

## **CAP. 2**

### **2. CARATTERIZZAZIONI**

In questo capitolo vengono considerati vari aspetti che servono a caratterizzare in generale i comportamenti richiesti o ottenuti, sia dei sottosistemi del mondo esterno, sia dei calcolatori.

- Errori dovuti a disturbi o di quantizzazione.
- Modello discreto del tempo.
- Comportamento autonomo: stati continui e discreti. Flussi di eventi e loro proprietà di contenuto informativo e temporali.
- Comportamento reattivo: le azioni.
- Comportamento trasformazionale: le attività.
- Il mondo esterno: informazioni, comandi, segnali e tolleranze.
- Prestazioni temporali richieste.
- Prestazioni temporali offerte.
- Carico di lavoro del calcolatore e relative componenti.

## 2.1 CARATTERIZZAZIONE DEGLI ERRORI

### ERRORI E APPROSSIMAZIONI

Gli errori costituiscono un importante concetto in molte discipline, ed in particolare nell'ingegneria. Poichè tali concetti si prestano a distorsioni anche notevoli si ritiene importante un minimo di classificazione e di analisi, con brevi richiami a considerazioni che dovrebbero essere note da altri corsi.

**Diciamo che una certa entità è affetta da errori se presenta scostamenti da una situazione ideale considerata corretta.**

Notiamo subito che potrebbe essere opportuno distinguere tra scostamenti di entità arbitrariamente ampia (veri e propri errori) e scostamenti confinati ad intorno dei valori corretti (approssimazioni). Normalmente però, nel mondo tecnico e scientifico, si adotta il termine errore per designare indifferentemente i due casi, e così faremo quindi nel seguito.

#### 2.1.1 ERRORI FORMALI

Gli errori formali si riferiscono alla **descrizione** di un certo oggetto (astratto o concreto). I linguaggi di descrizione sono dotati di una certa sintassi che descrive le regole da rispettare perchè le frasi utilizzate nelle descrizioni siano corrette e quindi associabili ad un ben preciso significato.

Il **mancato rispetto delle regole sintattiche** costituisce errori formali che rendono la descrizione priva di significato.

In questi casi non si tratta di approssimazioni, ma di mancata correttezza, anche se in alcuni casi l'intuito umano è in grado di ricostruire la versione corretta.

#### 2.1.2 ERRORI DI COMPORTAMENTO

Gli errori di comportamento si riferiscono ad **implementazioni** realizzate sulla base delle relative specifiche. Si hanno errori quando un'implementazione **non soddisfa le specifiche**.

Si noti che le specifiche devono precisare i casi ideali o nominali considerati corretti ma, per essere complete, devono anche precisare le **tolleranze** accettabili.

Gli errori che presentano scostamenti dal caso ideale, ma contenuti entro le fasce di tolleranza, costituiscono approssimazioni accettabili che quindi non inficiano la validità dell'implementazione.

Gli errori di comportamento possono presentarsi con valori diversi al ripetersi delle stesse funzioni, ed in questi casi ha senso parlare di caso peggiore e di distribuzione probabilistica degli errori.

Normalmente l'accettabilità di una implementazione si ha quando i valori estremi (casi peggiori) degli errori sono contenuti entro la fascia di tolleranza

In alcuni casi si hanno specifiche più raffinate che precisano anche la probabilità accettabile che gli errori ricadano entro le specificate fasce di tolleranza. In questi casi è considerata valida anche un'implementazione che presenti errori arbitrariamente elevati ma con probabilità inferiore alla soglia precisata nelle specifiche.

Gli errori di comportamento possono essere suddivisi in errori di valore ed errori temporali, a cui sono dedicati i paragrafi seguenti.

##### 2.1.2.1 ERRORI DI VALORE

Sono scostamenti del valore di un'informazione o di una grandezza fisica rispetto al **valore vero** opportunamente definito.

Può essere opportuno distinguere diversi tipi di errori di valore, in relazione alle diverse operazioni a cui sono associati, come brevemente presentato nel seguito.

#### **2.1.2.1.1 Errori di acquisizione (Misura)**

##### **Scostamento tra valore rilevato e valore vero.**

Nelle operazioni di acquisizione di misure si possono avere diverse cause di errore, dovute al comportamento che si scosta dal caso ideale di tutta una serie di fenomeni e dispositivi utilizzati per effettuare le misure stesse.

Le problematiche relative a questi errori sono molto importanti rispetto alle applicazioni di automazione, e vengono approfondite nel successivo paragrafo 2.1.3.

#### **2.1.2.1.2 Errori di rappresentazione (Quantizzazione)**

##### **Scostamento tra valore rappresentato e valore vero.**

Gli errori di rappresentazione si hanno in tutti i casi in cui i simboli utilizzati non possono assumere tante configurazioni quanti sono i valori delle informazioni da rappresentare. Poiché le configurazioni di simboli costituiscono insiemi discreti a cardinalità finita, come ad esempio nel caso binario, la rappresentazione di informazioni a cardinalità superiore o infinita richiede operazioni di quantizzazione.

Gli errori di quantizzazione sono tipici dei sistemi digitali e meritano quindi un approfondimento nel successivo paragrafo 2.1.4.

