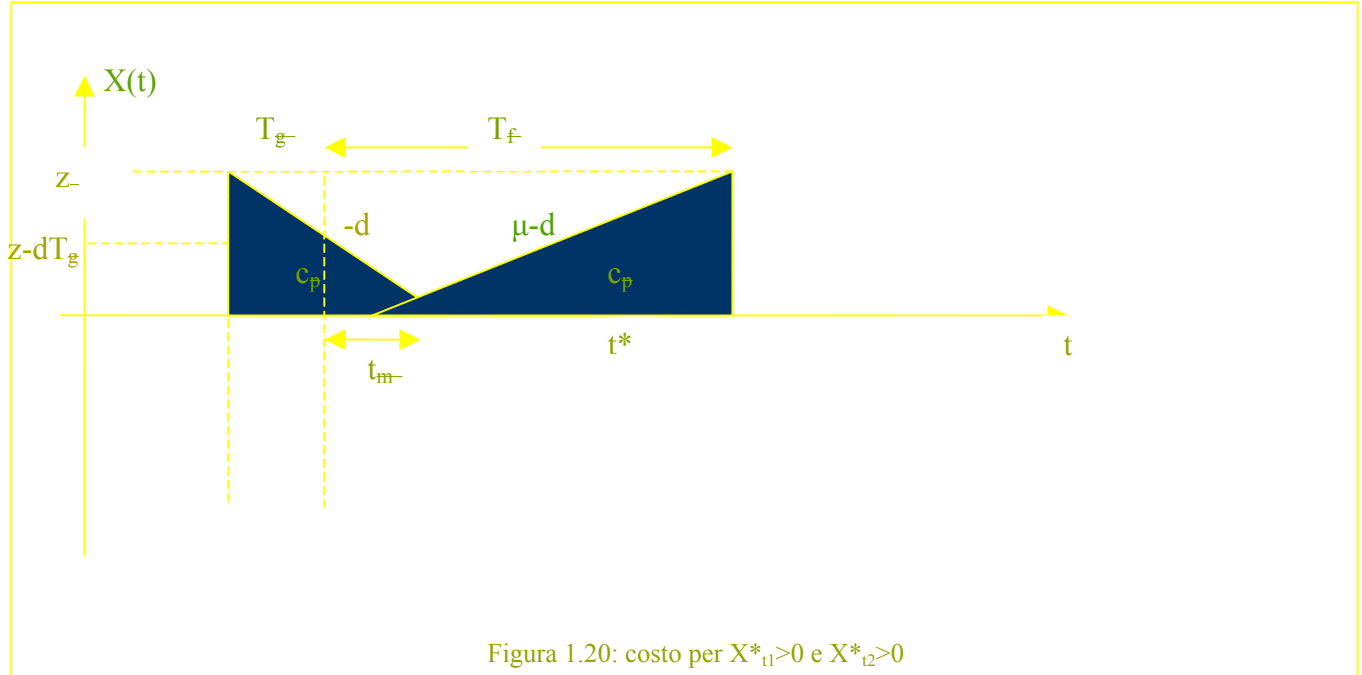
$$X(t) > 0 \quad (59)$$

Il valore minimo di scorta per cui vale questo è ovviamente quello determinato dalla (50) (vedi figura 1.17) per cui

$$z/(\mu - d) + (z - dT_g)/d = T_f \quad (60)$$

mentre il valore massimo coincide con la massima scorta assegnabile al sistema con la politica (35) scelta e cioè quella determinata dal sistema (37): lavoriamo perciò

nell'intervallo $z \in [\frac{d(\mu-d)(T_f+T_g)}{\mu}, (\mu-d)T_f]$ e ne studiamo l'integrale di costo in forma chiusa:



$$J(z) = \frac{c_p}{2T} \left[((z-dT_g) + (z-dT_g) - dt_m)t_m + (z+z-(T_f-t_m)(\mu-d))(T_f-t_m) + (z+z-dT_g)T_g \right] \quad (61)$$

considerando t_m quello definito nella (40a)

$$\begin{aligned} J(z) &= \frac{c_p}{2T} \left[\left(2z\mu - 2T_g d\mu - dT_f(\mu-d) + T_g d^2 \right) \left(T_f(\mu-d) + dT_g \right) \frac{1}{\mu^2} + \left(2z - \frac{(T_f\mu - T_f(\mu-d) + T_g d)(\mu-d)}{\mu} \right) \right. \\ &\quad \cdot \left. \left(\frac{(T_f\mu - T_f(\mu-d) + T_g d)}{\mu} \right) + (2z - T_g d)T_g \right] = \\ &= \frac{c_p}{2T} \left[\left(2z\mu - 2T_g d\mu - dT_f(\mu-d) + T_g d^2 \right) \left(T_f(\mu-d) + dT_g \right) \frac{1}{\mu^2} + \left(2z - \frac{(T_f d + T_g d)(\mu-d)}{\mu} \right) \right. \\ &\quad \cdot \left. \left(\frac{(T_f d + T_g d)}{\mu} \right) + 2zT_g - T_g^2 d \right] = \end{aligned}$$

$$\begin{aligned}
&= \frac{c_p}{2T\mu^2} \left[(2z\mu - 2T_g d\mu - dT_f \mu + Td^2)(T_f \mu + dT) + (2z\mu - (Td)(\mu - d))Td + 2zT_g \mu^2 - T_g^2 d\mu^2 \right] = \\
&= \frac{c_p}{2T\mu^2} \left[2z\mu^2 T_f - 2zTd\mu - 2T_g T_f d\mu^2 + 2TT_g d^2 \mu - T_f^2 d\mu^2 + TT_f d^2 \mu + TT_f d^2 \mu - T^2 d^3 + 2zTd\mu - T^2 d^2 \mu + T^2 d^3 + \right. \\
&\quad \left. + 2zT_g \mu^2 - T_g^2 d\mu^2 \right] = \\
&= \frac{c_p}{2T\mu^2} \left[2z\mu^2 T - T^2 d\mu^2 + 2T^2 d^2 \mu - T^2 d^2 \mu \right] = \frac{c_p}{2T\mu^2} \left[2z\mu^2 T - T^2 d\mu^2 + T^2 d^2 \mu \right]
\end{aligned}$$

semplificata in

$$J(z) = \frac{c_p}{2\mu} [2z\mu - Td\mu + Td^2] = \frac{c_p}{2\mu} [2z\mu - Td(\mu - d)] \quad (62)$$

termine che rappresenta una retta con pendenza

$$\frac{\partial J}{\partial z} = \frac{c_p 2\mu}{2\mu} = c_p \quad (63)$$

e il cui valore agli estremi dell'intervallo di nostro interesse è dato da

$$J\left(z \rightarrow \frac{dT(\mu - d)}{\mu}\right) = \frac{c_p}{2\mu} [2dT(\mu - d) - dT(\mu - d)] = \frac{c_p Td(\mu - d)}{2\mu} \quad (64)$$

$$\begin{aligned}
J(z \rightarrow T_f(\mu - d)) &= \frac{c_p}{2\mu} [2T_f \mu^2 - 2T_f d\mu - Td\mu + Td^2] = \frac{c_p}{2\mu} [2T_f \mu(\mu - d) - Td(\mu - d)] = \\
&= \frac{c_p}{2\mu} [(2T_f \mu - Td)(\mu - d)] \quad (65)
\end{aligned}$$

Otteniamo perciò il grafico relativo a questo tratto della funzione di costo del tipo di figura 1.21:

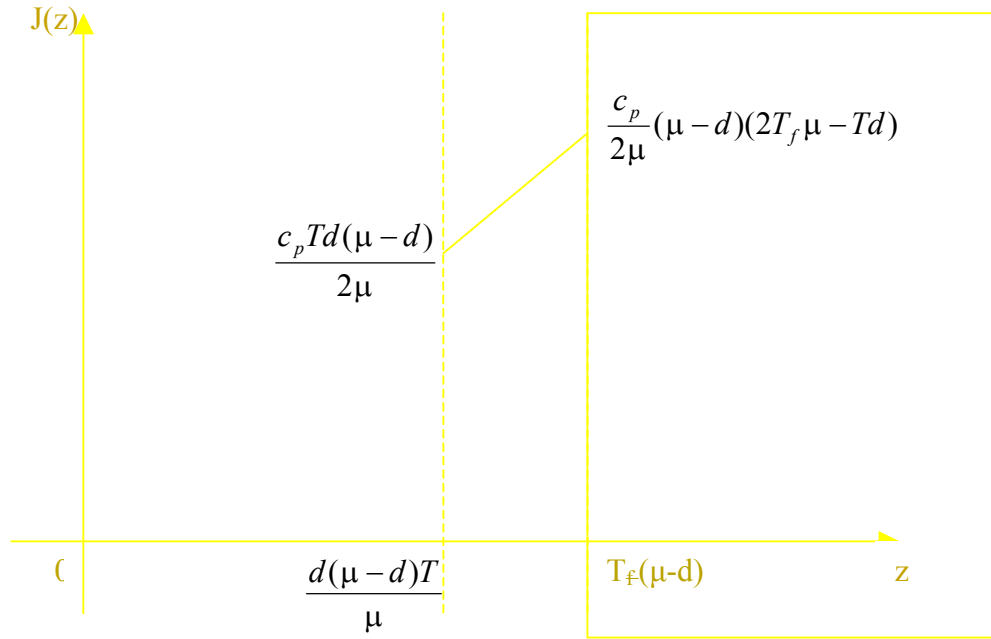


figura 1.21: grafico di $J(z)$ relativo allo stato 3

2.2.4 Studio della continuità e della derivabilità della $J(z)$

La continuità è assicurata in ogni punto del grafico: singolarmente ogni stato associa una funzione $J(z)$ continua e derivabile:

$$J(z) = \begin{cases} \frac{c_m}{T} \left[\frac{(z - dT_g)^2 \mu}{2(\mu - d)d} \right] + \frac{c_p}{T} \left[\frac{z^2 \mu}{2(\mu - d)d} \right] & z \in [0, d * T_g) \\ \frac{c_p}{T} \left[\frac{(z - dT_g)^2}{2d} + \frac{z^2}{2(\mu - d)} + \frac{T_g(2z - dT_g)}{2} \right] & z \in [d * T_g, \frac{d(\mu - d)(T_f + T_g)}{\mu}) \\ \frac{c_p}{2\mu} [2z\mu - Td(\mu - d)] & z \in [\frac{d(\mu - d)(T_f + T_g)}{\mu}, (\mu - d)T_f] \end{cases} \quad (66)$$

perciò per garantire la continuità di tutta la funzione dobbiamo studiare i limiti sinistro e destro sui punti di frontiera degli intervalli. Le (44b) e (53) dimostrano che nel punto di frontiera tra stato 1 e 2 ($z = d T_g$) non c'è discontinuità. Discorso analogo per il confine tra stato 2 e 3 ($z = \frac{d}{\mu}(\mu - d)(T_f + T_g)$) che garantisce così la continuità su tutto l'intervallo di interesse dei valori di scorta.

Discorso analogo per la derivabilità di tali punti. Sostituendo nella (45) e nella (55) il punto $z = d T_g$ otteniamo:

$$\lim_{z \rightarrow (dT_g)^-} \frac{\partial J}{\partial z} = \left[\frac{c_p \mu T_g}{T(\mu - d)} \right] \quad (67)$$

$$\lim_{z \rightarrow (dT_g)^+} \frac{\partial J}{\partial z} = \frac{c_p (d^2 T_g)}{Td(\mu - d)} + \frac{T_g c_p}{T} = \frac{T_g c_p}{T} \left(\frac{d}{\mu - d} + 1 \right) = \frac{T_g c_p (d + \mu - d)}{T(\mu - d)} = \frac{c_p \mu T_g}{T(\mu - d)} \quad (68)$$

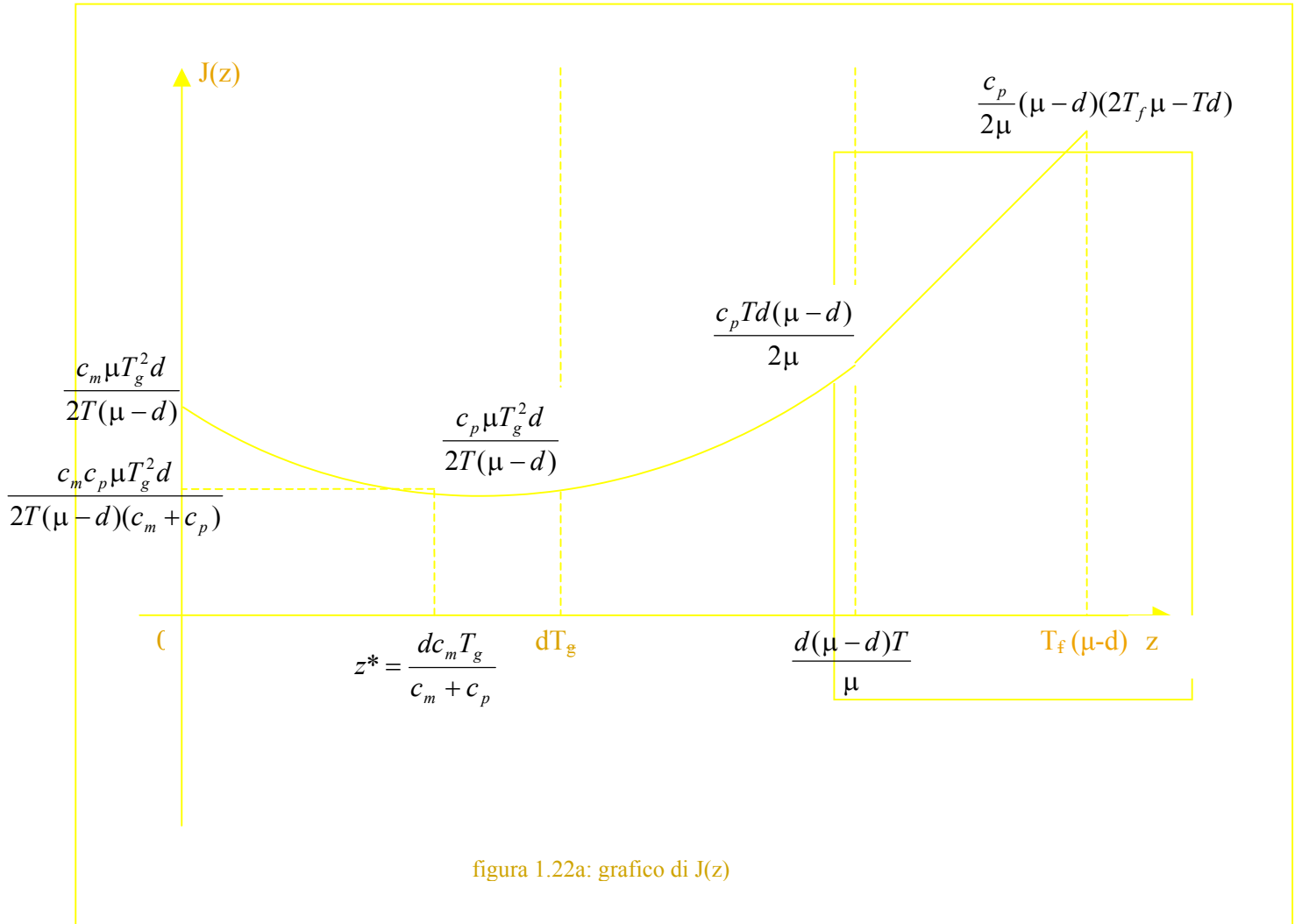
che evidenziano la continuità della derivata in quel punto. Otterremo lo stesso risultato facendo gli stessi calcoli nella (55) per il secondo punto di frontiera e confrontandolo con la (63):

$$\begin{aligned} \lim_{z \rightarrow \left(\frac{d(\mu - d)(T_f + T_g)}{\mu} \right)^-} \frac{\partial J}{\partial z} &= \frac{c_p (d(\mu - d)(T_f + T_g) - T_g d\mu + d^2 T_g)}{Td(\mu - d)} + \frac{T_g c_p}{T} = \\ &= \frac{c_p T_f d(\mu - d)}{Td(\mu - d)} + \frac{T_g c_p}{T} = \frac{c_p (T_f + T_g)}{T} = c_p \end{aligned} \quad (69)$$

$$\lim_{z \rightarrow \left(\frac{d(\mu - d)(T_f + T_g)}{\mu} \right)^+} \frac{\partial J}{\partial z} = c_p \quad (70)$$

2.3 Grafico di J(z)

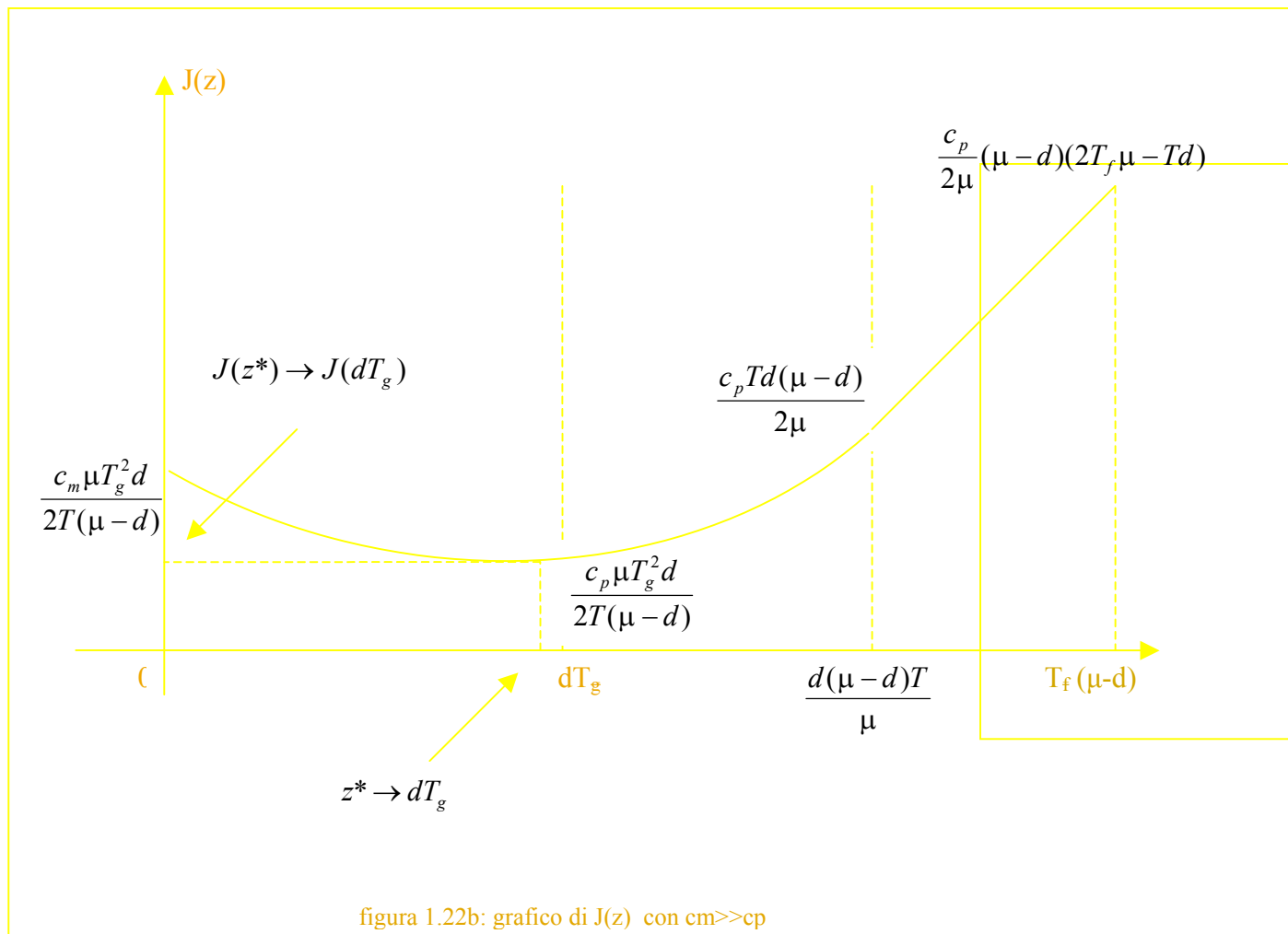
Otteniamo perciò il grafico completo della funzione di costo J(z) relativa a sistemi del tipo D/D/1:



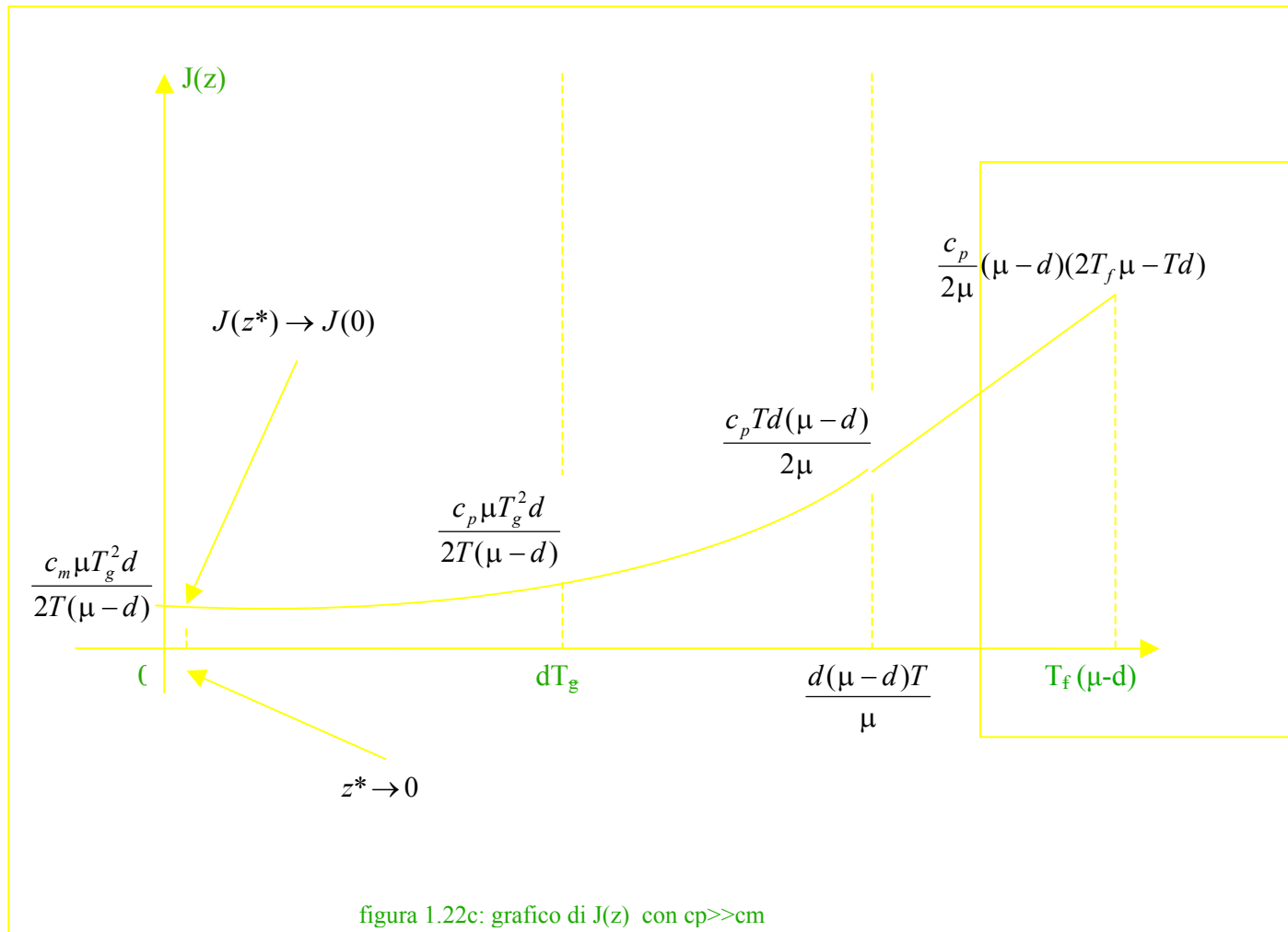
E' quindi dimostrato analiticamente che tale funzione ha un valore minimo in corrispondenza di z^* , livello di scorta ottimo per il sistema che lavori sotto la politica scelta.

Possiamo inoltre dedurre dalla somiglianza dei termini di costo nel primo tratto della funzione che le caratteristiche geometriche della parabola in questa regione dipenderanno dal rapporto tra i due parametri di costo. Nelle due pagine seguenti si mostreranno due probabili funzioni di costo a seconda dei valori reciproci tra c_p e c_m .

Se $c_m \gg c_p$



Se $c_p \gg c_m$



2.4 Simulazioni algoritmiche di sistemi D/D/1

Sono state effettuate delle prove attraverso un algoritmo implementato in C++ (presente nell'Appendice) che simula il comportamento del sistema in 1000 cicli di funzionamento-guasto per studiare a regime l'andamento della funzione $J(z)$ e ricavare a livello empirico il valore di z^* minimo per un costo ottimo. Impostati i parametri di produzione (μ, d), di costo (c_m, c_p) e di tempo (T_f, T_g) l'algoritmo visualizza i valori di costo corrispondente ad ogni valore di scorta analizzato e salva i risultati su un file .m utilizzato poi per graficare la funzione $J(z)$ con Matlab.

Esempio 2.1

Se analizziamo il sistema con le caratteristiche:

PARAMETRI DI LAVORO DELL'ALGORITMO

Potenza max dell'impianto μ : 3

Tasso di domanda costante d : 2

c_m : 80

c_p : 20

Tempo di guasto (t_g): 5

Tempo di funzionamento (t_f): 150

Fattore di stabilità del sistema S : 140

Passo di t_{star} (istante di inizio del lavoro a potenza max): 1

Passo di z_{star} (massima scorta dell'impianto): 1

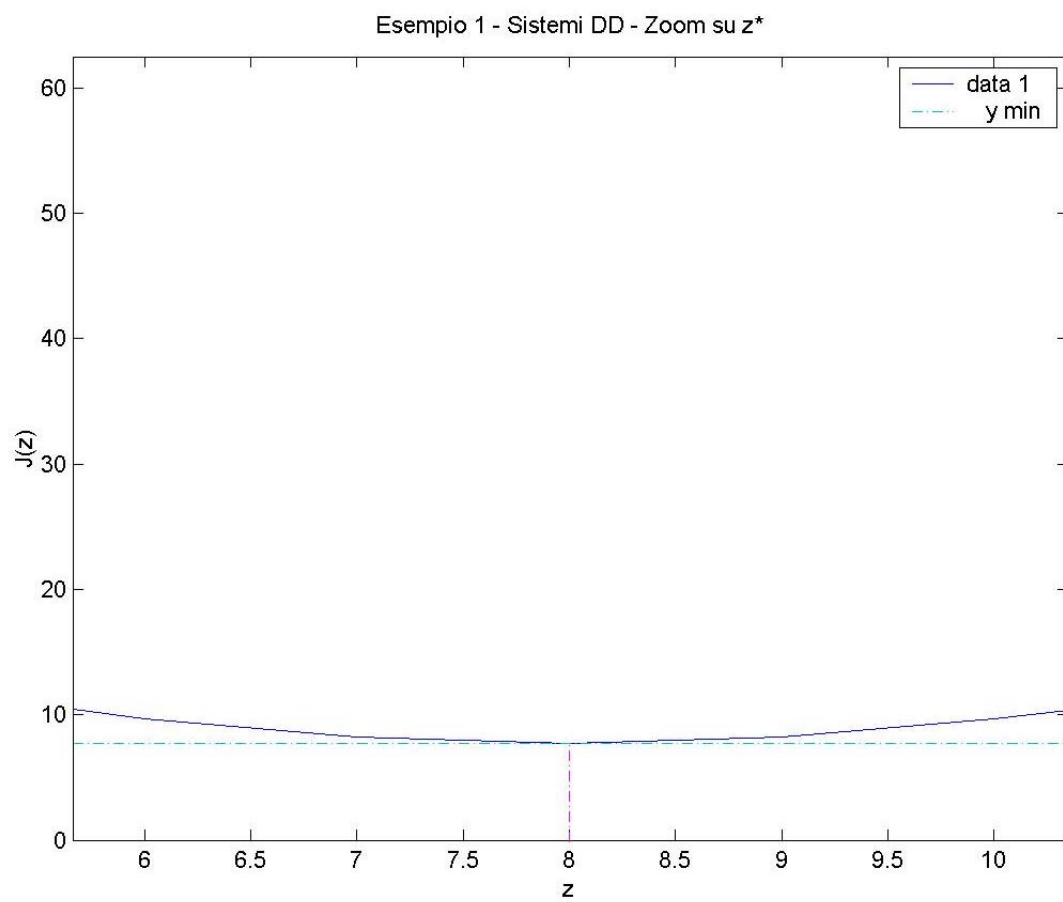
Valore di SCORTA OTTIMA atteso: 8

Valore di COSTO MINIMO atteso: 7.74194

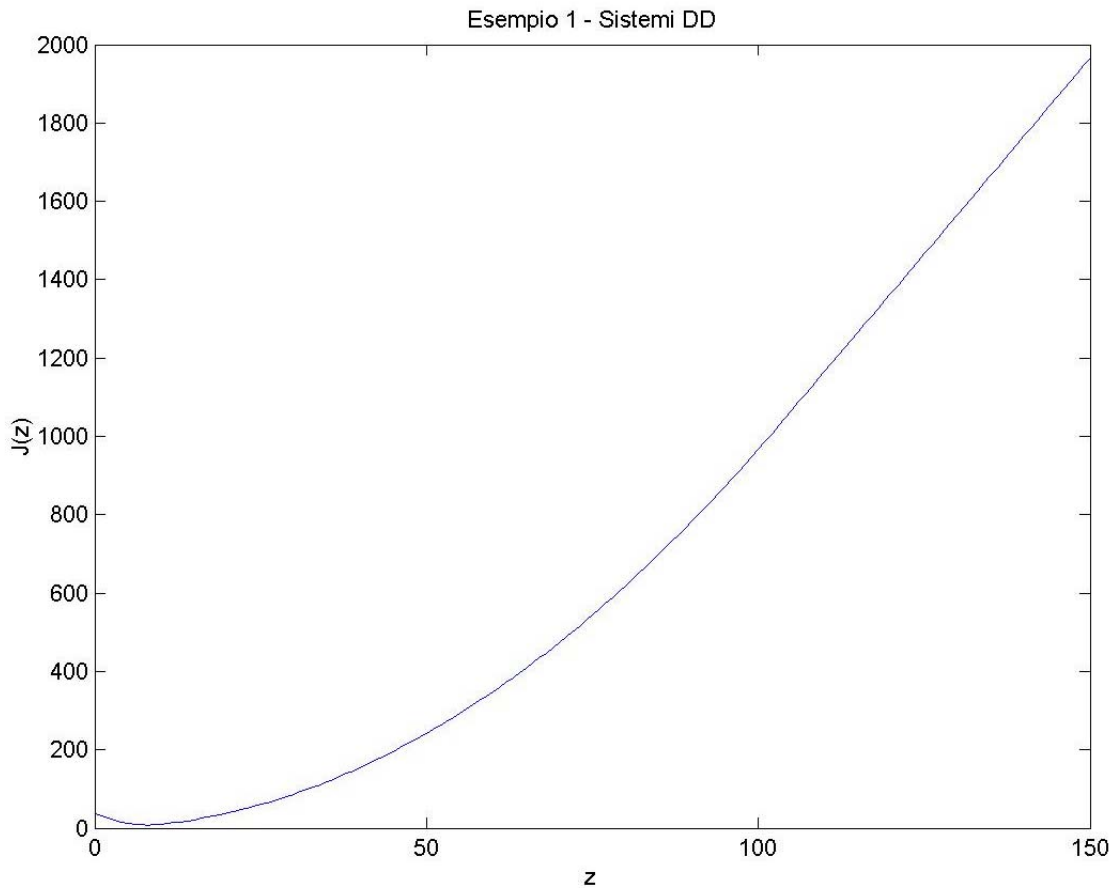
.....

dall'algoritmo otteniamo

“Il valore minimo di costo trovato è: 7.73987 corrispondente allo z^ : 8”*



Il grafico di $J(z)$ corrispondente a questa simulazione è



Esempio 2.2

Per un sistema al limite di stabilità con queste caratteristiche

PARAMETRI DI LAVORO DELL'ALGORITMO

Potenza max dell'impianto μ : 3

Tasso di domanda costante d : 2.8

c_m : 80

c_p : 20

Tempo di guasto (t_g): 5

Tempo di funzionamento (t_f): 150

Fattore di stabilità del sistema S : 16

Passo di t_{star} (istante di inizio del lavoro a potenza max): 1

Passo di z_{star} (massima scorta dell'impianto): 0.5

Valore di SCORTA OTTIMA atteso: 11.2

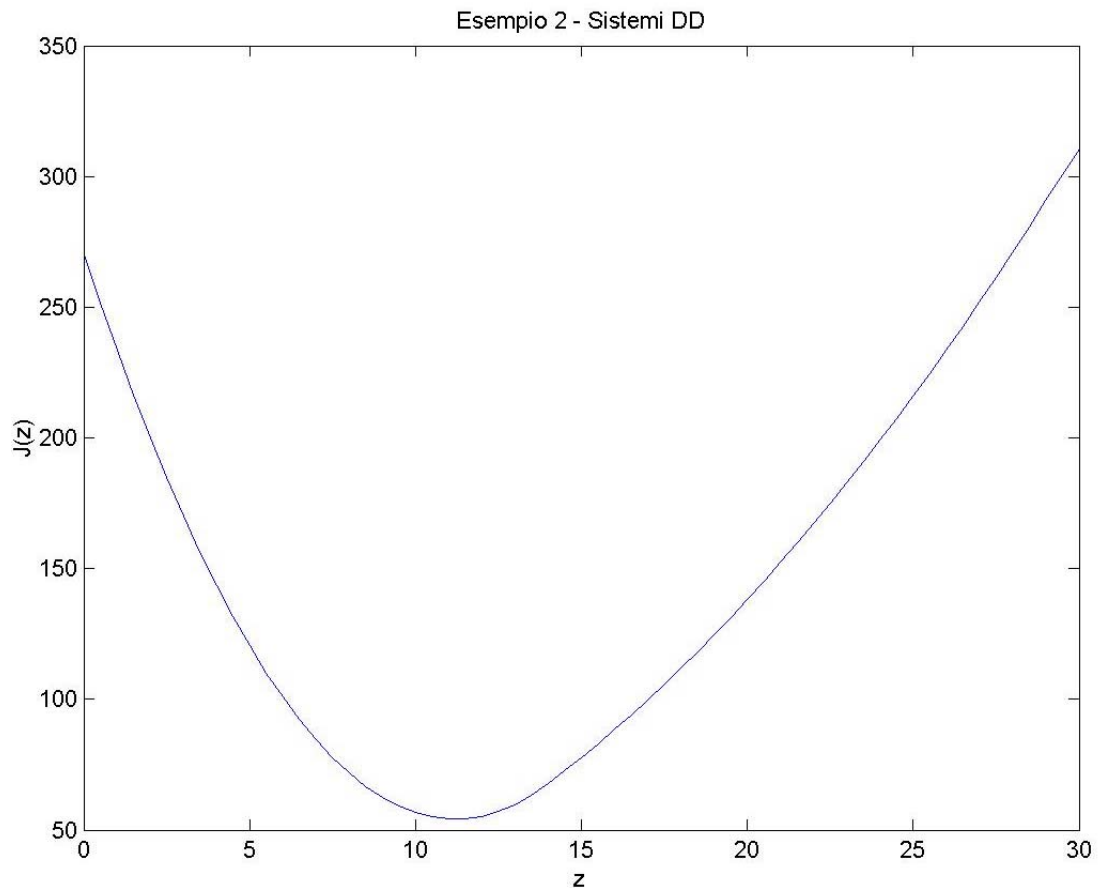
Valore di COSTO MINIMO atteso: 54.1935

.....

otteniamo un errore di calcolo della scorta ottima sempre minimo:

“Il valore minimo di costo trovato è: 54.2394 corrispondente allo z_{star} : 11”

con un grafico di $J(z)$



Esempio 2.3

Analogamente per un sistema con alto margine di stabilità

PARAMETRI DI LAVORO DELL'ALGORITMO

Potenza max dell'impianto μ : 3

Tasso di domanda costante d : 1

c_m : 80

c_p : 20

Tempo di guasto (t_g): 5

Tempo di funzionamento (t_f): 150

Fattore di stabilità del sistema S : 295

Passo di t_{star} (istante di inizio del lavoro a potenza max): 1

Passo di z_{star} (massima scorta dell'impianto): 0.5

Valore di SCORTA OTTIMA atteso: 4

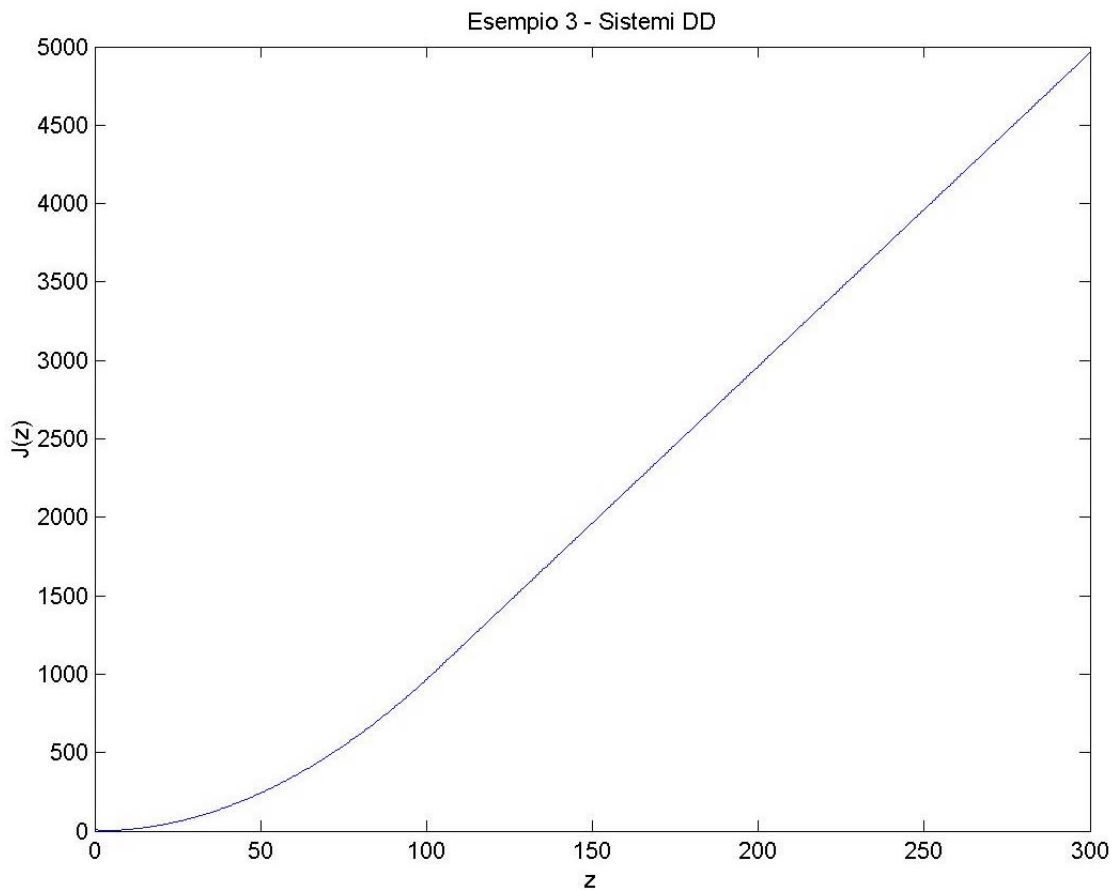
Valore di COSTO MINIMO atteso: 1.93548

.....