#### **2.1.2.1.3 Errori di elaborazione**

##### **Scostamento tra risultato ottenuto e risultato ideale.**

Gli errori di elaborazione, a parte il caso banale di algoritmi non corretti rispetto alle specifiche, possono essere introdotti dalla limitatezza dei dispositivi di calcolo (*overflow*, troncamenti, ecc.) o dall'uso di metodi di calcolo intrinsecamente approssimati (metodi ad approssimazioni successive, ecc.). In questa sede l'argomento non viene ulteriormente approfondito, ma si invita il lettore a dedicare intelligente attenzione anche a questo tipo di errori.

#### **2.1.2.1.4 Errori di emissione**

##### **Scostamento tra valore vero e valore voluto.**

Analogamente a quanto detto per l'acquisizione di misure, anche l'emissione di informazioni può basarsi su dispositivi che, con il loro comportamento diverso dall'ideale, introducono scostamenti tra i valori effettivamente emessi ed i valori voluti.

A conclusione di quanto fin qui detto, si ricorda che gli errori introdotti da operazioni in sequenza (misura - rappresentazione - elaborazioni - emissione) sono generalmente cumulativi. Le verifiche di accettabilità vanno ovviamente eseguite sugli errori totali, mentre considerazioni di efficienza ingegneristica normalmente suggeriscono di ripartire equamente i margini di errore sulle diverse cause. In altre parole non ha molto senso ridurre (con costi elevati) gli errori di misura, se poi si commettono grossolani errori di elaborazione o di emissione.

#### **2.1.2.2 ERRORI TEMPORALI**

##### **Scostamento temporale tra l'istante in cui si compie un'azione e l'istante specificato per quell'azione.**

La correttezza del comportamento temporale costituisce l'aspetto centrale dei sistemi cosiddetti in tempo reale (*real-time*), ed a tale argomento vengono dedicate diverse parti di questo testo.

Anche la considerazione del tempo come una **grandezza fisica** da misurare presenta importanti aspetti che verranno approfonditi, e che in parte ricadono nella problematica degli errori di misura.

NOTA.

In alcuni casi gli errori temporali possono manifestarsi come errori di valore. Si pensi ad esempio al campionamento di una grandezza variabile nel tempo: un piccolo ritardo o anticipo dell'istante in cui si acquisisce il valore della grandezza porta ad ottenere un valore che si scosta da quello assunto dalla grandezza stessa nell'istante nominale.

### 2.1.3 ERRORI DI MISURA

In questa sezione non si intende trattare a fondo la teoria degli errori e le relative implicazioni filosofiche, ma solo richiamare e cercare di far chiarezza sui concetti operativi di base che dovrebbero essere bagaglio culturale di ogni ingegnere.

Gli errori che qui consideriamo sono dovuti a **imperfezioni** del dispositivo a **perturbazioni** da fenomeni interagenti. Rimandiamo ad un successivo paragrafo l'esame degli errori di **rappresentazione**, che peraltro sono intrinsecamente connessi con le misure concrete. Gli errori di misura sono generalmente **ineliminabili**, ma esistono varie tecniche per contenerli o anche ridurli. Poichè si è costretti a convivere con essi, è importante saperli valutare e confrontare con i requisiti di accettabilità.

Chiamiamo:

- **V** il valore numerico **Vero**, prescindendo da considerazioni su se e come sia eventualmente possibile conoscerlo, e che assumiamo appartenga all'insieme dei numeri reali;
- **M** la **Misura** del valore **V**, ottenuta con opportune tecniche e rappresentata con un valore numerico appartenente ad un opportuno insieme (interi o reali).

Definiamo **Errore Assoluto**  $E_a$ :

$$E_a = M - V$$

L'errore assoluto ha le stesse dimensioni del valore vero e della misura.

Definiamo **Errore Relativo**  $E_r$

$$E_r = \frac{M - V}{V} \approx \frac{E_a}{M}$$

L'errore relativo è adimensionale e generalmente viene espresso in percentuale.

#### 2.1.3.1 Distribuzione degli errori con misure ripetute

Se si effettuano ripetute operazioni di misura di **uno stesso valore** **V**, si otterranno in generale diversi valori **M**.

Sia  $M(k)$  il valore della  $k$ -esima esecuzione della misura di **V**, a cui è associato l'errore assoluto  $E_a(k) = M(k) - V$ .

Immaginando una serie infinita di misure, avremo una densità di probabilità degli errori che potrà assumere la forma di una Gaussiana con valore medio diverso da zero.

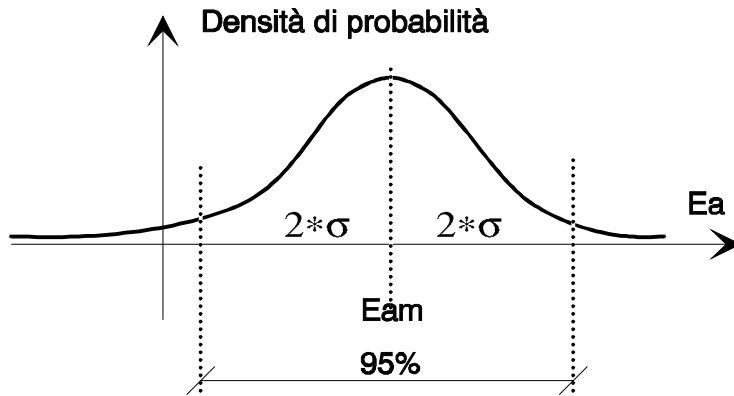


Fig. 2.1 - Densità di probabilità degli errori assoluti su infinite misure.

Il valore medio della distribuzione,  $E_{am}$ , è detto componente **sistematica** dell'errore  $E_a(k)$ , mentre lo scostamento  $E_a(k) - E_{am}$  è detto componente **accidentale** dell'errore  $E_a(k)$ .

Viene detto **errore limite di ripetibilità** il valore  $2 \cdot \sigma$ , cioè il doppio dello scarto quadratico medio, all'interno del quale  $(-2 \cdot \sigma < E_a < 2 \cdot \sigma)$  ricadono il 95% delle misure.

Riportiamo ora le definizioni di termini che caratterizzano la distribuzione probabilistica degli errori, secondo la terminologia standardizzata nel mondo della strumentazione, anche con i corrispondenti termini inglese e francese che, purtroppo, presentano assonanze che possono creare confusione.

**Ripetibilità** (inglese = *precision*, francese = *fidélité*)

E' la proprietà di una tecnica di misura che presenta una ridotta componente *accidentale* degli errori assoluti in misure eseguite sullo stesso valore  $V$  nel breve periodo (ridotto valore di  $\sigma$ ).

**Stabilità**

Analoga alla precedente, ma sul lungo periodo, e corrisponde ad una bassa *deriva* degli strumenti adottati.

**Accuratezza** (inglese = *unbias*, francese = *justesse*)

E' la proprietà di una tecnica di misura che presenta una ridotta componente *sistematica* degli errori assoluti in misure eseguite sullo stesso valore  $V$ .

**Precisione** (inglese = *accuracy*, francese = *précision*)

E' la proprietà di una tecnica di misura che presenta ridotti valori degli errori assoluti *globali*, cioè sia una ridotta componente accidentale che una ridotta componente sistematica.

**NOTA**

E' interessante notare come la ripetibilità di misure **medie** sia migliore della ripetibilità di misure singole. La varianza  $\sigma_N$  di valori ottenuti ognuno come media di  $N$  misure singole, supposte **scorrelate** rispetto ai disturbi, vale infatti:

$$\sigma_N = \frac{\sigma}{\sqrt{N}}$$

Quindi ad esempio si dimezzano gli errori accidentali assumendo come misura la media di 4 campioni. Attenzione ai disturbi periodici (ad es. da rete a 50 Hz) che producono errori non scorrelati.

### 2.1.3.2 Distribuzione degli errori per diversi valori $V$

La distribuzione degli errori potrà in generale dipendere dal valore vero  $V$ .

Consideriamo, per semplicità, i due casi tipici di errore **assoluto** costante ed errore **relativo** costante al variare di  $V$ .

Considereremo solo misure accurate, cioè con componente sistematica nulla, dato che in generale è possibile adottare provvedimenti di correzione di tale componente, mentre non è altrettanto facile ridurre le componenti accidentali.

Si noti che nel seguito verrà detto brevemente errore il valore dell'errore limite della distribuzione di errori per ogni  $V$

#### ERRORE ASSOLUTO COSTANTE

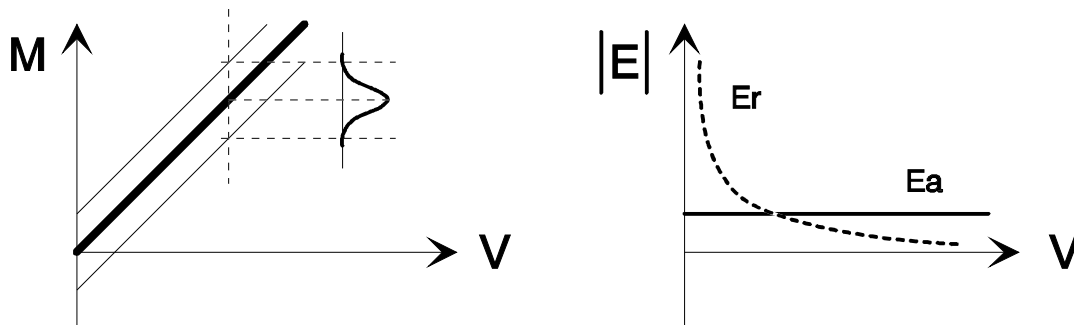


Fig. 2.2 - Andamento degli scostamenti delle misure in funzione di  $V$  e andamento dei moduli degli errori assoluti  $E_a$  e relativi  $E_r$

Si noti che con errore assoluto costante il modulo dell'errore relativo presenta un andamento iperbolico che tende ad infinito per valori di  $V$  tendenti a zero.

Una distribuzione di questo tipo è generalmente accettabile solo per misure di grandezze ad **origine arbitraria**, come è il caso del tempo assoluto e dello spazio assoluto, oppure per grandezze per cui il campo di valori utili è relativamente stretto e lontano dallo zero.