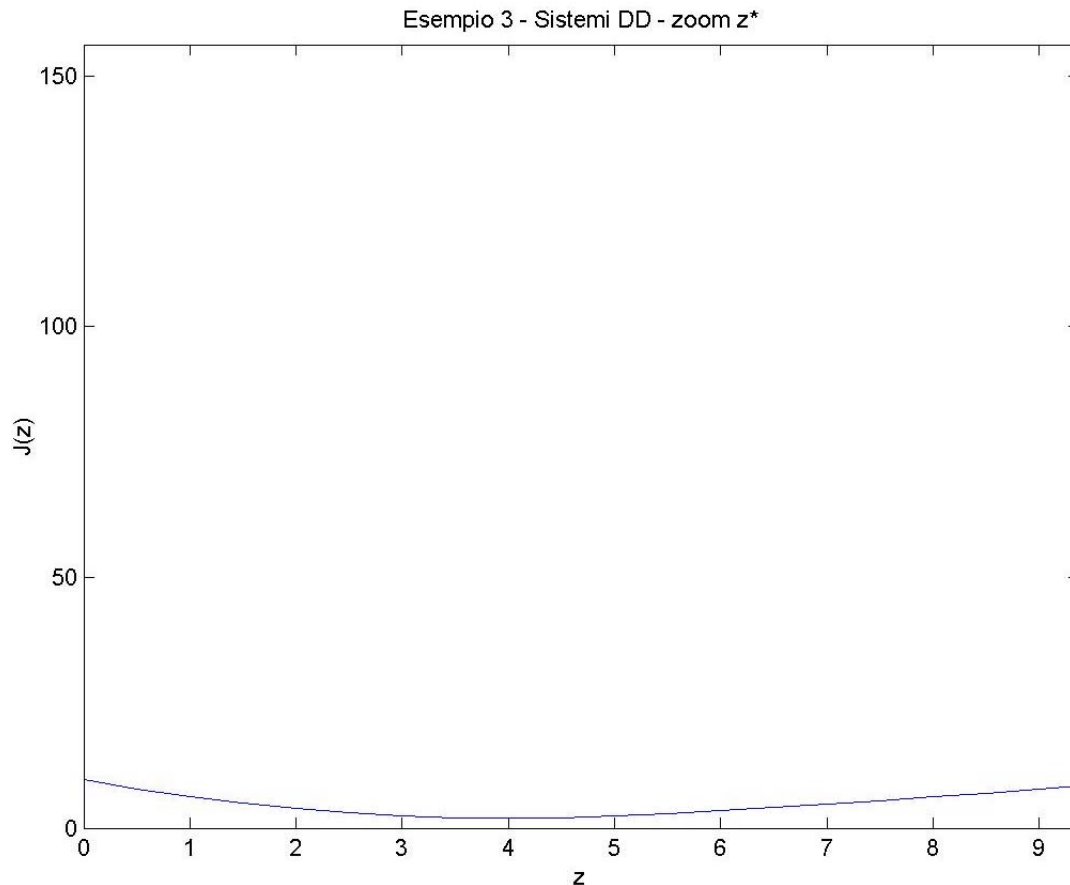
Si ottiene:

“Il valore minimo di costo trovato è: 1.93523 corrispondente allo z_{star} : 4”

con il grafico corrispondente di $J(z)$



che nel punto di minimo appare



Esempio 2.4

Lavorando invece sui parametri di costo per il sistema mostrato nell'esempio 1 possiamo verificare le supposizioni fatte studiando il grafico generale della $J(z)$.
Con $c_m \gg c_p$:

PARAMETRI DI LAVORO DELL'ALGORITMO

Potenza max dell'impianto μ : 3

Tasso di domanda costante d : 2

c_m : 150

c_p : 5

Tempo di guasto (t_g): 5

Tempo di funzionamento (t_f): 150

Fattore di stabilità del sistema S : 140

Passo di t_{star} (istante di inizio del lavoro a potenza max): 1

Passo di z_{star} (massima scorta dell'impianto): 0.5

Valore di SCORTA OTTIMA atteso: 9.67742

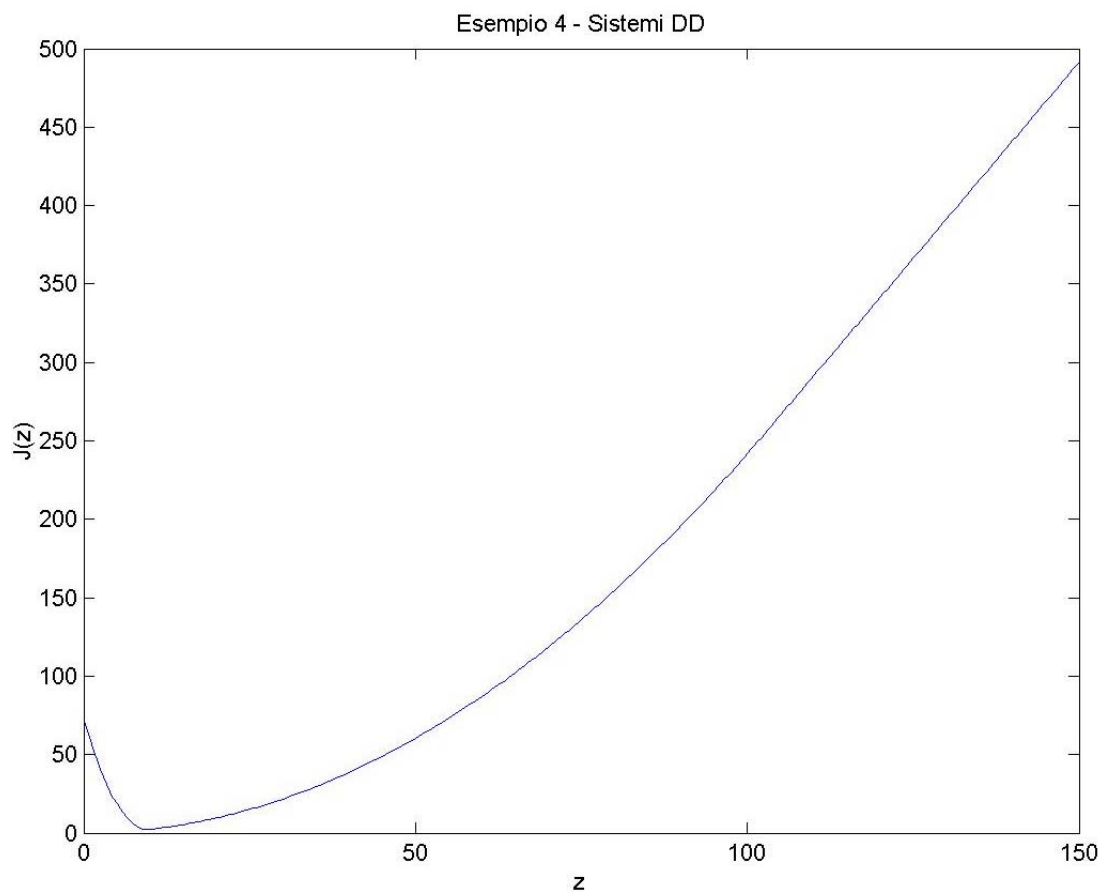
Valore di COSTO MINIMO atteso: 2.34131

.....

otteniamo

Il valore minimo di costo trovato è: 2.36468 corrispondente allo z^ : 9.5*

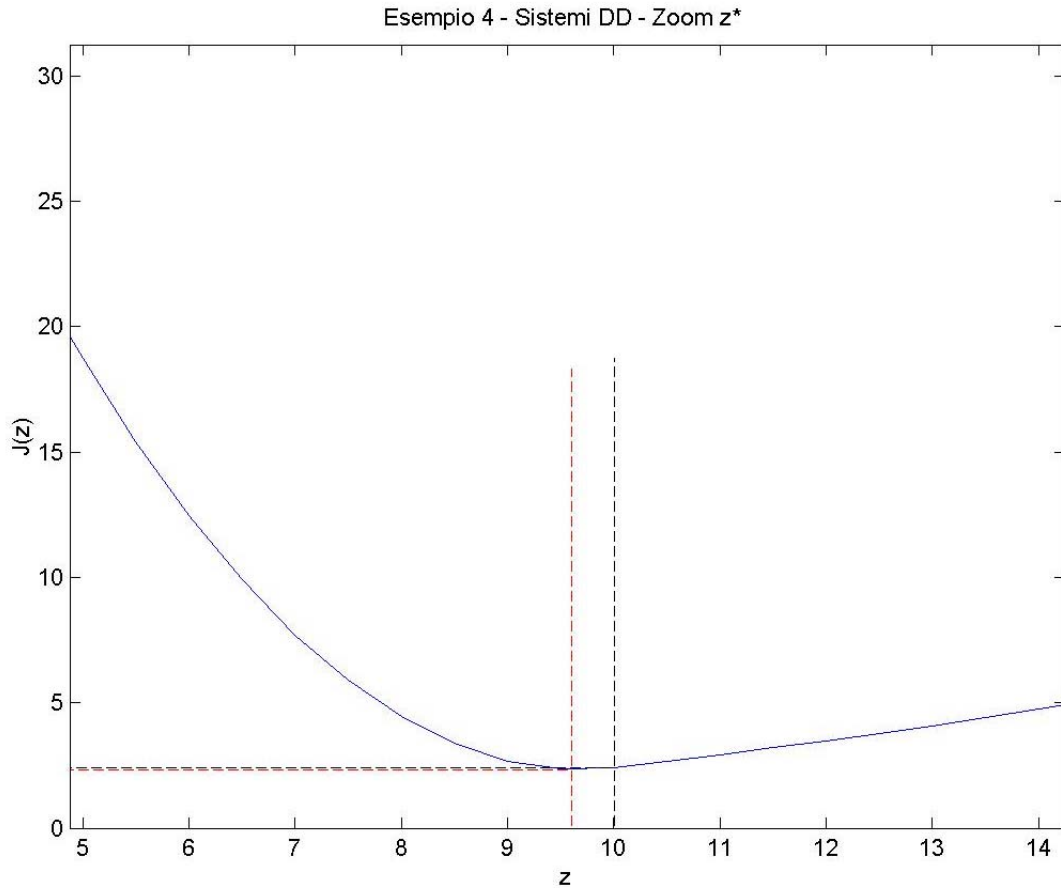
e il grafico corrispondente



da cui si può vedere nel dettaglio che

$$z^* \rightarrow dT_g$$

$$J(z^*) \rightarrow J(dT_g)$$



Esempio 2.5

Per la scelta opposta di costo, cioè $c_p \gg c_m$ otteniamo risultati analogamente immaginabili

PARAMETRI DI LAVORO DELL'ALGORITMO

Potenza max dell'impianto μ : 3

Tasso di domanda costante d : 2

c_m : 20

c_p : 150

Tempo di guasto (t_g): 5

Tempo di funzionamento (t_f): 150

Fattore di stabilità del sistema S : 140

Passo di t_{star} (istante di inizio del lavoro a potenza max): 1

Passo di z_{star} (massima scorta dell'impianto): 0.5

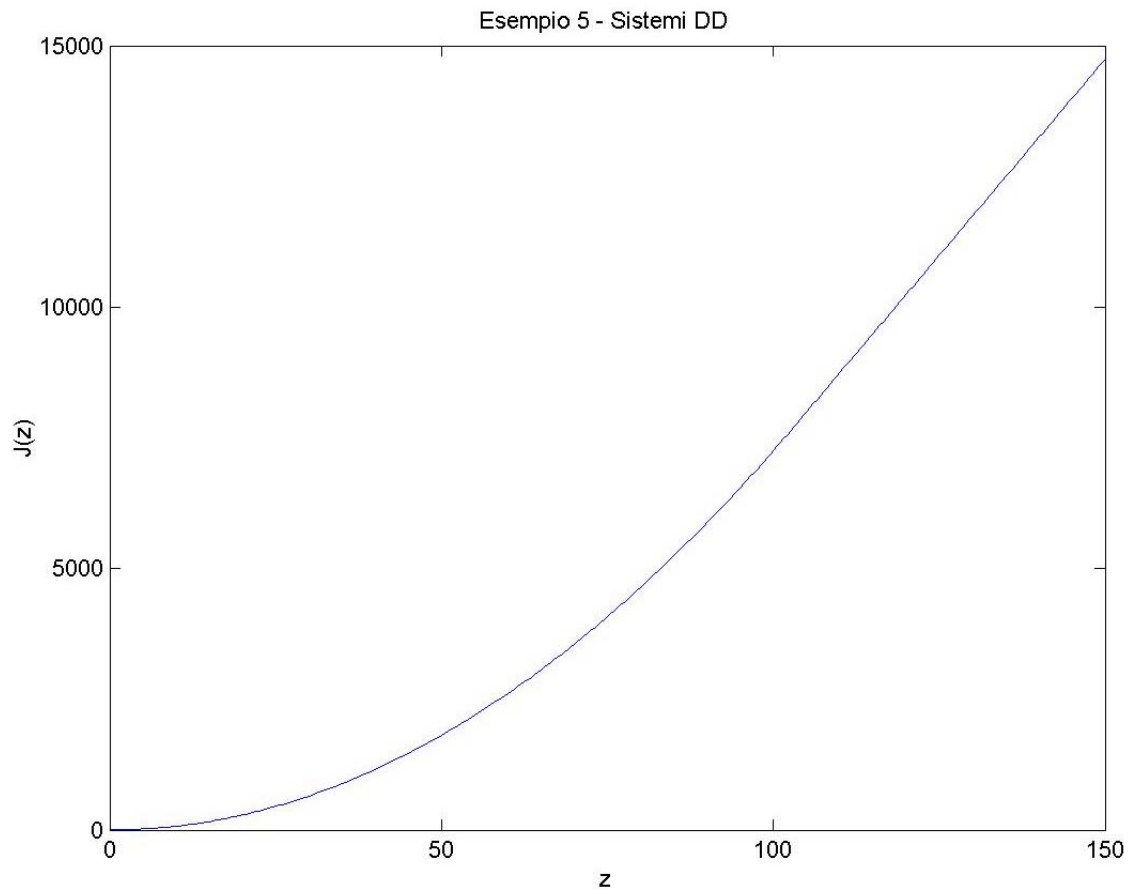
Valore di SCORTA OTTIMA atteso: 1.17647

Valore di COSTO MINIMO atteso: 8.5389

.....

“Il valore minimo di costo trovato è: 8.55406 corrispondente allo z^ : 1”*

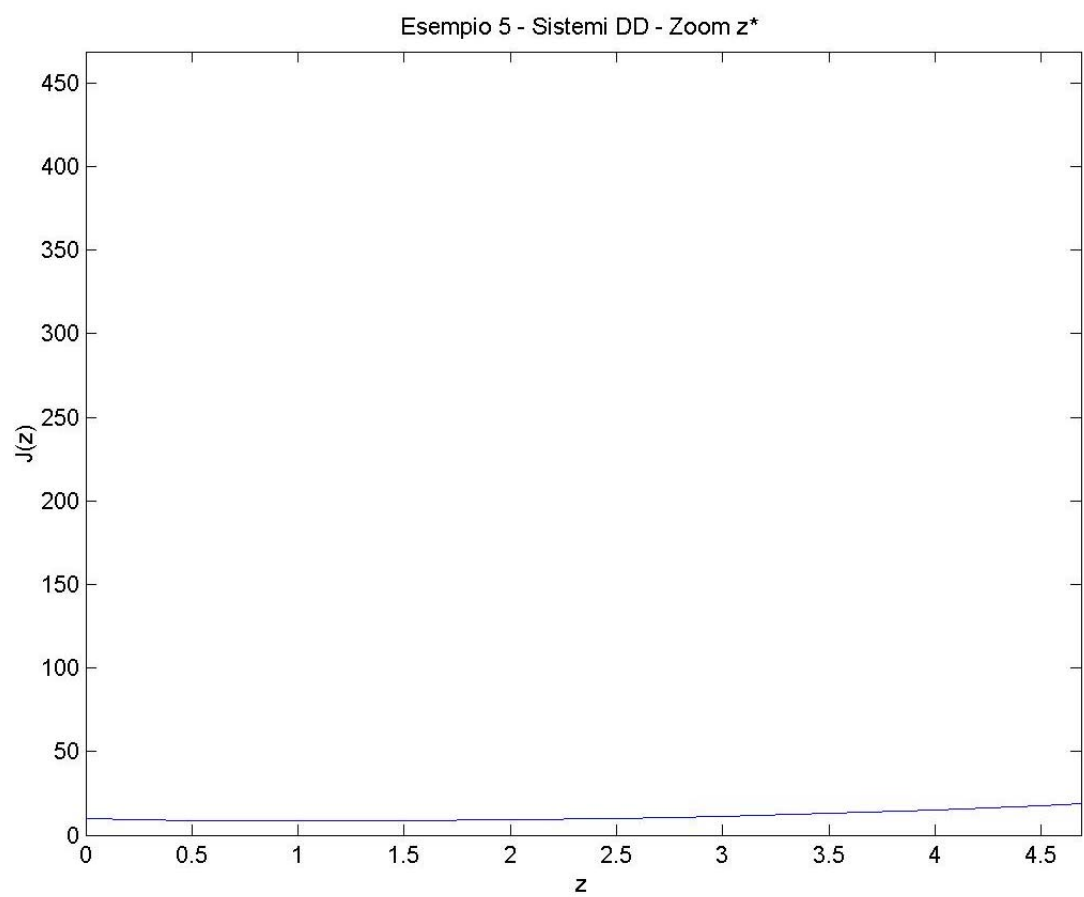
Il grafico di $J(z)$ mostra l’attendibilità delle supposizioni fatte in sede di analisi teorica



Si può vedere dal seguente dettaglio che

$$z^* \rightarrow 0$$

$$J(z^*) \rightarrow J(0)$$



CAPITOLO 3 - Tempo di funzionamento uniforme e tempo di guasto deterministico

3.1 Analisi generale

Per i sistemi del tipo U/D/1, cioè con un tempo di guasto costante e la cui densità di probabilità del tempo di funzionamento è uniforme tra due estremi, chiamati rispettivamente $T_{funz_{in}}$ e $T_{funz_{fin}}$, si è cercato di simulare un comportamento a regime senza poter risolvere analiticamente il problema di minimizzazione della funzione di costo.

Lavoriamo su un sistema manifatturiero a macchina singola, con domanda d costante e tasso di produzione $0 \leq u_t \leq \alpha_t \mu$ con α_t stato del sistema (0 se è guasta, 1 se è in funzione).

La funzione di costo è lineare a tratti e definita come $g(x) = c_p x^+ + c_m x^-$ con c_p e c_m costanti non negative corrispondenti rispettivamente a scorta positiva e negativa.

Perciò viene definita la funzione

$$J = \frac{1}{T} E \left[\int_0^T (c_p x^+ + c_m x^-) dt \right] \quad (71)$$

che sarà oggetto delle simulazioni.

Avendo ad ogni ciclo di funzionamento/guasto un valore diverso e casuale di T_f il calcolo di $J(z)$ risulta più complesso del caso precedente a regime. La condizione necessaria per la stabilità è ora

$$(\mu - d)\bar{T}_f - dT_g > 0 \quad (72)$$

con il valore medio del tempo di funzionamento dato da

$$\bar{T}_f = \frac{T_{funz_{in}} + T_{funz_{fin}}}{2} \quad (73)$$

che verrà spesso usato nei calcoli numerici come approssimazione a regime di T_f .

Definiamo t^* l'istante in cui il sistema inizia a produrre a potenza massima per raggiungere il valore z di scorta. La nostra scelta è stata generica proprio per la natura puramente simulativa dello studio effettuato sui sistemi U/D/1. Per questo sono stati implementati due algoritmi diversi: nel primo è stata effettuata la stessa scelta enunciata nel capitolo 2 dalla (35) per i sistemi D/D/1; nell'altro algoritmo è stato lasciato t^* come parametro indipendente da z^* e perciò con la possibilità di valutare graficamente la differenza con una scelta diversa rispetto a quella finora enunciata per gli altri casi. E' comunque necessario

$$0 \leq t^* \leq t_{funz_{fin}} \quad (74)$$

Fa parte delle approssimazioni effettuate per rendere migliore l'analisi sul sistema anche il fatto che si prenderanno in considerazione solo

$$t^* \leq \bar{T}_f \quad (75)$$

scelta che garantisce un livello di scorta positivo. Difatti se non fosse verificata la (75) non sarebbe possibile raggiungere un valore non nullo.

Se vale la (75) otteniamo che

$$z^* \leq (\mu - d)(\bar{T}_f - t^*) \quad (76)$$

mentre valori maggiori non sono raggiungibili.

Con queste ipotesi non si trova la politica ottima ma si valutano euristiche derivate dall'analisi di sistemi U/D/1 con i corrispondenti tempi di funzionamento assimilati al loro valor medio.