#### ERRORE RELATIVO COSTANTE

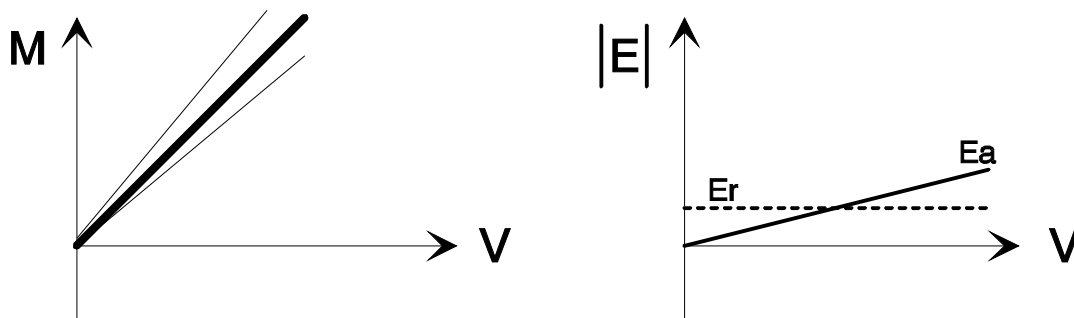


Fig. 2.3 - Scostamenti delle misure in funzione di V e andamenti dei moduli degli errori assoluti  $E_a$  e relativi  $E_r$ .

Una distribuzione ad errore relativo costante è adatta per tutte le grandezze con **origine assoluta** (intrinseca), e quindi anche per **intervalli** di tempo e **distanze** spaziali.

**NOTA**

**CLASSE DI PRECISIONE DI UNO STRUMENTO**

La classe di precisione costituisce un parametro standard per caratterizzare la precisione di uno strumento, o in generale di una tecnica di misura, ed esprime in percentuale il **rapporto fra il massimo modulo dell'errore assoluto  $E_a(V)$  e il valore di fondo scala  $V_{fs}$** .

$$Cl = \frac{\max(|E_a(V)|)}{V_{fs}} \cdot 100 \quad \text{con } 0 \leq V \leq V_{fs}$$

Quindi si parlerà di Cl 1.5 di uno strumento i cui errori assoluti in tutta la scala di misura sono inferiori in modulo a 1.5% del valore di fondo scala.

## 2.1.4 ERRORI DI QUANTIZZAZIONE

Gli errori di quantizzazione non sono dovuti a fenomeni perturbanti, ma alla particolare **rappresentazione** dei valori delle misure  $M$  con un numero finito di simboli, come è tipico delle rappresentazioni binarie con un numero finito di bit.

Gli errori di quantizzazione sono dovuti al fatto che **tutto un intervallo** di valori veri viene rappresentato con un **unico** valore  $M$ .

Gli errori di quantizzazione sono additivi rispetto ad altri eventuali errori, presentano una dispersione nulla rispetto alla ripetibilità, mentre presentano distribuzioni rispetto ai valori di  $V$  che andiamo a considerare nei due casi tipici di rappresentazione di  $M$  come valore `integer` e come valore `float`.

Chiamiamo **granularità** (o risoluzione)  $Gr(V)$  l'ampiezza dell'intervallo di valori di  $V$  che è rappresentato da uno stesso valore di  $M(V)$ .

### 2.1.4.1 MISURE RAPPRESENTATE CON INTERI

Se rappresentiamo  $M(V)$  con valori interi abbiamo una granularità

$$Gr(V) = \text{costante}$$

Sono interessanti due situazioni, che vengono presentate nelle figure seguenti.

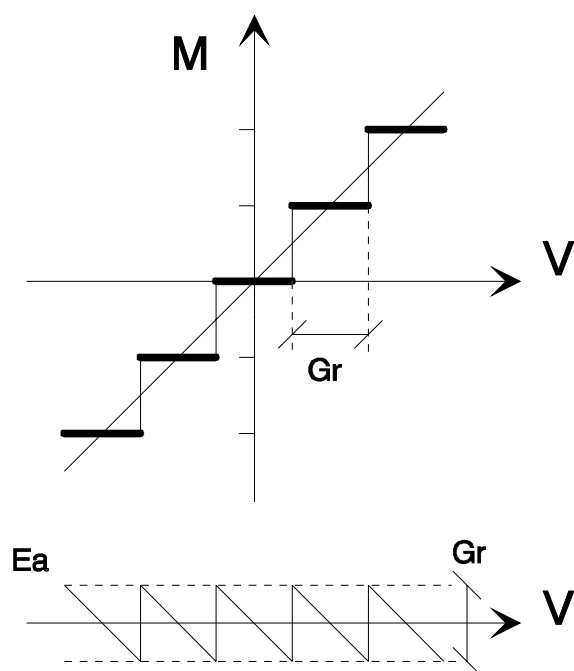


Fig. 2.4 - Andamento di M in funzione di V e corrispondente errore assoluto  $Ea(V)$ , nel caso di perfetta accuratezza (valor medio degli errori pari a zero). - Arrotondamento.

La figura precedente (fig. 2.4) rappresenta la situazione che si ottiene, ad esempio, con un convertitore Analogico / Digitale accuratamente tarato in modo che il valore  $M = 0$  corrisponda esattamente all'intervallo di valori

$$\frac{-Gr}{2} \leq V < \frac{+Gr}{2}$$

In questo caso, per ogni V il valore dell'errore assoluto è compreso tra i valori

$$\frac{-Gr}{2} \leq Ea < \frac{+Gr}{2}$$

Errori di questo tipo si ottengono anche con operazioni di **arrotondamento**.

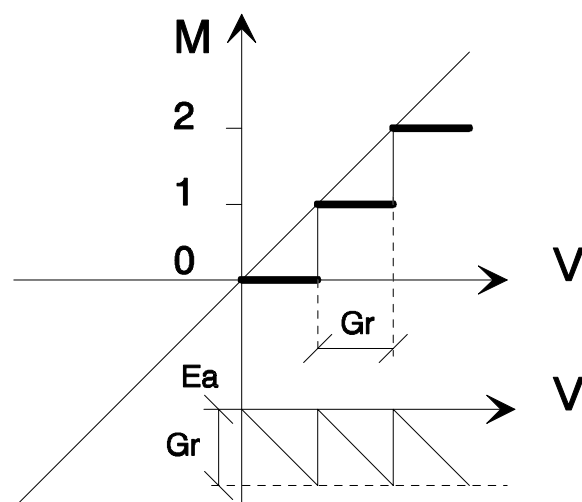


Fig. 2.5 - Andamento di  $M(V)$  e corrispondente errore assoluto  $Ea(V)$  con numerazione naturale. Troncamento.



Il caso rappresentato nella figura precedente (fig.2.5) si presenta quando si eseguono misure di tempo mediante conteggi incrementali a partire dalla posizione di “zero”, cioè con l’orologio **attivato all’origine dei tempi**.

Questi errori, sempre per difetto, sono propri delle operazioni di **troncamento**.

In questo caso il valore dell’errore assoluto è compreso tra i valori

$$-Gr < Ea \leq 0$$

NOTA. Nelle misure di spazio percorso, ottenute contando gli impulsi di un encoder incrementale rispetto alla posizione di zero-macchina, il passaggio da  $M = 0$  a  $M = 1$  può verificarsi (di solito casualmente) per valori di  $V$  compresi tra

$$0 < V < +Gr$$

E’ del tutto analogo a questo anche il caso di misure di tempo con orologio che conti con continuità (*free running*) e non sincronizzato con l’origine dei tempi.

NOTA

Tra arrotondamento (RND) e troncamento (TRUNC) vale la relazione:

$$RND(X) = TRUNC(X + Gr/2)$$

per  $X > 0$

### 2.1.4.2 MISURE RAPPRESENTATE CON VIRGOLA MOBILE

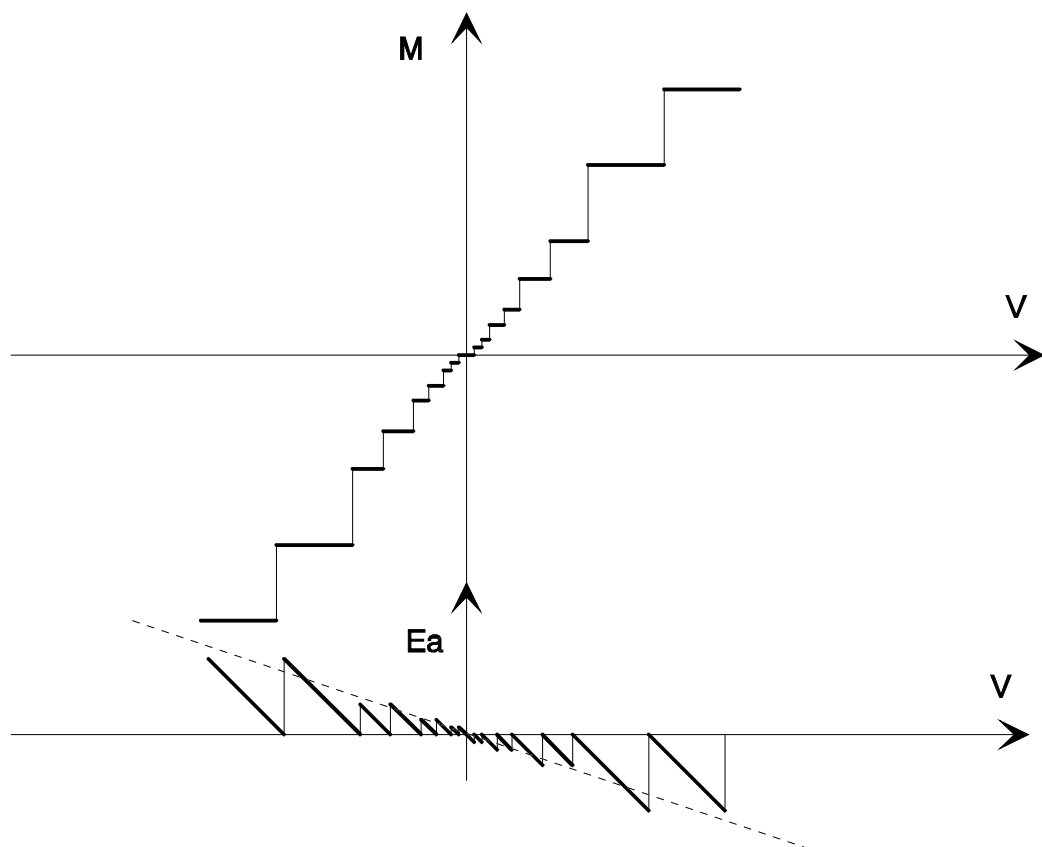


Fig. 2.6 - Andamento indicativo di rappresentazione in virgola mobile e relativi errori.