Per effettuare valutazioni aggiuntive sul sistema conoscendo i suoi parametri è stato fissato il valore di stabilità già definito nella (42) ma considerando i tempi medi di funzionamento:

$$S := \bar{T}_f(\mu - d) - T_g d \quad (77)$$

3.2 Possibili andamenti di $J(z)$ nel tempo di funzionamento

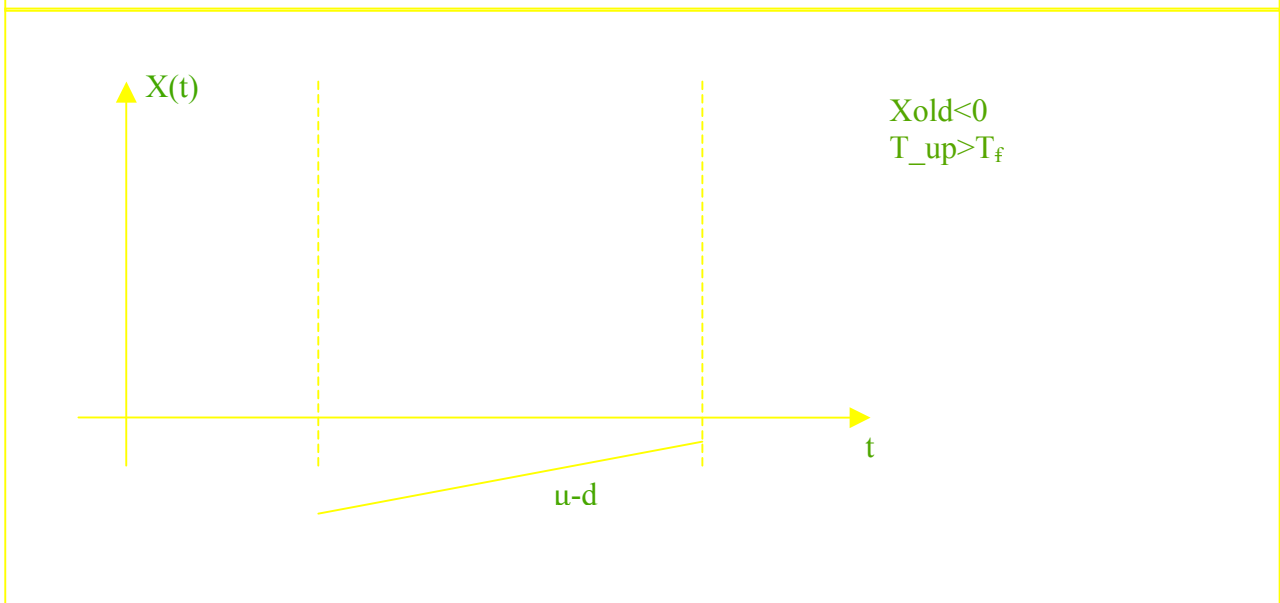
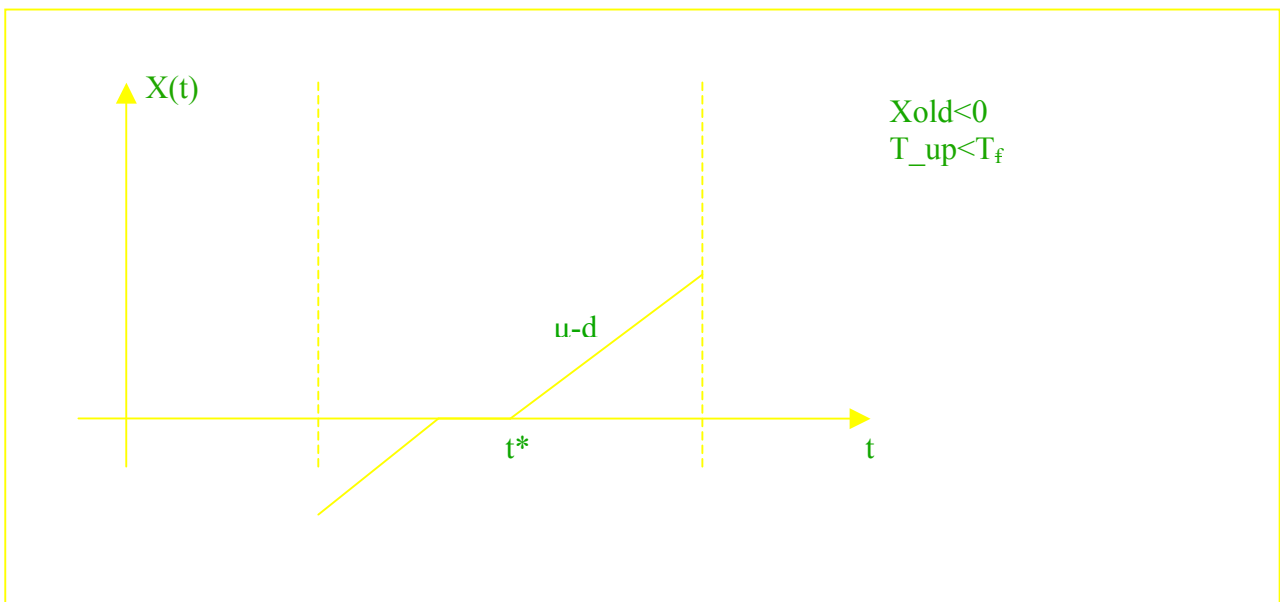
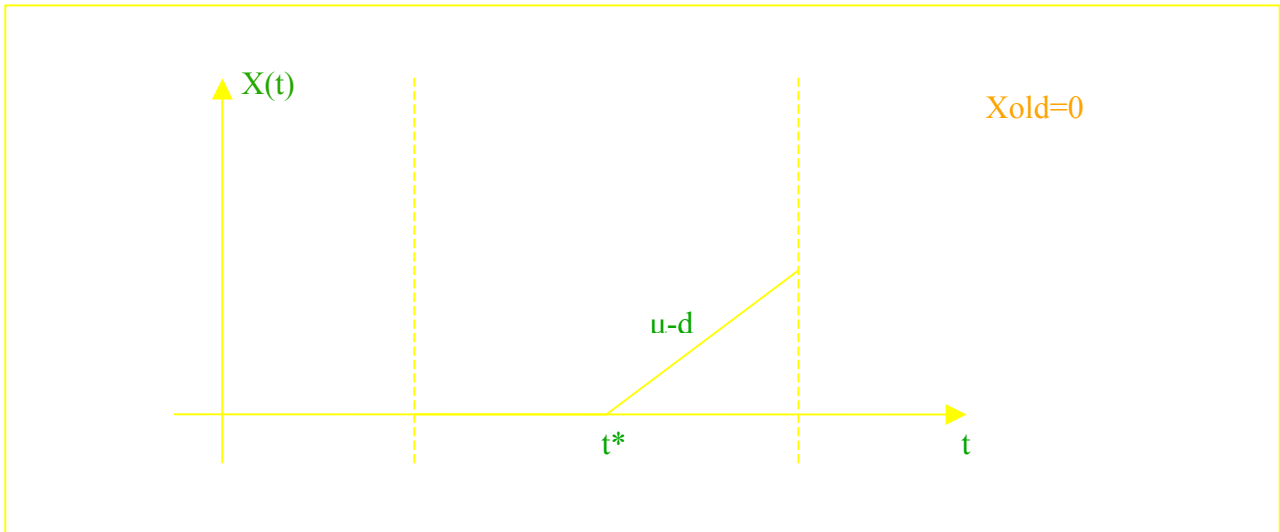
Visualizziamo ora la sequenza dei vari comportamenti possibili in un ciclo di funzionamento tralasciando i calcoli sul costo implementati nell'algoritmo che sono semplicemente il calcolo delle rispettive aree evidenziate. Con X_{old} e X_{new} sono indicati il valore iniziale e finale della scorta rispetto a T_f , mentre identifichiamo con t_{up} e t_{down} i tempi che impiega il buffer a portarsi al valore nullo rispettivamente da valori negativi e positivi che assume all'inizio del tempo di funzionamento.

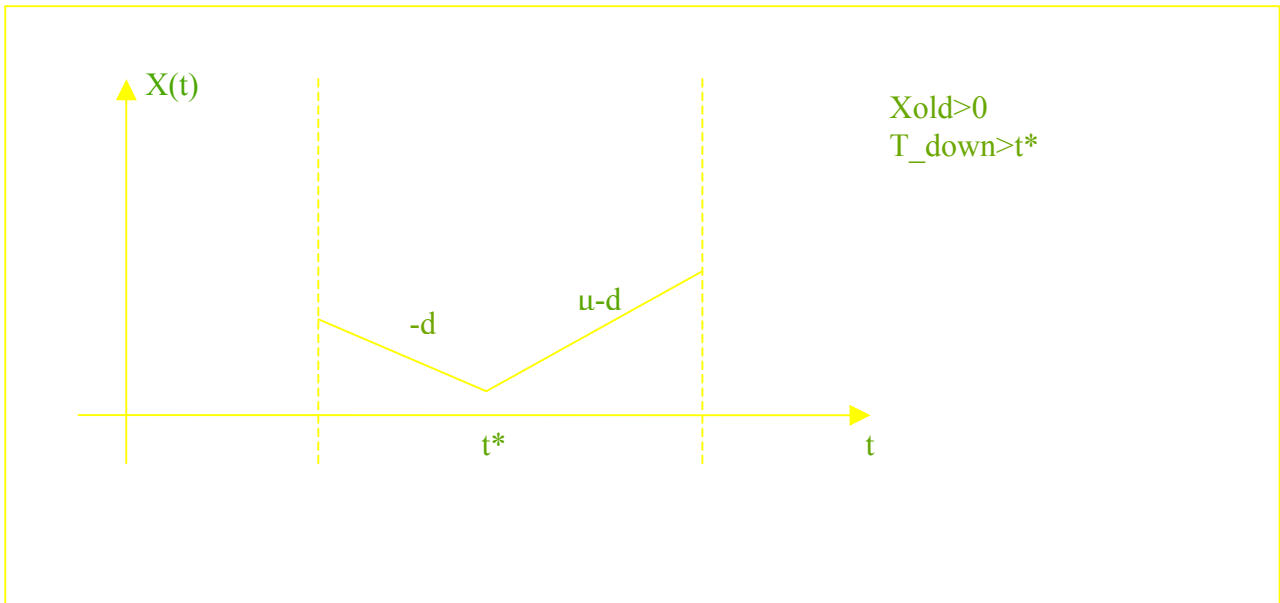
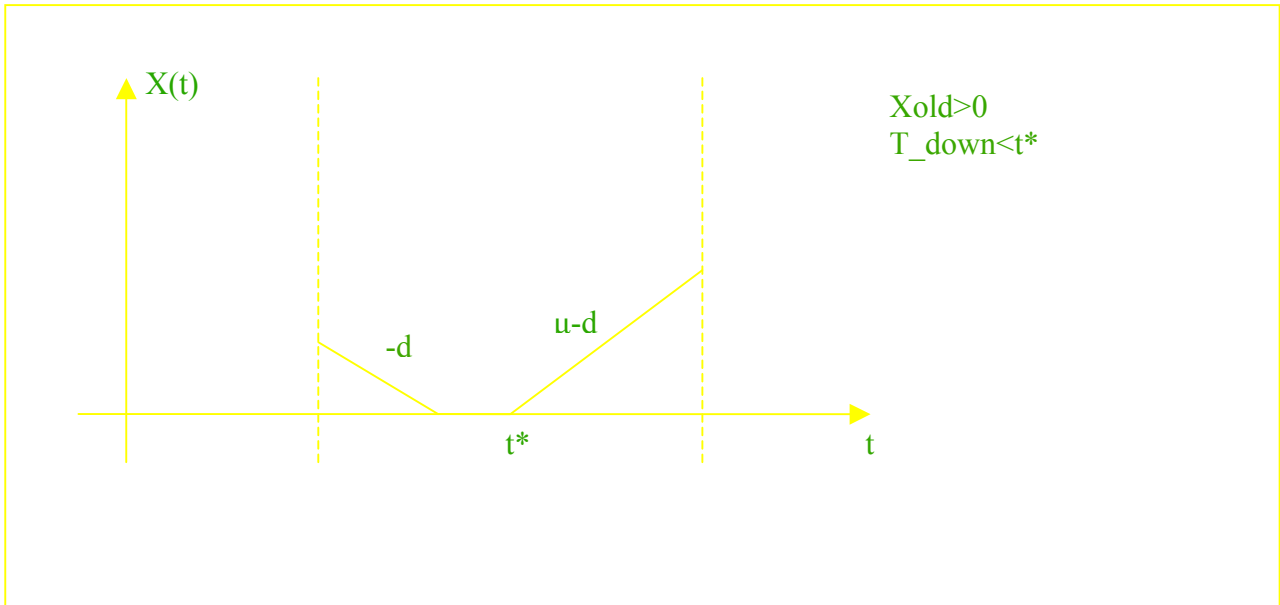
Perciò

$$T_{up} = X_{old}/(\mu - d)$$

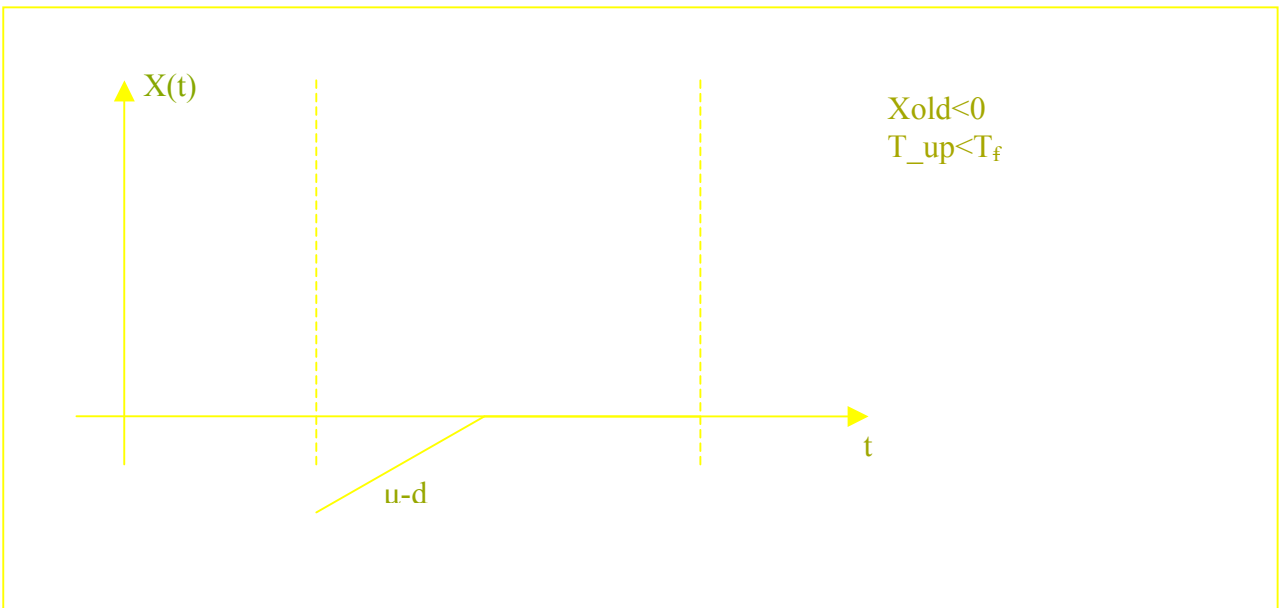
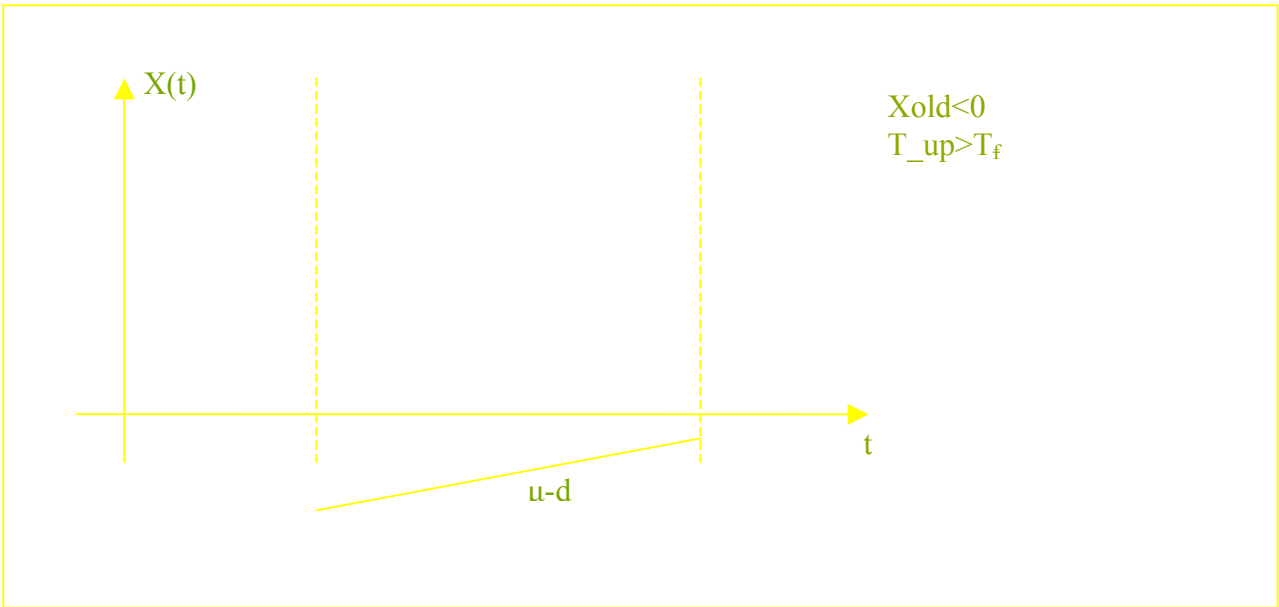
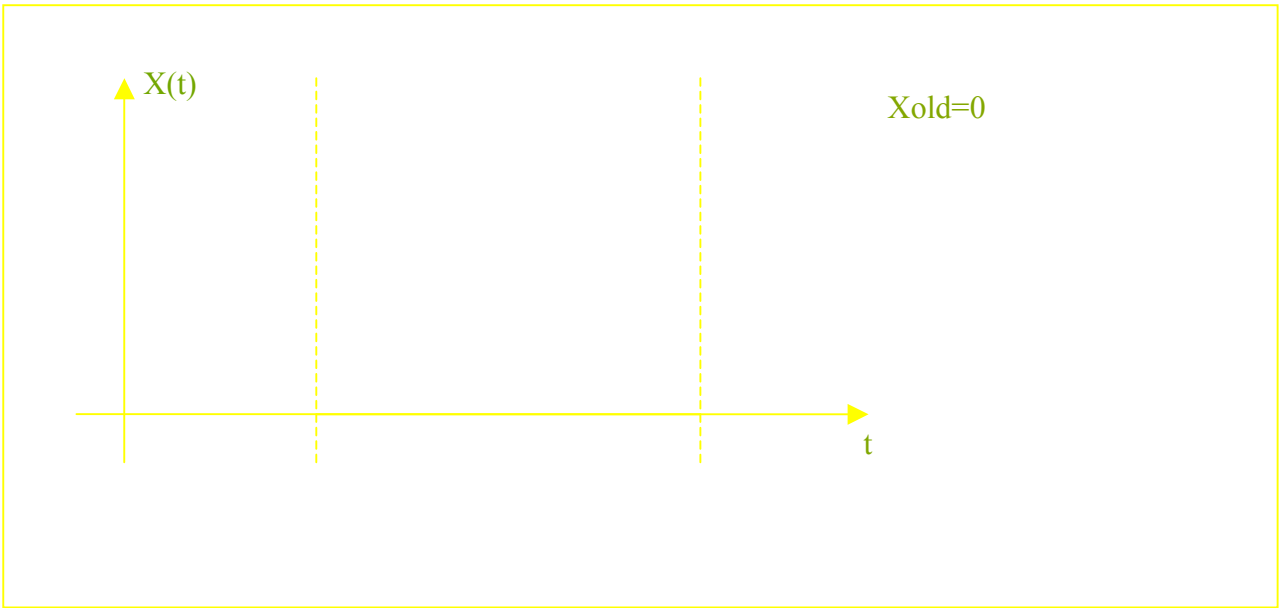
$$T_{down} = X_{old}/d$$