Con la rappresentazione in virgola mobile (`float`) la granularità, e quindi gli errori di quantizzazione, non è costante, ma cresce proporzionalmente al valore numerico rappresentato.

NOTA. Più correttamente la granularità non cresce con continuità, ma raddoppia ad intervalli ognuno di lunghezza doppia del precedente.

La figura precedente (fig. 2.6) corrisponde al caso di mantissa di un solo bit, e non è quindi rappresentativa di un caso reale, ma vuole solo evidenziare il concetto intuitivo di errori assoluti crescenti linearmente con il valore  $V$ , e quindi errori relativi costanti.

## 2.2 CARATTERIZZAZIONE DEL TEMPO

Tempo fisico lineare continuo  
 Istanti e intervalli espressi da n. reali matematici  
 Tempo discreto: filosofia e conseguenze  
 Visione con diverse scale temporali  
 Continuo virtuale

Il concetto di tempo è tanto intuitivo quanto difficile da definire univocamente. Mentre non è questa la sede per una dissertazione filosofica, la pregnanza del tempo in molti problemi, ed in particolare nelle applicazioni di elaborazione in "tempo reale", suggerisce un approfondimento di questo concetto almeno in relazione al dominio applicativo che qui ci interessa.

E' interessante notare che il fluire del tempo è sempre stato messo in corrispondenza con l'evoluzione di fenomeni, tanto da rendere difficile dire se è preferibile pensare al tempo come un "contenitore" all'interno del quale si svolgono i fenomeni, o se i fenomeni stessi sono da ritenere come concetti primari che in qualche modo "provocano" il fluire del tempo.

Seguendo un'attitudine mentale diffusa assumeremo che il tempo sia un concetto primario, cioè il suo fluire procede indipendentemente dal fatto che si verifichino o meno dei fenomeni.

Operativamente però adotteremo alcuni particolari fenomeni detti "**orologi**" (clock), generatori di eventi assunti come "regolari", come strumenti per quantificare il trascorrere del tempo da mettere a sua volta in relazione con gli altri fenomeni.

### 2.2.1 IL MODELLO DEL TEMPO

Il modello di tempo a cui siamo abituati a far riferimento è basato su una corrispondenza biunivoca tra istanti temporali e numeri reali (in senso matematico) con una scelta arbitraria dell'origine (istante zero). Con tale assunzione gli istanti temporali assumono le caratteristiche proprie dei numeri reali (continuità, limite, ecc.) che ne consentono la trattazione analitica su cui sono basate, ad esempio, le leggi della fisica classica.

Il concetto di istante, con durata temporale rigorosamente nulla, veramente non è facile da mettere in corrispondenza con i fenomeni fisici che si assumono come orologi.

Infatti, qualunque sia la legge fisica su cui si basa il fenomeno "orologio", questo evolve secondo fasi di durata che per quanto sia breve è tuttavia finita.

Si tratta comunque di un'astrazione significativa da un punto di vista ingegneristico, data la notevole differenza di valori di velocità dei vari sistemi dinamici (tra l'anno ed il nanosecondo ci sono circa 15 ordini di grandezza).

In ogni scala temporale si assumono quindi come nulle durate temporali durante le quali i fenomeni in esame non subiscono evoluzione di rilievo e di conseguenza si assumono come "istantanei" i fenomeni (ad es. commutazioni del clock) che in tali durate si esauriscono.

In altre parole, dato un determinato **contesto di fenomeni** da correlare al tempo, è necessario individuare un fenomeno di qualche ordine di grandezza più rapido dei fenomeni di tale contesto, così da poterlo considerare un **buon orologio**.

In particolare i fenomeni elettronici sono in genere ottimi candidati al ruolo di orologi, e tipicamente i cosiddetti oscillatori ad "onda quadra", i cui fronti presentano tempi di commutazione di pochi ns, considerabili spesso come trascurabili.

Mentre è comodo per il ragionamento umano mantenere il modello di tempo continuo, per i sistemi di elaborazione digitali si deve adottare un modello di tempo che tenga conto del “**discreto**” intrinseco sia nella rappresentazione digitale dei **valori** (si noti: anche con le variabili di tipo *real*) sia nell'evoluzione **temporale** “a **passi**” prodotta dall'esecuzione delle istruzioni.

Un'interessante ed approfondita trattazione di modelli del tempo, anche per il caso più complesso di sistemi distribuiti, è presentata nel cap. 2 di [LEAG90], a cui si rimanda il lettore interessato, mentre nel seguito viene presentato e sinteticamente discusso un modello orientato all'implementazione e riferito ad un singolo processore.

Chiamiamo **orologio** una sorgente di informazione sul tempo corrente, il cui dominio è costituito dai **numeri naturali** (`unsigned int`) codificati con un numero di bit adeguato al campo di valori da utilizzare (tipicamente 16 bit, ma talora anche 24 o 32 bit).

Assumiamo cioè un modello di tempo **lineare** (qui significa ad un solo valore, cioè non “ramificato”) **monotono** (almeno concettualmente) e **discreto**.

Nel seguito il termine *locale* significa *relativo all'(i-esimo) orologio considerato*, se nel sistema sono presenti diversi orologi.

- **Unità di tempo locale** (*utl*) = intervallo di durata in tempo assoluto corrispondente ad una unità del valore numerico orologio.

Questa unità di tempo rappresenta la **risoluzione temporale** locale ed è una proprietà della rappresentazione locale del tempo.

- **Base tempo locale** (*BTL*) = multiplo (generalmente intero) dell'unità di tempo locale, che rappresenta il periodo del fenomeno ciclico assunto come orologio e quindi corrisponde all'intervallo di aggiornamento della variabile orologio.

La base tempo locale rappresenta la **granularità temporale** locale ed è una proprietà del comportamento fisico dell'orologio.

Il caso  $BTL = 1$  comporta la ragionevole coincidenza di risoluzione e granularità.

Il caso  $BTL = K > 1$  risulta comodo se si preferisce mantenere un'unità di tempo convenzionale, che sia ad esempio un sottomultiplo del *secondo*, pur non disponendo di un fenomeno ciclico di periodo corrispondente.

Si noti, ad esempio, che nei PC con MS-DOS il valore usuale di  $K$  è  $K = 55$  ms (circa), cioè la variabile orologio che il DOS mette a disposizione delle applicazioni presenta una risoluzione (unità di tempo locale) di 1 ms ma una granularità (base tempo locale) di 55 ms. In altre parole il tempo è rappresentato in millisecondi, ma viene aggiornato da un interrupt che si verifica 18.2 volte al secondo, cioè ogni 55 ms.

Operativamente si assume che per ogni processore sia definita almeno una variabile di tipo "orologio" che gode della proprietà di incrementare di BTL il proprio valore negli istanti in cui si verificano particolari eventi, che chiameremo "**tick**". Tali eventi si verificano ad intervalli di tempo assoluto regolari e pari a BTL unità di tempo.

Chiamiamo "**T**" il tempo rappresentato dall'orologio e "**t**" il tempo assoluto (cioè "reale").

Il comportamento di tale orologio potrà essere espresso da una funzione nel dominio del tempo:

$$T = f(t)$$

definita come:  $T = n * BTL$

per  $n * BTL * utl \leq t < (n+1) * BTL * utl$

con **n** appartenente all'insieme dei numeri naturali

Si noti che con un orologio di **N** bit l'espressione di **T** è più correttamente:

$$T = (n * BTL) \bmod 2^N$$

Nel seguito però, per semplicità, si assumerà che il numero di bit **N** sia abbastanza grande da poter assumere che per i valori degli intervalli di tempo **X** che ci interessano sia  $X = X \bmod 2^N$ .

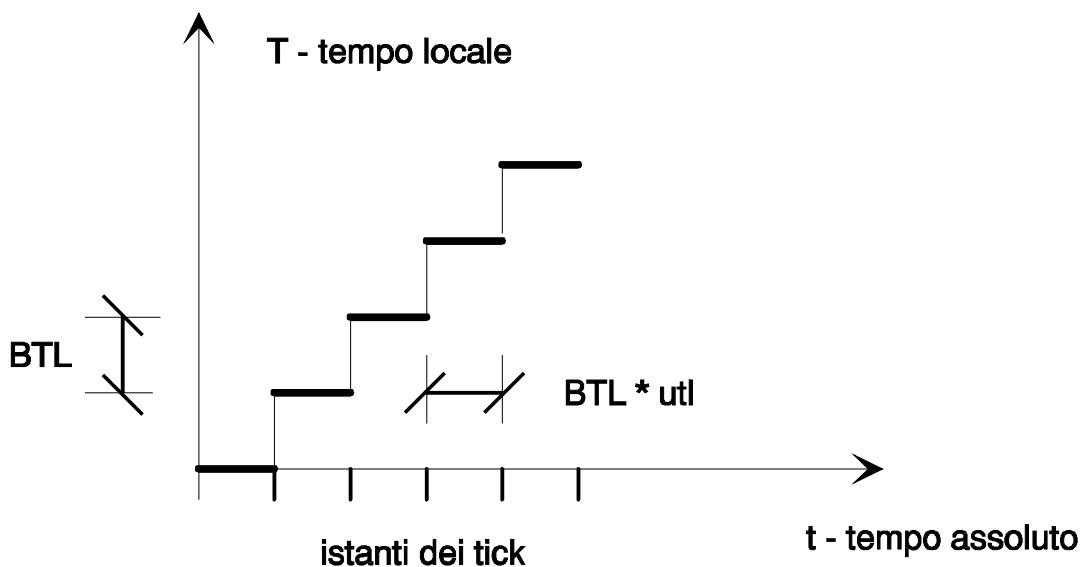


Fig. 2.7 - Corrispondenza tra tempo locale e tempo assoluto.

Si ha in tal modo un modello di tempo discreto, che rappresenta con uno stesso valore tutti gli istanti contenuti in un intervallo tra due tick consecutivi.