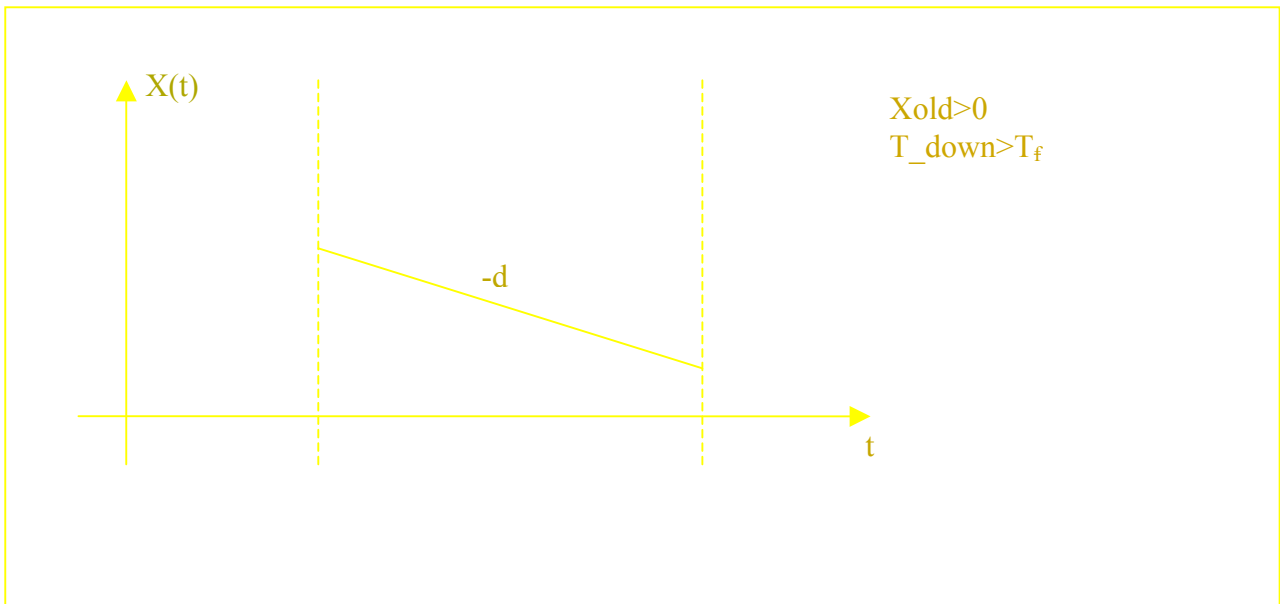
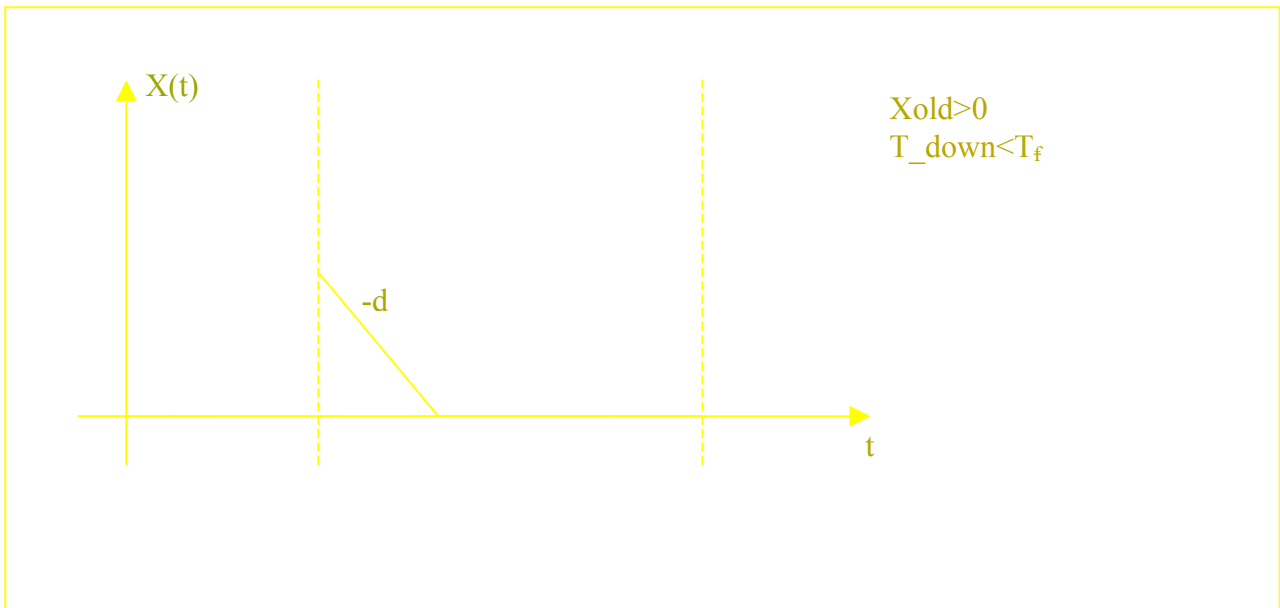
- Se $T_f > t^*$





- Se, invece, $T_f < t^*$





3.3 Simulazioni algoritmiche di sistemi U/D/1

I due algoritmi in C++ cui si accennava nel precedente paragrafo hanno una struttura generale molto simile a quello utilizzato per le simulazioni di sistemi D/D/1, se non nell'aggiunta di un parametro N equivalente al numero di simulazioni da effettuare per ogni z^* : difatti la natura aleatoria della probabilità del tempo di funzionamento induce a considerare un valor medio dei costi trovati in N iterazioni. I parametri costanti rimangono quelli di produzione (μ, d) , di costo (c_p, c_m) e di tempo $(t_{funz_{in}}, t_{funz_{fin}}, t_g)$ impostati dall'utente. Nel primo algoritmo “ t^*_{fisso} ” avviene lo stesso processo simulativo dell'algoritmo DD, con una scansione lungo tutti gli z assegnabili mantenendo t^* vincolato a z secondo la politica illustrata nel capitolo 2 e un salvataggio dei dati di costo medio da graficare in Matlab. Il secondo algoritmo “ $t^*_{variabile}$ ”, invece, cerca di analizzare in maniera generale il problema, scandendo

non solo gli z ma anche i vari t^* assegnabili e visualizzando la scelta migliore di questo parametro per ogni z in base al costo medio minimo. E' perciò un algoritmo che riassume i risultati su un grafico non esatto analiticamente in quanto relativo ai costi minimi separati, non necessariamente appartenenti alla stessa scelta di t^* , ma indicativo per valutare le politiche di costo scelte. Sullo stesso grafico sono inoltre presenti le funzioni che contemplano la varianza dei valori di costo analizzati e che saranno confrontate con la funzione di costo media.

Esempio 3.1

Analizziamo un sistema di media stabilità come il 2.1

PARAMETRI DI LAVORO DELL'ALGORITMO

Potenza max dell'impianto μ : 3

Tasso di domanda costante d : 2

c_m : 80

c_p : 20

Tempo di guasto (t_g): 5

Tempo di funzionamento (t_f): [100,200]

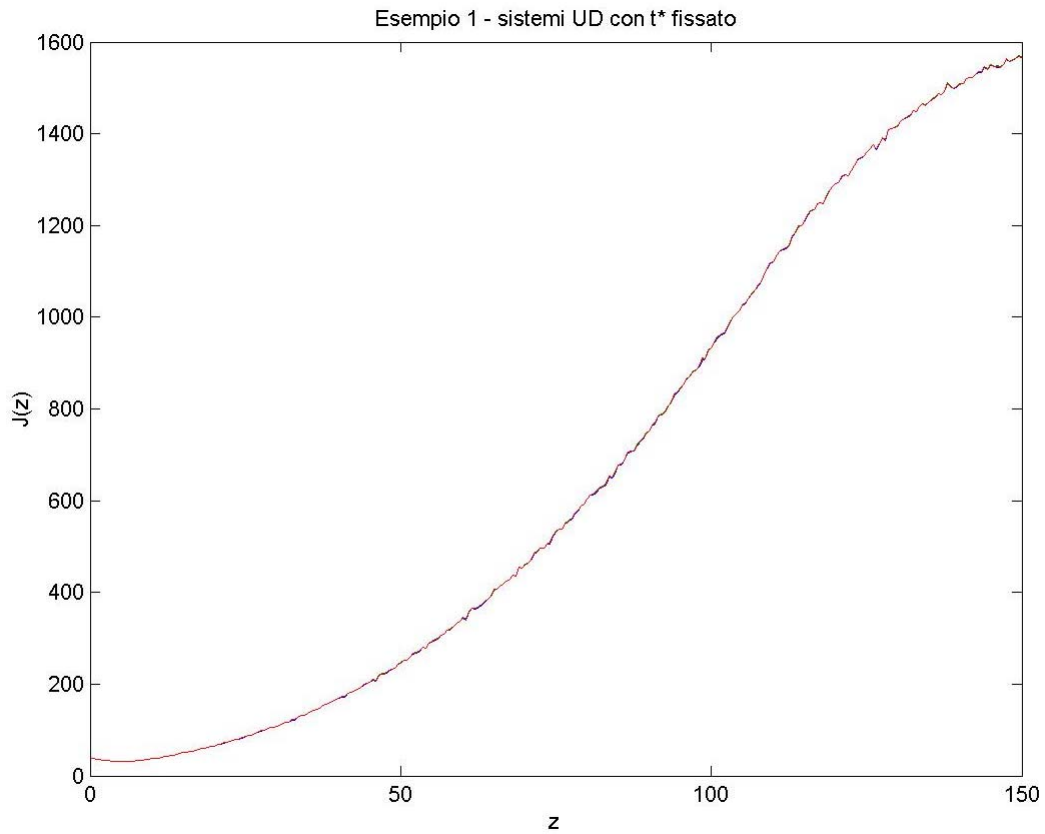
Fattore di stabilità del sistema S : 140

Passo di t_{star} (istante di inizio del lavoro a potenza max): 1

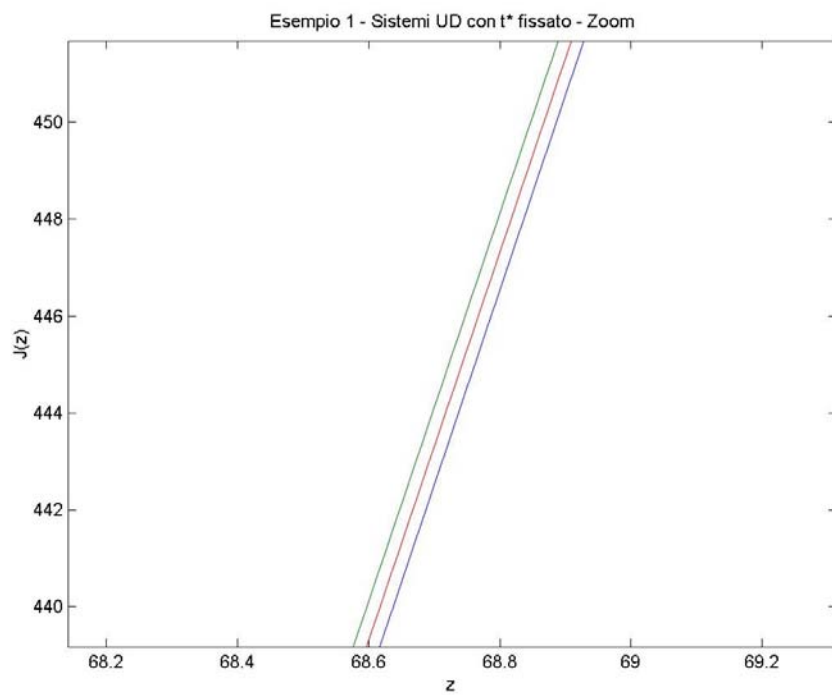
Passo di z_{star} (massima scorta dell'impianto): 0.5

Si può vedere che il tempo medio di funzionamento corrisponde al valore 150 che era il tempo deterministico nell'esempio 2.4.1. Utilizzeremo proprio questa analogia per confrontare i sistemi. Nel caso t^* _fisso:

Il valore minimo di costo trovato è: 31.845 corrispondente allo z_{star} : 5.5 e cioè al t_{star} 144.5



in cui utilizzando uno zoom possiamo evidenziare l'errore minimo tra il costo medio e i valori di deviazione standard:

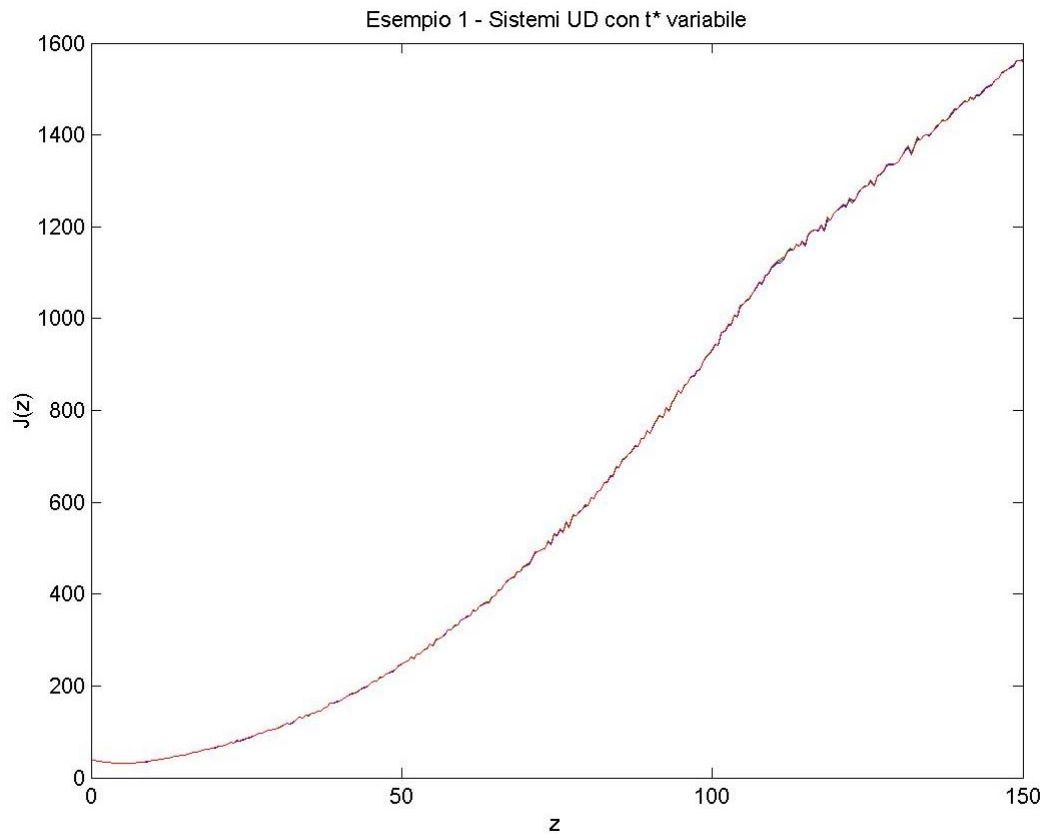


mentre con t^* _variabile:

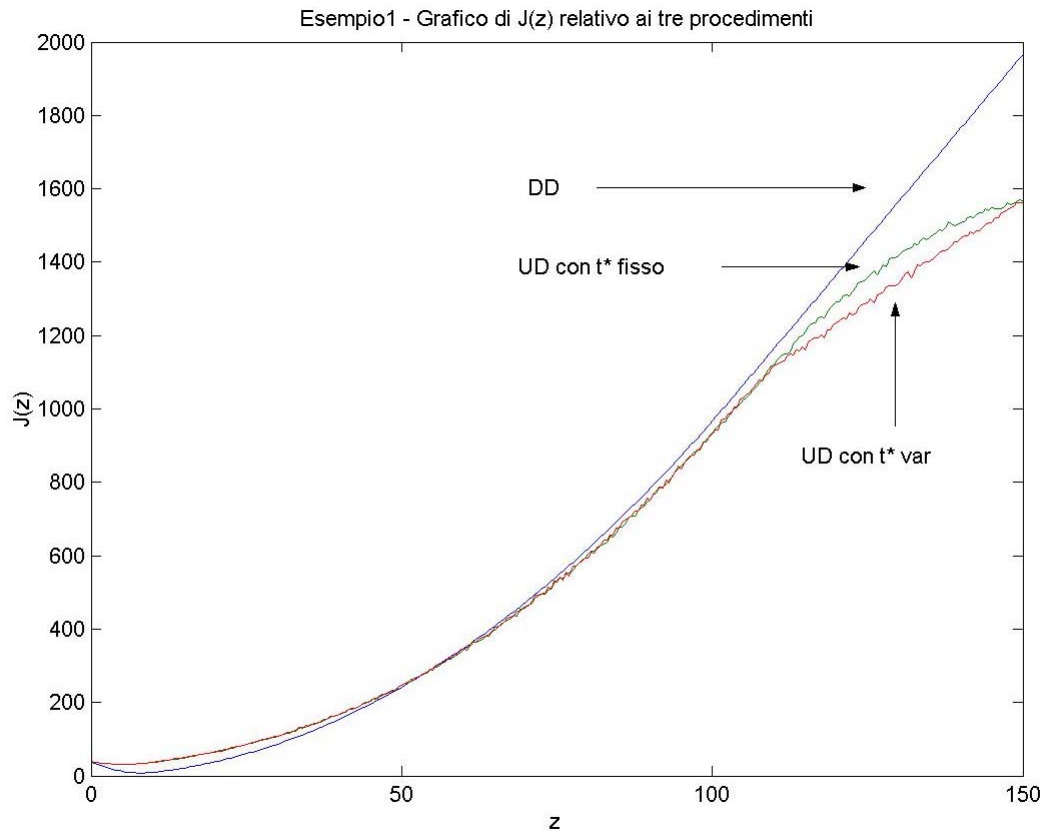
Il valore minimo di costo trovato è: 31.5407 corrispondente allo z^ : 5 e cioè*

al tstar 145

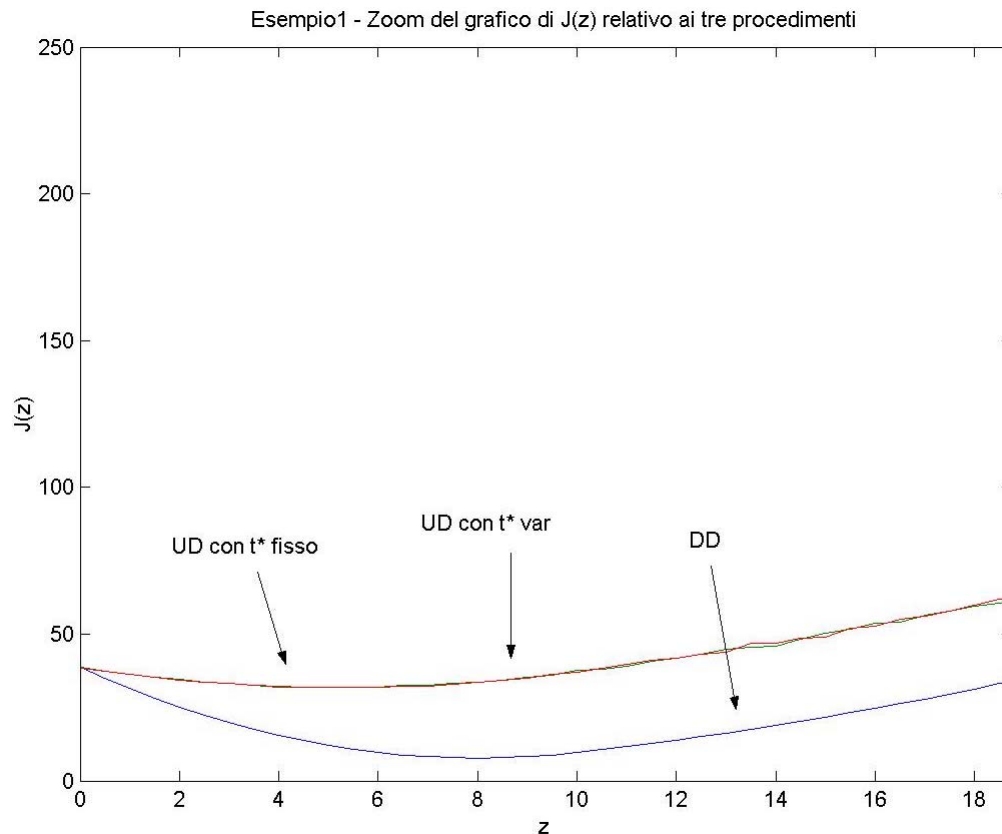
Valutandone il grafico evidenziamo anche qui l'errore minimo, specialmente per valori bassi di scorta, tra costo medio riportato e possibile errore di misura, analisi che d'ora in poi lasceremo al lettore attraverso il grafico di $J(z)$.



Già numericamente possiamo notare una somiglianza tra l'utilizzo della politica ottima per i D/D/1 e questi due grafici, sia con t^* fisso che con una visione più generale dei costi ottimi. Possiamo visualizzare questa differenza in:



dove si distinguono i grafici relativi a DD, UD con t^* fisso e UD con t^* variabile. Se visualizziamo la zona dove sono presenti i minimi delle funzioni vediamo che il caso DD si distacca per valori dal caso UD, anche se presenta un medesimo andamento di $J(z)$, sia per il costo minimo che per la scorta ottima, ma vediamo che il caso generale di UD con t^* variabile rientra nel caso particolare di UD con t^* dipendente da z^* , che possiamo dedurre quindi ottimo.



Esempio 3.2

Procedendo allo stesso modo per un sistema con margine di stabilità molto inferiore, del tipo

PARAMETRI DI LAVORO DELL'ALGORITMO

Potenza max dell'impianto μ : 3

Tasso di domanda costante d : 2.8

c_m : 80

c_p : 20

Tempo di guasto (t_g): 5

Tempo di funzionamento (t_f): [100,200]

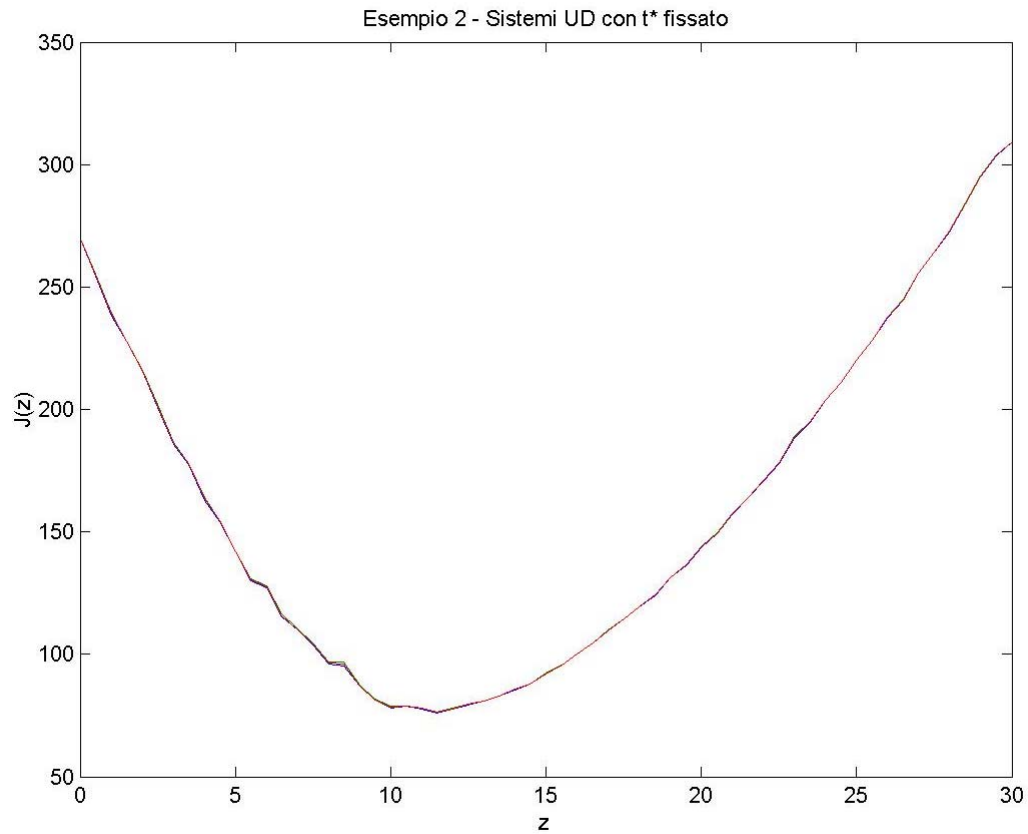
Fattore di stabilità del sistema S : 16

Passo di t_{star} (istante di inizio del lavoro a potenza max): 1

Passo di z_{star} (massima scorta dell'impianto): 0.5

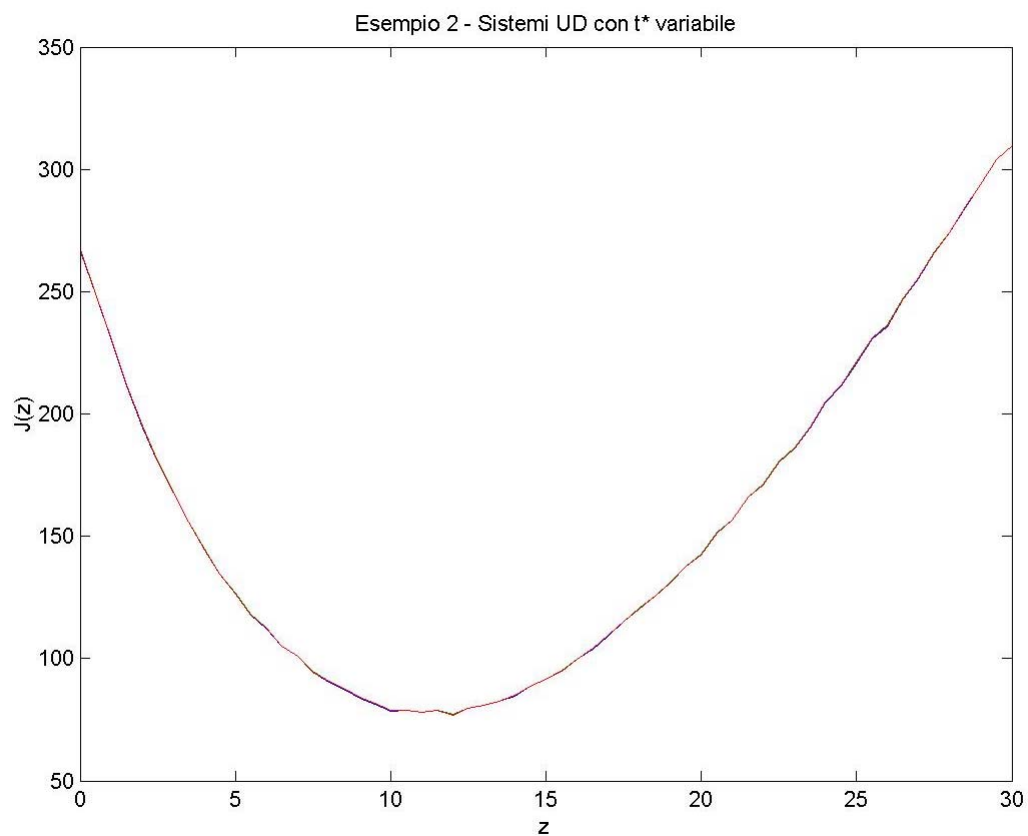
Si ottiene con t^* fisso:

Il valore minimo di costo trovato è: 76.475 corrispondente allo z^ : 11 e cioè al t^* 95*

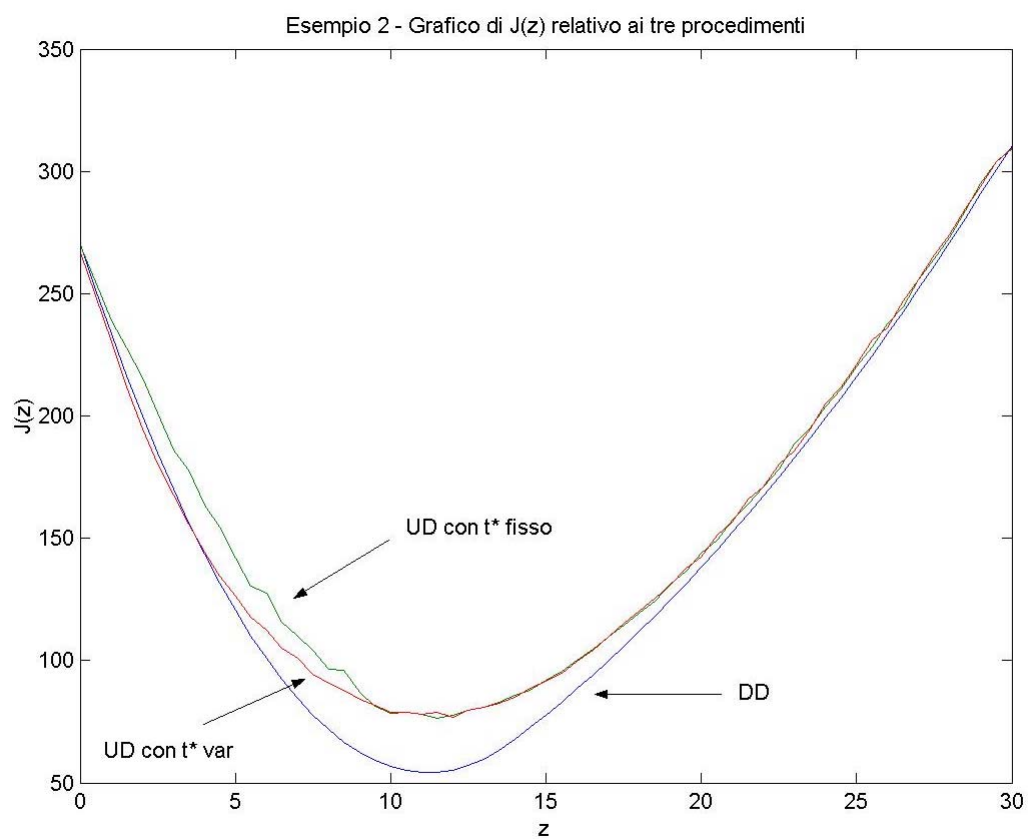


in cui se non con un ottimo ingrandimento non si avverte l'errore di approssimazione mentre con t^* variabile si ottiene

Il valore minimo di costo trovato è: 78.2874 corrispondente allo z^ : 10.5 e cioè al t^* 97.5*



Le tre funzioni sovrapposte appaiono nel grafico seguente



Esempio 3.3

Per il sistema con un ampio margine di stabilità del tipo dell'esempio 2.3

PARAMETRI DI LAVORO DELL'ALGORITMO

Potenza max dell'impianto μ : 3

Tasso di domanda costante d : 1

c_m : 80

c_p : 20

Tempo di guasto (t_g): 5

Tempo di funzionamento (t_f): [100,200]

Fattore di stabilità del sistema S : 295

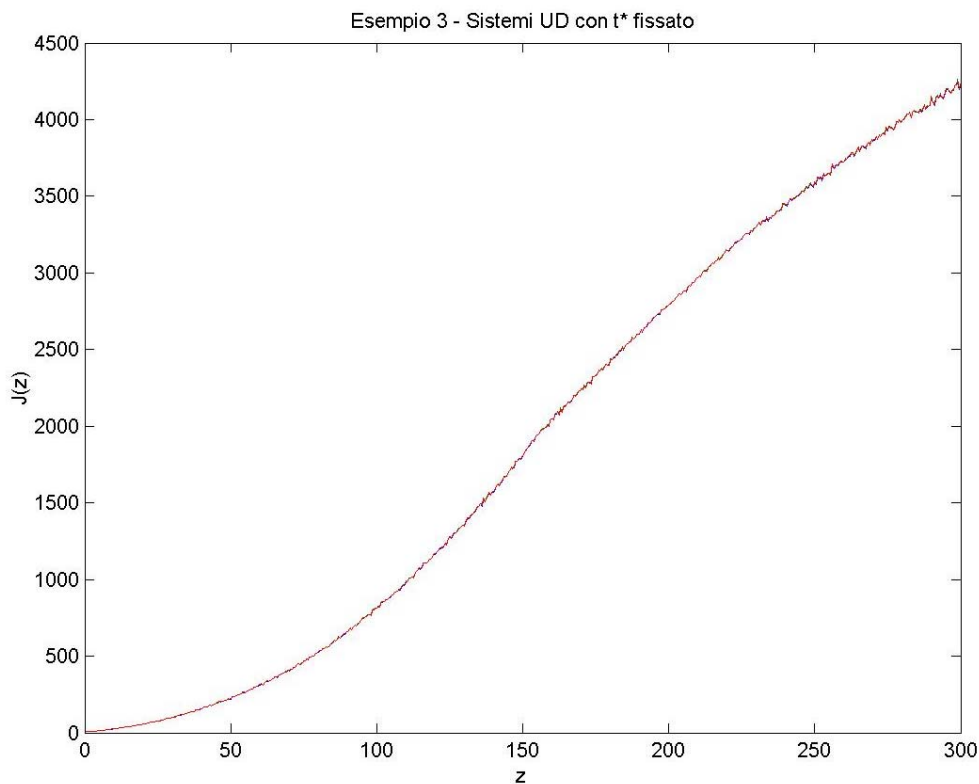
Passo di t_{star} (istante di inizio del lavoro a potenza max): 1

Passo di z_{star} (massima scorta dell'impianto): 0.5

.....

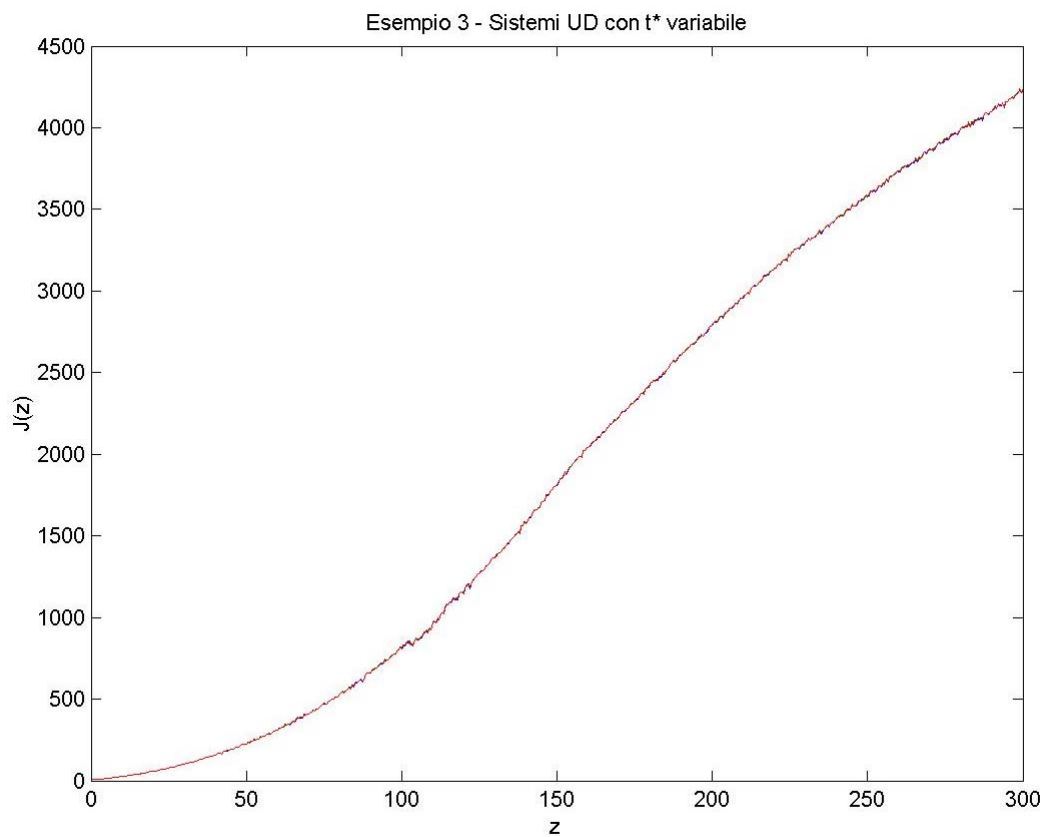
Si ottiene mantenendo fisso t^* che

Il valore minimo di costo trovato è: 9.46474 corrispondente allo z_{star} : 1 e cioè al t_{star} 149.5

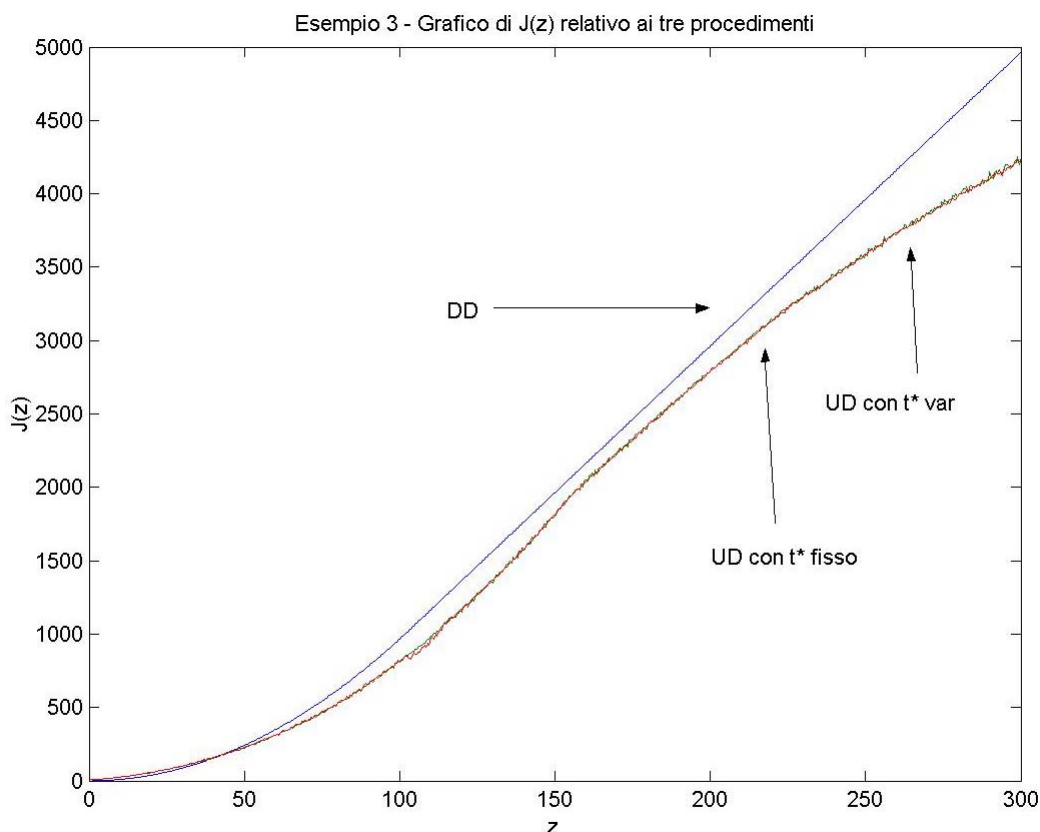


mentre per t^* variabile:

Il valore minimo di costo trovato è: 9.53723 corrispondente allo z^ : 0.5 e cioè al t^* 149.75*



e come nei casi precedenti dal grafico riassuntivo vediamo la parziale sovrapposizione delle tre funzioni di costo almeno per valori bassi di scorta.



3.4 Osservazioni sugli esempi

Gli esempi mostrati sono solo una parte di quelli effettivamente studiati in sede di analisi e i casi più rappresentativi inseriti in questo lavoro possono ricondurre a comportamenti generali per questo tipo di sistemi. Innanzitutto dal confronto con i grafici relativi alla politica utilizzata per i sistemi D/D/1 si può vedere che la differenza esistente sia per il valore di scorta ottima che per il costo minimo possibile non è così ampia da ritenere inutile l'applicazione di tale politica a sistemi U/D/1. Infatti, nonostante il costo risulti sempre sensibilmente inferiore nel caso deterministico, non possiamo certo trascurare il fatto che studiando tutte le possibili scelte di t^* , senza vincoli rispetto a z , la scelta che sembra avvicinarsi molto a quella ottima è quella analizzata in particolare con l'algoritmo che mantiene t^* dipendente strettamente da z secondo il legame definito per la politica applicata ai sistemi D/D/1. Possiamo perciò affermare che scegliere anche per sistemi U/D/1 un t^* strettamente vincolato a z può essere una scelta molto buona.

CONCLUSIONI E SVILUPPI FUTURI

E' stato perciò raggiunto l'obiettivo posto all'inizio del lavoro di applicare una politica e di trovare il valore ottimo di scorta corrispondente per sistemi completamente deterministici. Il grafico qualitativo di $J(z)$ per questo tipo di sistemi e l'estrapolazione di un valore esatto da assegnare alla scorta per minimizzare i costi, con il possibile riscontro grafico delle simulazioni, è un risultato che può essere utilizzato per studi futuri più approfonditi sull'argomento.

Per quanto riguarda l'altra classe di sistemi analizzati, del tipo U/D/1, dove la densità di probabilità del tempo di funzionamento è uniforme, sono state effettuate considerazioni diverse, basate prima sull'analisi delle situazioni possibili quando la macchina è in servizio, in seguito attraverso grafici simulativi. Questo tipo di analisi ha evidenziato che la politica che abbiamo considerato per i sistemi D/D/1 è certamente una tra quelle candidate ad essere ottime per i sistemi U/D/1, per i quali si è valutato più genericamente anche il caso generale di t^* indipendente.

Quest'ultima considerazione è importantissima per eventuali sviluppi futuri in quanto consente di ripartire per una valutazione stavolta analitica e non più unicamente simulativa dei sistemi U/D/1 fissando la politica enunciata nel capitolo 2, e cercando di arrivare alla determinazione del valore ottimo di scorta, come è stato fatto nel caso deterministico.

Inoltre è possibile applicare gli stessi ragionamenti per diverse distribuzioni di probabilità del tempo di funzionamento e di guasto, e si potrebbero analizzare nello specifico sistemi del tipo N/D/1 (con t_f di tipo gaussiano), N/N/1 (con entrambe le distribuzioni di tempo gaussiane), U/U/1 (con entrambe le distribuzioni uniformi) ecc.. o anche verificare i calcoli effettuati nel caso deterministico per sistemi D/D/n con la produzione di n parti.

Sono presenti nell'appendice a tal fine gli algoritmi utilizzati per le simulazioni, che restituiscono risultati con precisione di 10^{-2} , e che sono facilmente modificabili per eventuali analisi supplementari.

BIBLIOGRAFIA

- [1] F. Martinelli *“Dispense di Automazione Manifatturiera”*
- [2] Bielecki, Kumar *“Optimality of zero-inventory policies for unreliable manufacturing systems”*
- [3] S.P.Sethi, H.Yan, Q.Zhang *“Optimal and Hierarchical Controls in Dynamic Stochastic Manufacturing Systems: A Review”*
- [4] Y.Feng, H.Yan *“Optimal Production Control in a Discrete Manufacturing System with Unreliable Machines and Random Demands”*
- [5] Y.Feng, B.Xiao *“Optimal Threshold Control in discrete Failure-Prone Manufacturing Systems”*
- [6] J.R.Perkins, R.Srikant *“Failure-Prone Production Systems with Uncertain Demand”*
- [7] J.Hu, P.Vakili, G.Yu *“Optimality of Hedging Point Policies in the Production Control of Failure Prone Manufacturing Systems”*
- [8] J.R.Perkins, R.Srikant *“Hedging policies for Failure-Prone Manufacturing Systems: Optimality of JIT and Bounds on Buffer Levels”*
- [9] J.R.Perkins, R.Srikant *“Scheduling Multiple Part-Types in an Unreliable Single Machine Manufacturing System”*
- [10] M.Veatch, M.Caramanis *“Optimal Manufacturing Flow Controllers: Zero-Inventory Policies and Control Switching Sets”*
- [11] J.Hu, D.Xiang *“Optimal Control for Systems with Deterministic Production Cycles”*

APPENDICE

Algoritmo per sistemi D/D/1

```
#include <iostream.h>
#include <stdlib.h>
#include <math.h>
#include <string.h>
#include <stdio.h>
#include "conio.c"

//*****ALGORITMO DI CALCOLO DELLA SCORTA OTTIMA PER UN IMPIANTO D-D *****
FILE *pFile;
int n;
char str[18], nomefile[15];
double mi,d,S;
const int Ncicli=1000; // durata di ciascuna simulazione in termini di cicli di guasto/funzionamento
const int varianza0=5; /* nome un po' improprio per indicare il numero di simulazioni usato per stimare
ciascuna prestazione*/
double zstar0=0; // valore iniziale della scorta
int x=0;
int tg; // tempo deterministico di guasto
int cm; //costi di gestione dell'impianto per scorte positive o negative
int cp;
int tf; //tempo deterministico di funzionamento
double tStep=0,zStep=0,zstar_ok;
double mediacosti=0;

void ins_parametri ()
{
char ch;
// RICHIESTA PARAMETRI
cout<<"Inserisci MI (potenza massima del sistema): ";
cin>>mi;
cout<<"\nInserisci D (domanda costante): ";
cin>>d;
cout<<"\nVuoi utilizzare le scelte di default per gli altri parametri? (s/n)";
cin>>ch;
if (ch=='n') //INSERIMENTO PARAMETRI NON MEMORIZZATI
{
int par1,par2,par3,par4;
cout<<"\nInserisci T_FUNZ (tempo di funzionamento deterministico della macchina): ";
cin>>par1;
tf=par1;
cout<<"\nInserisci T_GUASTO (tempo di guasto deterministico della macchina): ";
cin>>par2;
tg=par2;
cout<<"\nInserisci CM (costo di mantenimento dell'impianto a scorta negativa): ";
cin>>par3;
cm=par3;
cout<<"\nInserisci CP (costo di mantenimento dell'impianto con scorta positiva): ";
cin>>par4;
cp=par4;
}
else //PARAMETRI DI DEFAULT
```

```

{
    tg=5;
    cm=80;
    cp=20;
    tf=150;
}
if (mi*tf<d*(tf+tg)) //STABILITA' DEL SISTEMA
{
    cout<<"\nIl sistema non e stabile\n\n";
    getch();
    exit;
}
S=mi*tf-d*(tf+tg); //MARGINE DI STABILITA'
cout<<"\nIl margine di stabilit... S e: ";
cout<<S;
cout<<"\n\nScegli TSTEP (passo di scansione per i valori di tstar)( 1-25, consigliato 1-5): \n";
cin>>tStep;
cout<<"\n\nScegli ZSTEP (passo di scansione per i valori di scorta massima dell'impianto)( 0.5-20,
consigliato 0.5-3): \n\n\n";
cin>>zStep;
clrscr();
cout<<"*****";
cout<<"\n          PARAMETRI DI LAVORO DELL'ALGORITMO\n\n";
cout<<"\nPotenza max dell'impianto MI: "<<mi;
cout<<"\nTasso di domanda costante D: "<<d;
cout<<"\nCM: "<<cm;
cout<<"\nCP: "<<cp;
cout<<"\nTempo di guasto: "<<tg;
cout<<"\nTempo di funzionamento: "<<tf;
cout<<"\nMargine di stabilit... del sistema: "<<S;
cout<<"\nPasso di tstar (istante di inizio del lavoro a potenza max): "<<tStep;
cout<<"\nPasso di zstar (massima scorta dell'impianto): "<<zStep;
cout<<"\nValore di SCORTA OTTIMA atteso: "<<((d*tg*cm)/(cm+cp));
cout<<"\nValore di COSTO MINIMO atteso: "<<(d*mi*cm*cp*tg*tg)/(2*(tf+tg)*(mi-d)*(cm+cp));
cout<<"\n*****\n\n\n";
getch();
}

int main()
{
    cout<<"          Universit... degli Studi di Roma \"Tor Vergata\" \n\n\n";
    cout<<"  ALGORITMO PER IL CALCOLO DELLA SCORTA OTTIMA IN UN SISTEMA D/D/I \n  (con
tempo di funzionamento deterministico e tempo di guasto deterministico)\n\n";
    ins_parametri(); //INIZIALE INSERIMENTO DEI PARAMETRI DI SISTEMA
    double costo_min;
    zstar_ok=(mi-d)*(tf); //Valore di scorta massima raggiungibile dal sistema realmente
    double valoricostimeno[int(zstar_ok/zStep)+1];
    double valoricostipiu[int(zstar_ok/zStep)+1];
    double valoricosti[int(zstar_ok/zStep)+1];
    double zcosti[int(zstar_ok/zStep)+1],costoJ[int(zstar_ok/zStep)+1];
    cout<<"\n\nInserisci il nome del file (max 15 caratteri). \nSe e' gia' esistente verra' sovrascritto.\n\n";
    cin>>nomefile; //INIZIALIZZAZIONE FILE DI SALVATAGGIO DATI
    strcpy(str, nomefile);
    strcat(str, ".m");
    pFile = fopen (str, "w");
    fprintf (pFile, "z=[");

```

```

int var2=0;
for (double zstar=zstar0; zstar<zstar_ok+0.1; zstar=zstar+zStep)
// CICLO DI LAVORO CON LA SCANSIONE DEGLI Z* RAGGIUNGIBILI PER VALUTARNE IL COSTO
CORRISPONDENTE
{
double tstar=tf-(zstar/(mi-d)); //DEFINIZIONE DI T* DIPENDENTE DA Z*
int var1=0;
int varianza=varianza0;
double mediacosti=0;
double TEMPO2=0;
int stato=1;
double xold=0;
double sommaC=0;
double tempo,rate,xnewtemp,xnew,deltaT_up,tm,deltaT_down,deltaT,deltaT_star;
for (int ciclo=1; ciclo<Ncikli+1; ciclo++) //SIMULAZIONE DI CICLI DI FUNZIONAMENTO-
GUASTO
{
double u=rand()%100;
u=u/100;
if (stato==0) //MACCHINA GUASTA
{
tempo=tg; // caso deterministico con tg costante
rate=-d;
xnewtemp=xold+rate*tempo;
}
else //stato==1 la macchina è in funzionamento
{
tempo=tf; //è la durata dell'intervallo di funzionamento
deltaT_up=-xold/(mi-d);
deltaT_down=xold/d;
tm=(xold+(mi-d)*tf-zstar)/mi;
if (xold<0) //XOLD NEGATIVO
{
if (deltaT_up<tf && deltaT_up<tstar) xnewtemp=(mi-d)*(tempo-tstar);
else if (deltaT_up<tf && deltaT_up>tstar) xnewtemp=(mi-d)*(tempo-deltaT_up);
else xnewtemp= (mi-d)*tempo + xold;
}
else if (xold==0) xnewtemp=(mi-d)*(tempo-tstar); //XOLD NULLO
else //XOLD POSITIVO
{
if ((zstar/(mi-d)+(xold/d))<tempo) xnewtemp=(mi-d)*(tempo-tstar);
else xnewtemp=xold - d*tm + (mi-d)*(tempo-tm);
}
}
TEMPO2=TEMPO2+tempo;
if (stato==0) //macchina guasta
{
xnew=xnewtemp;
if (xold*xnew>0) //entrambi pos/neg
{
if (xold>0) sommaC=sommaC+cp*(xnew+xold)*tempo/2; //entrambi pos
else sommaC=sommaC-cm*(xnew+xold)*tempo/2; //entrambi neg
}
else
{
//xold pos e xnew neg
deltaT=-xold/rate;

```

```

        sommaC=sommaC+cp*xold*deltaT/2-cm*xnew*(tempo-deltaT)/2;
    }
}
else //stato 1- macchina in funzione
{
    xnew=xnewtemp;
    if (xold*xnew>0) //xold e xnew con stesso segno
    {
        if (xold<0) sommaC=sommaC -cm*(xold+xnew)*tempo/2; //xold e xnew negativi
        else //xold e xnew positivi
        {
            if((xnew/(mi-d)+xold/d)<tempo)
sommaC=sommaC + cp*((xold*xold)/(2*d)+(xnew*xnew)/(2*(mi-d)));
            else sommaC=sommaC + cp*((xold+(xold-d*tm))*tm/2+(xnew+(xnew-(mi-d)*(tempo-
tm)))*(tempo-tm)/2);
        }
    } // XOLD NEGATIVO E XNEW POSITIVO
    else sommaC=sommaC + cm*(xold*(xold/(mi-d))/2)+cp*(xnew*(xnew/(mi-d))/2);
} //fine stato 1
xold=xnew;
stato=1-stato;
} //fine ciclo for su Ncicli
costoJ[var2]=sommaC/TEMPO2; //CALCOLO DI J
fprintf (pFile, "%f\n",zstar);
cout<<"\nIl valore di costo per zstar "<<zstar<<" risulta "<<costoJ[var2];//valoricosti[var2];
var2++;
} //fine del ciclo for su zStep
double min=costoJ[0];
int ind_min=0;
for (int j=0; j<int(zstar_ok/zStep)+1;j++)
{
    if (costoJ[j]<min)
    {
        min=costoJ[j];
        ind_min=j;
    }
}
costo_min=min;
//VISUALIZZAZIONE DEI DATI PRINCIPALI
cout<<"\nFattore di stabilit... del sistema: "<<S;
cout<<"\nIl valore minimo di costo trovato e: "<<costo_min<<" corrispondente allo zstar:
"<<zstar0+ind_min*zStep<<" e cioe al tstar "<<tf-((zstar0+ind_min*zStep)/(mi-d));
cout<<"\nLo zStep utilizzato era "<<zStep;
getch();
//MEMORIZZAZIONE DEI DATI DI COSTO SUL FILE
fprintf (pFile, "]\nJ=[");
for (int i=0 ; i<int(zstar_ok/zStep)+1 ; i++)
{
    fprintf (pFile, "%f\n",costoJ[i]);
}
fprintf (pFile, "]);
fclose (pFile);
return 0;
}

```


Algoritmo per sistemi U/D/1 con t^* fissato

```
#include <iostream.h>
#include <stdlib.h>
#include <time.h>
#include <math.h>
#include <string.h>
#include <stdio.h>
#include "conio.c"

//*****ALGORITMO DI CALCOLO DELLA SCORTA OTTIMA PER UN IMPIANTO U-D*****
FILE * pFile;
int n;
char str[18], nomefile[15];
double mi,d,S;
int Ncicli=1000; // durata di ciascuna simulazione in termini di cicli di guasto/funzionamento
int N=10;
double zstar0=0; // valore iniziale della scorta
int x=0;
int tg; // tempo deterministico di guasto
int cm; //costi di gestione dell'impianto per scorte positive o negative
int cp;
int tfunz_in, tfm; //la probabilità del tempo di funzionamento è uniforme tra tfunz_in e tfunz_fin
int tfunz_fin;
double tStep=0,zStep=0,tstar,zstar_ok;
double mediacosti=0;

void ins_parametri ()
{
    char ch;
    cout<<"Inserisci MI (potenza massima del sistema): ";
    cin>>mi;
    cout<<"\nInserisci D (domanda costante): ";
    cin>>d;
    cout<<"\nVuoi utilizzare le scelte di default per gli altri parametri? (s/n)";
    cin>>ch;
    if (ch=='n') //INSERIMENTO PARAMETRI
    {
        int par1,par2,par3,par4,par5,par6;
        cout<<"\nInserisci TFUNZ_IN (inizio dell'intervallo di probabilit... uniforme di rottura della macchina): ";
        cin>>par1;
        tfunz_in=par1;
        cout<<"\nInserisci TFUNZ_FIN (fine dell'intervallo di probabilit... uniforme di rottura della macchina): ";
        cin>>par2;
        tfunz_fin=par2;
        cout<<"\nInserisci T_GUASTO (tempo di guasto deterministico della macchina): ";
        cin>>par3;
        tg=par3;
        cout<<"\nInserisci CM (costo di mantenimento dell'impianto a scorta negativa): ";
        cin>>par4;
        cm=par4;
        cout<<"\nInserisci CP (costo di mantenimento dell'impianto con scorta positiva): ";
        cin>>par5;
        cp=par5;
    }
}
```

```

    }
else //PARAMETRI DI DEFAULT
{
    tg=5;
    cm=80;
    cp=20;
    tfunz_in=100;
    tfunz_fin=200;
}
if (mi*(tfunz_in+tfunz_fin)/2<d*((tfunz_in+tfunz_fin)/2+tg)) //STABILITA' DEL SISTEMA
{
    cout<<"\nIl sistema non e stabile\n\n";
    getch();
    exit;
}
S=mi*(tfunz_in+tfunz_fin)/2-d*((tfunz_in+tfunz_fin)/2+tg);
tfm= (tfunz_in+tfunz_fin)/2;
cout<<"\nIl parametro di stabilit... S e: ";
cout<<S;
cout<<"\n\nScegli TSTEP (passo di scansione per i valori di tstar)( 1-25, consigliato 1-5): \n";
cin>>tStep;
cout<<"\n\nScegli ZSTEP (passo di scansione per i valori di scorta massima dell'impianto)( 0.5-20,
consigliato 0.5-3): \n\n";
cin>>zStep;
clrscr();
cout<<"*****";
cout<<"\n          PARAMETRI DI LAVORO DELL'ALGORITMO\n\n";
cout<<" \nPotenza max dell'impianto MI: "<<mi;
cout<<" \nTasso di domanda costante D: "<<d;
cout<<" \nCM: "<<cm;
cout<<" \nCP: "<<cp;
cout<<" \nTempo di guasto: "<<tg;
cout<<" \nTempo di funzionamento: ["<<tfunz_in<<","<<tfunz_fin<<"]";
cout<<" \nFattore di stabilit... del sistema: "<<S;
cout<<" \nPasso di tstar (istante di inizio del lavoro a potenza max): "<<tStep;
cout<<" \nPasso di zstar (massima scorta dell'impianto): "<<zStep;
cout<<"\n*****\n\n\n";
getch();
}

int main()
{
    cout<<"          Universit... degli Studi di Roma \"Tor Vergata\" \n\n\n";
    cout<<"  ALGORITMO PER IL CALCOLO DELLA SCORTA OTTIMA IN UN SISTEMA U/D/1 \n
(con tempo di funzionamento uniforme e tempo di guasto deterministico)\n\n";
    ins_parametri(); //INIZIALE INSERIMENTO DEI PARAMETRI DI SISTEMA
    double costo_min;
    zstar_ok=(mi-d)*(tfm); //Valore di scorta massima raggiungibile dal sistema realmente
    double zcosti[int(zstar_ok/zStep)+1],costoJ[int(zstar_ok/zStep)+1],costoJpiu[int(zstar_ok/zStep)+1],
    costoJmeno[int(zstar_ok/zStep)+1];
    cout<<"\n\nInserisci il nome del file (max 15 caratteri). \nSe e' gia' esistente verra' sovrascritto.\n\n";
    cin>>nomefile; //INIZIALIZZAZIONE FILE DI SALVATAGGIO DATI
    strcpy(str, nomefile);
    strcat(str, ".m");
    pFile = fopen (str, "w");

```

```

fprintf(pFile, "z=[");
int var2=0;
srand (time(NULL));
for (double zstar=zstar0; zstar<=(zstar_ok)+0.1; zstar=zstar+zStep)
// CICLO DI LAVORO CON LA SCANSIONE DEGLI Z RAGGIUNGIBILI PER VALUTARNE IL
COSTO CORRISPONDENTE
{
    tstar=tfm-(zstar/(mi-d)); //DEFINIZIONE DI T* DIPENDENTE DA Z
    double mediacosti=0,varianza=0;
    double TEMPO2=0;
    int stato=1;
    double xold=0, costi[N];
    double sommaC=0;
    double tempo,rate,xnewtemp,xnew,deltaT_up,tm,deltaT_down,deltaT,deltaT_star;
    for (int k=0; k<N; k++)
    {
        for (int ciclo=1; ciclo<Ncicli+1; ciclo++)
        {
            double u=rand()%100;
            u=u/100;
            if (stato==0)
            {
                tempo=tg; // caso deterministico con tg costante
                rate=-d;
                xnewtemp=xold+rate*tempo;
            }
            else //stato==1 la macchina è in funzionamento
            {
                tempo=tfunz_fin - (u*tfunz_in); //è la durata dell'intervallo di funzionamento
                deltaT_up=-xold/(mi-d);
                deltaT_down=xold/d;
                if (tempo<tstar) //caso in cui la macchina si guasta prima che lavori a max regime
                {
                    if (xold==0) xnewtemp=0; //xold nullo
                    else if (xold<0) //xold negativo
                    {
                        if (deltaT_up<tempo) xnewtemp=0;
                        else xnewtemp=xold + (mi-d)*tempo; //il buffer non riesce ad arrivare a zero prima della
rottura
                    }
                }
                else
                {
                    //xold positivo
                    if (deltaT_down<tempo) xnewtemp=0; //il buffer arriva a zero prima della rottura
                    else xnewtemp=xold + (-d)*tempo; //il buffer non riesce ad arrivare a zero prima della
rottura
                }
            }
            else
            {
                //tempo>tstar
                tm=(xold+(mi-d)*tfm-zstar)/mi;
                if (xold<0) //XOLD NEGATIVO
                {
                    if (deltaT_up<tfm && deltaT_up<tstar) xnewtemp=(mi-d)*(tempo-tstar);
                    else if (deltaT_up<tfm && deltaT_up>tstar) xnewtemp=(mi-d)*(tempo-deltaT_up);
                    else xnewtemp= (mi-d)*tempo + xold;
                }
            }
        }
    }
}

```

```

else if (xold==0) xnewtemp=(mi-d)*(tempo-tstar); //XOLD NULLO
else //XOLD POSITIVO
{
    if ((zstar/(mi-d)+(xold/d))<tempo) xnewtemp=(mi-d)*(tempo-tstar);
    else xnewtemp=xold - d*tm + (mi-d)*(tempo-tm);
}
}
}
TEMPO2=TEMPO2+tempo;
if (stato==0) //macchina guasta
{
    xnew=xnewtemp;
    if (xold*xnew>0) //entrambi pos/neg
    {
        if (xold>0) sommaC=sommaC+cp*(xnew+xold)*tempo/2; //entrambi pos
        else sommaC=sommaC-cm*(xnew+xold)*tempo/2; //entrambi neg
    }
    else //xold pos e xnew neg
    {
        deltaT=-xold/rate;
        sommaC=sommaC+cp*xold*deltaT/2-cm*xnew*(tempo-deltaT)/2;
    }
}
else //stato 1- macchina in funzione
{
    if (xnewtemp<=zstar)
    {
        xnew=xnewtemp;
        if (xnew==0)
        {
            if (xold<0) sommaC=sommaC-cm*xold*deltaT_up/2;
            else if (xold>0) sommaC=sommaC+cp*xold*deltaT_down/2;
            else sommaC=sommaC;
        }
        else //xnew diverso da zero
        {
            if (xold*xnew>0) //entrambi pos/neg
            {
                if (xold>0) //entrambi pos
                {
                    if (deltaT_down>tempo) sommaC=sommaC+cp*(xold+xnew)*tempo/2;
                    else if (deltaT_down<tstar) sommaC=sommaC + cp*xold*deltaT_down/2
+cp*xnew*(tempo -tstar)/2;
                    else sommaC=sommaC + cp*(xold+xold-d*tstar)*tstar/2 + cp*(xnew+xnew-(mi-
d)*(tempo-tstar))*(tempo-tstar)/2;
                }
                else sommaC=sommaC-cm*(xold+xnew)*tempo/2; //entrambi neg
            }
            else //xold neg e xnew pos
            {
                if (xold<0 && xnew>0) sommaC=sommaC-cm*xold*deltaT/2+cp*xnew*(tempo-
tstar)/2;
                else if (xold==0 && xnew>0) sommaC=sommaC+cp*xnew*(tempo-tstar)/2;
            }
        }
    }
}
}

```

```

else //xnewtemp>zstar
{
    deltaT_star=(zstar)/(mi-d);
    xnew=zstar;//entrambi pos
    if (xold*xnew>0 && deltaT_down<tstar)
sommaC=sommaC+cp*xnew*deltaT_star/2+cp*zstar*(tempo-deltaT_star-tstar)+cp*xold*deltaT_down/2;
    else if (xold*xnew>0 && deltaT_down>=tstar) sommaC=sommaC + cp*(xold+xold-
d*tstar)*tstar/2 + cp*xnew*deltaT_star/2+cp*zstar*(tempo-deltaT_star-tstar);
    else if (xold==0) sommaC=sommaC + cp*xnew*deltaT_star/2 + cp*xnew*(tempo-tstar-
deltaT_star);
    else //xold neg e xnew pos
    sommaC=sommaC-cm*xold*deltaT_up/2+cp*xnew*(deltaT_star)/2+cp*zstar*(tempo-
deltaT_star-tstar);
}
} //fine stato 1
xold=xnew;
stato=1-stato;
} //fine ciclo for su Ncicli
costi[k]=sommaC/TEMPO2;
} //fine ciclo di N prove
for (int i=0; i<N; i++)
{
    mediacosti=mediacosti+costi[i]; //MEDIA degli N valori di J
}
mediacosti=mediacosti/N;
for (int i=0; i<N; i++)
{
    varianza=(mediacosti-costi[i])*(mediacosti-costi[i]);
}
varianza=varianza/N;
varianza=sqrt(varianza);
costoJ[var2]=mediacosti; //CALCOLO DI J
costoJpiu[var2]=mediacosti+varianza;
costoJmeno[var2]=mediacosti-varianza;
fprintf (pFile, "%f\n",zstar);
cout<<"\nIl valore medio di costo per zstar "<<zstar<<" risulta "<<costoJ[var2];//valoricosti[var2];
var2++;
} //fine del ciclo for su zStep
double min=costoJ[0];
int ind_min=0;
for (int j=0; j<int(zstar_ok/zStep)+1;j++)
{
    if (costoJ[j]<min)
    {
        min=costoJ[j];
        ind_min=j;
    }
}
costo_min=min;
//VISUALIZZAZIONE DEI DATI PRINCIPALI
cout<<"\n\nFattore di stabilit... del sistema: "<<S;
cout<<"\nIl valore minimo di costo trovato e: "<<costo_min<<" corrispondente allo zstar:
"<<zstar0+ind_min*zStep<<" e cioe al tstar "<<tfm-((zstar0+ind_min*zStep)/(mi-d));
cout<<"\n\nLo zStep utilizzato era "<<zStep;
getch();

```

```

// MEMORIZZAZIONE DEI DATI DI COSTO SUL FILE
fprintf(pFile, "\nJ1=[");
for (int i=0 ; i<int(zstar_ok/zStep)+1 ; i++)
{
    fprintf(pFile, "%f\n",costoJ[i]);
}
fprintf(pFile, "\nJ1piu=[");
for (int i=0 ; i<int(zstar_ok/zStep)+1 ; i++)
{
    fprintf(pFile, "%f\n",costoJpiu[i]);
}
fprintf(pFile, "\nJ1meno=[");
for (int i=0 ; i<int(zstar_ok/zStep)+1 ; i++)
{
    fprintf(pFile, "%f\n",costoJmeno[i]);
}
fprintf(pFile, "]");
fclose(pFile);
return 0;
}

```

Algoritmo per sistemi U/D/1 con t^* variabile

```

#include <iostream.h>
#include <stdlib.h>
#include <time.h>
#include <math.h>
#include <string.h>
#include <stdio.h>
#include "conio.c"

//*****ALGORITMO DI CALCOLO DELLA SCORTA OTTIMA PER UN IMPIANTO U-D*****
FILE * pFile;
int n;
char str[18], nomefile[15];
double mi,d,S;
int N=10;
int Ncicli=1000; // durata di ciascuna simulazione in termini di cicli di guasto/funzionamento
//N=15; // numero degli hedging point simulati
int varianza0=5; /* nome un po' improprio per indicare il numero di simulazioni usato per stimare
                    ciascuna prestazione*/
double zstar0=0; // valore iniziale della scorta
int x=0;
int tg; // tempo deterministico di guasto
int cm; //costi di gestione dell'impianto per scorte positive o negative
int cp;
int tfunz_in, tfm; //la probabilità del tempo di funzionamento è uniforme tra tfunz_in e tfunz_fin
int tfunz_fin;
double tStep=0,zStep=0,zstar_ok,zstarteoricoinf;
double mediacosti=0;

void ins_parametri ()
{
    char ch;

```

```

cout<<"Inserisci MI (potenza massima del sistema): ";
cin>>mi;
cout<<"\nInserisci D (domanda costante): ";
cin>>d;
cout<<"\nVuoi utilizzare le scelte di default per gli altri parametri? (s/n)";
cin>>ch;
if (ch=='n')          //INSERIMENTO PARAMETRI
{
    int par1,par2,par3,par4,par5,par6;
    cout<<"\nInserisci TFUNZ_IN (inizio dell'intervallo di probabilit... uniforme di rottura della macchina):
";
    cin>>par1;
    tfunz_in=par1;
    cout<<"\nInserisci TFUNZ_FIN (fine dell'intervallo di probabilit... uniforme di rottura della macchina): ";
    cin>>par2;
    tfunz_fin=par2;
    cout<<"\nInserisci T_GUASTO (tempo di guasto deterministico della macchina): ";
    cin>>par3;
    tg=par3;
    cout<<"\nInserisci CM (costo di mantenimento dell'impianto a scorta negativa): ";
    cin>>par4;
    cm=par4;
    cout<<"\nInserisci CP (costo di mantenimento dell'impianto con scorta positiva): ";
    cin>>par5;
    cp=par5;
}
else                  //PARAMETRI DI DEFAULT
{
    tg=5;
    cm=80;
    cp=20;
    tfunz_in=100;
    tfunz_fin=200;
}
if (mi*(tfunz_in+tfunz_fin)/2<d*((tfunz_in+tfunz_fin)/2+tg)) //STABILITA' DEL SISTEMA
{
    cout<<"\nIl sistema non e stabile\n\n";
    getch();
    exit;
}
S=mi*(tfunz_in+tfunz_fin)/2-d*((tfunz_in+tfunz_fin)/2+tg);
tfm= (tfunz_in+tfunz_fin)/2;
cout<<"\nIl parametro di stabilit... S e: ";
cout<<S;
cout<<"\n\nScegli TSTEP (passo di scansione per i valori di tstar)( 1-25, consigliato 1-5): \n";
cin>>tStep;
cout<<"\n\nScegli ZSTEP (passo di scansione per i valori di scorta massima dell'impianto)( 0.5-20,
consigliato 0.5-3): \n\n";
cin>>zStep;
clrscr();
cout<<"*****";
cout<<"\n          PARAMETRI DI LAVORO DELL'ALGORITMO\n\n";
cout<<"\nPotenza max dell'impianto MI: "<<mi;
cout<<"\nTasso di domanda costante D: "<<d;
cout<<"\nCM: "<<cm;
cout<<"\nCP: "<<cp;

```

```

cout<<" \nTempo di guasto: "<<tg;
cout<<" \nTempo di funzionamento: ["<<tfunz_in<<","<<tfunz_fin<<"]";
cout<<" \nFattore di stabilit... del sistema: "<<S;
cout<<" \nPasso di tstar (istante di inizio del lavoro a potenza max): "<<tStep;
cout<<" \nPasso di zstar (massima scorta dell'impianto): "<<zStep;
cout<<"\n*****\n\n\n\n";
getch();
}

int main()
{
cout<<"          Universit... degli Studi di Roma \"Tor Vergata\" \n\n\n";
cout<<"  ALGORITMO PER IL CALCOLO DELLA SCORTA OTTIMA IN UN SISTEMA U/D/1 \n
(con tempo di funzionamento uniforme e tempo di guasto deterministico)\n\n";
ins_parametri();      //INIZIALE INSERIMENTO DEI PARAMETRI DI SISTEMA
double costo_min;
zstar_ok=(mi-d)*(tfm); //Valore di scorta massima raggiungibile dal sistema realmente
double zcosti[int(zstar_ok/zStep)+1],costoJ[int(zstar_ok/zStep)+1];
double costoJpiu[int(zstar_ok/zStep)+1], costoJmeno[int(zstar_ok/zStep)+1];
cout<<"\n\nInserisci il nome del file (max 15 caratteri). \nSe e' gia' esistente verra' sovrascritto.\n\n";
cin>>nomefile;      //INIZIALIZZAZIONE FILE DI SALVATAGGIO DATI
strepv(str, nomefile);
strcat(str, ".m");
pFile = fopen (str, "w");
fprintf (pFile, "z=[");
srand (time(NULL));
int var2=0;
for (double zstar=zstar0; zstar<=(zstar_ok)+0.1; zstar=zstar+zStep)
// CICLO DI LAVORO CON LA SCANSIONE DEGLI Z* RAGGIUNGIBILI PER VALUTARNE IL
COSTO CORRISPONDENTE
{
int var1=0;
double costoT[int((tfm-(zstar/(mi-d)))/tStep)+1],costoTpiu[int((tfm-(zstar/(mi-
d)))/tStep)+1],costoTmeno[int((tfm-(zstar/(mi-d)))/tStep)+1];
for (double tstar=0; tstar<=(tfm-(zstar/(mi-d)))+0.1; tstar=tstar+tStep)
// CICLO DI LAVORO CON LA SCANSIONE DEGLI T* PER VALUTARNE IL COSTO
CORRISPONDENTE
{
double TEMPO2=0, varianza=0, mediacosti=0, costi[N];
int stato=1;
double xold=0;
double sommaC=0;
double tempo=0,rate=0,xnewtemp=0,xnew=0,deltaT_up=0,tm=0,deltaT_down=0,deltaT=0,
deltaT_star=0;
for (int k=0; k<N; k++)
{
for (int ciclo=1; ciclo<Ncicli+1; ciclo++)
{
double u=rand()%100;
u=u/100;
if (stato==0)
{
tempo=tg;          // caso deterministico con tg costante
rate=-d;
xnewtemp=xold+rate*tempo;
}

```



```

else //stato==1 la macchina è in funzionamento
{
tempo=tfunz_fin - (u*tfunz_in); //è la durata dell'intervallo di funzionamento
deltaT_up=-xold/(mi-d);
deltaT_down=xold/d;
if (tempo<tstar) //caso in cui la macchina si guasta prima che lavori a max regime
{
if (xold==0) xnewtemp=0; //xold nullo
else if (xold<0) //xold negativo
{
if (deltaT_up<tempo) xnewtemp=0;
else xnewtemp=xold + (mi-d)*tempo; //il buffer non riesce ad arrivare a zero prima della
rottura
}
else
{
//xold positivo
if (deltaT_down<tempo) xnewtemp=0; //il buffer arriva a zero prima della rottura
else xnewtemp=xold + (-d)*tempo; //il buffer non riesce ad arrivare a zero prima della
rottura
}
}
}
else
{
//tempo>tstar
tm=(xold+(mi-d)*tfm-zstar)/mi;
if (xold<0) //XOLD NEGATIVO
{
if (deltaT_up<tfm && deltaT_up<tstar) xnewtemp=(mi-d)*(tempo-tstar);
else if (deltaT_up<tfm && deltaT_up>tstar) xnewtemp=(mi-d)*(tempo-deltaT_up);
else xnewtemp= (mi-d)*tempo + xold;
}
else if (xold==0) xnewtemp=(mi-d)*(tempo-tstar); //XOLD NULLO
else //XOLD POSITIVO
{
if ((zstar/(mi-d)+(xold/d))<tempo) xnewtemp=(mi-d)*(tempo-tstar);
else xnewtemp=xold - d*tm + (mi-d)*(tempo-tm);
}
}
}
}
TEMPO2=TEMPO2+tempo;
if (stato==0) //macchina guasta
{
xnew=xnewtemp;
if (xold*xnew>0) //entrambi pos/neg
{
if (xold>0) sommaC=sommaC+cp*(xnew+xold)*tempo/2; //entrambi pos
else sommaC=sommaC-cm*(xnew+xold)*tempo/2; //entrambi neg
}
else
{
//xold pos e xnew neg
deltaT=-xold/rate;
sommaC=sommaC+cp*xold*deltaT/2-cm*xnew*(tempo-deltaT)/2;
}
}
}
else //stato 1- macchina in funzione
{
if (xnewtemp<=zstar)

```

```

{
xnew=xnewtemp;
if (xnew==0)
{
if (xold<0) sommaC=sommaC-cm*xold*deltaT_up/2;
else if (xold>0) sommaC=sommaC+cp*xold*deltaT_down/2;
else sommaC=sommaC;
}
else //xnew diverso da zero
{
if (xold*xnew>0) //entrambi pos/neg
{
if (xold>0) //entrambi pos
{
if (deltaT_down>tempo) sommaC=sommaC+cp*(xold+xnew)*tempo/2;
else if (deltaT_down<tstar) sommaC=sommaC + cp*xold*deltaT_down/2
+cp*xnew*(tempo -tstar)/2;
else sommaC=sommaC + cp*(xold+xold-d*tstar)*tstar/2 + cp*(xnew+xnew-(mi-
d)*(tempo-tstar))*(tempo-tstar)/2;
}
else sommaC=sommaC-cm*(xold+xnew)*tempo/2; //entrambi neg
}
else //xold neg e xnew pos
{
if (xold<0 && xnew>0) sommaC=sommaC-cm*xold*deltaT/2+cp*xnew*(tempo-
tstar)/2;
else if (xold==0 && xnew>0) sommaC=sommaC+cp*xnew*(tempo-tstar)/2;
}
}
}
else //xnewtemp>zstar
{
deltaT_star=(zstar)/(mi-d);
xnew=zstar;//entrambi pos
if (xold*xnew>0 && deltaT_down<tstar)
sommaC=sommaC+cp*xnew*deltaT_star/2+cp*zstar*(tempo-deltaT_star-tstar)+ cp*xold*deltaT_down/2;
else if (xold*xnew>0 && deltaT_down>=tstar)
sommaC=sommaC + cp*(xold+xold-d*tstar)*tstar/2 + cp*xnew*deltaT_star/2+cp*zstar*(tempo-
deltaT_star-tstar);
else if (xold==0)
sommaC=sommaC + cp*xnew*deltaT_star/2 + cp*xnew*(tempo-tstar-deltaT_star);
else //xold neg e xnew pos
sommaC=sommaC-cm*xold*deltaT_up/2+cp*xnew*(deltaT_star)/2+cp*zstar*(tempo-deltaT_star-tstar);
}
} //fine stato 1
xold=xnew;
stato=1-stato;
} //fine ciclo for su Ncicli
costi[k]=sommaC/TEMPO2;
} //fine ciclo di N prove
for (int i=0; i<N; i++)
{
mediacosti=mediacosti+costi[i]; //MEDIA degli N valori di J
}
mediacosti=mediacosti/N;
for (int i=0; i<N; i++)

```

```

    {
        varianza=(mediacosti-costi[i])*(mediacosti-costi[i]);
    }
    varianza=varianza/N;
    varianza=sqrt(varianza);
    costoT[var1]=mediacosti;          //CALCOLO DI J
    costoTpiu[var1]=mediacosti+varianza;
    costoTmeno[var1]=mediacosti-varianza;
    var1++;
}                                     // fine ciclo su TSTAR
double min_t=costoT[0];
int ind_min_t=0;
for (int j=0; j<int((tfm-(zstar/(mi-d)))/tStep)+1;j++)
{
    if (costoT[j]<min_t)
    {
        min_t=costoT[j];
        ind_min_t=j;
    }
}
costoJ[var2]=min_t;                  //CALCOLO DI J
costoJpiu[var2]=costoTpiu[ind_min_t];
costoJmeno[var2]=costoTmeno[ind_min_t];
fprintf (pFile, "%f\n",zstar);
cout<<"\nIl valore di costo per zstar " <<zstar<<" risulta " <<costoJ[var2]<<" con tstar:
"<<ind_min_t*tStep;//valoricosti[var2];
var2++;
}                                     //fine del ciclo for su zStep
double min=costoJ[0];
int ind_min=0;
for (int j=0; j<int(zstar_ok/zStep)+1;j++)
{
    if (costoJ[j]<min)
    {
        min=costoJ[j];
        ind_min=j;
    }
}
costo_min=min;
//VISUALIZZAZIONE DEI DATI PRINCIPALI
cout<<"\n\nFattore di stabilit... del sistema: "<<S;
cout<<"\nIl valore minimo di costo trovato e: "<<costo_min<<" corrispondente allo zstar:
"<<zstar0+ind_min*zStep<<" e cioe al tstar "<<tfm-((zstar0+ind_min*zStep)/(mi-d));
cout<<"\n\nLo zStep utilizzato era " <<zStep;
getch();
// MEMORIZZAZIONE DEI DATI DI COSTO SUL FILE
fprintf (pFile, "\nJ2=[");
for (int i=0 ; i<int(zstar_ok/zStep)+1 ; i++)
{
    fprintf (pFile, "%f\n",costoJ[i]);
}
fprintf (pFile, "\nJ2piu=[");
for (int i=0 ; i<int(zstar_ok/zStep)+1 ; i++)
{
    fprintf (pFile, "%f\n",costoJpiu[i]);
}

```

```
fprintf(pFile, "\nJ2meno=[");  
for (int i=0 ; i<int(zstar_ok/zStep)+1 ; i++)  
{  
    fprintf (pFile, "%f\n",costoJmeno[i]);  
}  
fprintf (pFile, "];");  
fclose (pFile);  
return 0;  
}
```