Nel seguito, per semplicità, assumiamo che si adotti la stessa unità di misura per **T** e **t**, cioè che sia  $utl = 1$ .

### 2.2.1.1 ERRORI

Gli errori che si commettono nell'adottare come misura del tempo una variabile orologio di tempo locale, come sopra descritto, sono scomponibili in due contributi:

- errori dovuti al fenomeno assunto come orologio (precisione)
- errori dovuti alla granularità (quantizzazione)

#### 2.2.1.1.1 Errori del fenomeno

Consideriamo per ora gli errori dovuti allo scostamento del fenomeno assunto come orologio dal comportamento ideale.

Diciamo:- **Preciso** (orologio ideale) un orologio per cui, per ogni "n", sia:

$$\begin{aligned} & t(\text{tick}[n]) - t(\text{tick}[n-1]) = \text{BTL} + e[n] \\ \text{con errore} \quad & e[n] = 0 \end{aligned}$$

La precisione di un orologio esprime la regolarità dei suoi tick e la corrispondenza del loro periodo con il valore  $\text{BTL} \cdot \text{utl}$ .

Se gli **errori**  $e[n]$  sono **sistematici** (orologio poco accurato) il fenomeno periodico assunto come sorgente presenta una frequenza non corretta.

Se gli **errori**  $e[n]$  sono a **valor medio nullo** (orologio poco ripetibile) il fenomeno non è rigorosamente periodico, oppure (più spesso) è irregolare il meccanismo di aggiornamento del valore **T**.

Le irregolarità di periodo a valor medio nullo sono compensanti (cioè non inficiano l'accuratezza) ed in gergo elettronico sono dette "**jitter**".

Diciamo: - **Corretto al tempo assoluto t** un orologio per cui sia:

$$\text{abs}(T - t) < \text{BTL}$$

La correttezza di un orologio esprime la sincronizzazione del tempo locale **T** da esso rappresentato, col tempo assoluto **t**, a meno dell'intrinseco errore di quantizzazione **BTL**.

Un orologio **preciso** se è corretto al tempo **t** lo è anche per ogni  $t' > t$ .

La precisione esprime la capacità di mantenere la correttezza.

La correttezza si perde in presenza di inaccuratezza sistematica ( $e\text{-medio} \neq 0$ ), mentre tende a mantenersi con errori a valor medio nullo.

La precisione può essere quantificata con il rapporto

$$\text{BTL} / e\text{-medio}$$

e con orologi basati su oscillatori a quarzo può avere valori di  $\approx 10^4 \dots 10^6$ .

Si noti che gli orologi realizzati con tecniche digitali presentano caratteristiche di stabilità, accuratezza e precisione che sono esattamente quelle dell'oscillatore usato per realizzarli.

**NOTA**

Si rifletta sul fatto che gli errori sistematici di un orologio provocano una deriva, cioè uno scostamento del valore di tempo rappresentato **T** dal tempo reale **t** progressivamente crescente in modulo. Come conseguenza anche inaccuratezze più che accettabili nel breve periodo, portano nel **lungo periodo ad errori inaccettabili**. Nessun orologio reale manterrà indefinitamente uno scostamento entro limiti finiti.

Si pensi alla comune esperienza di aggiustare l'ora dei nostri orologi.

Si invita ancora a riflettere sul fatto che ciò dipende dalla particolare caratteristica della grandezza **tempo di non avere un'origine assoluta** (cosa che del resto vale anche per lo spazio). Invece gli **intervalli di tempo**, cioè le durate, ricadono nelle normali considerazioni, per cui l'errore che si commette dipende solo dalla imprecisione dell'orologio e non dall'istante in cui lo si consulta.

### 2.2.1.1.2 Errori di quantizzazione temporale

E' importante notare che con questo modello di tempo discreto si hanno conseguenze dovute agli errori di quantizzazione, di cui occorre tener conto.

Si ricorda che per semplicità di notazione si assume che l'unità di tempo locale (utl) coincida con l'unità di tempo utilizzata per il tempo assoluto, cioè  $utl = 1$ .

Di conseguenza la base tempo locale (granularità) BTL esprime l'intervallo tra due *tick* consecutivi sia nella unità di misura del tempo locale che in quella del tempo assoluto.

Negli esempi si adotterà  $BTL = 55 \text{ ms}$ , come è nel caso di PC con sistema operativo MS-DOS.

- **Ordinamento** - Non è possibile, in base a marcature temporali effettuate usando il tempo locale (*time stamping*), individuare un ordinamento temporale tra eventi verificatisi nell'intervallo tra due *tick* consecutivi;

- **Contemporaneità** - Eventi che si verificano nell'intervallo tra due *tick* consecutivi sono considerati contemporanei nel tempo locale pur distando di un tempo assoluto  $Dt < BTL$ ;

**Es.** eventi "contemporanei" possono avere distanza reale fino a 55- ms

- **Misure di intervalli** - Eventi a distanza temporale  $Dt$  nel tempo assoluto sono osservati nel tempo locale a distanza  $M(Dt)$  (misura dell'intervallo  $Dt$ ) con  $N \cdot BTL \leq M(Dt) \leq (N + 1) \cdot BTL$ , dove N vale:

$$N = \left\lfloor \frac{M(Dt)}{BTL} \right\rfloor$$

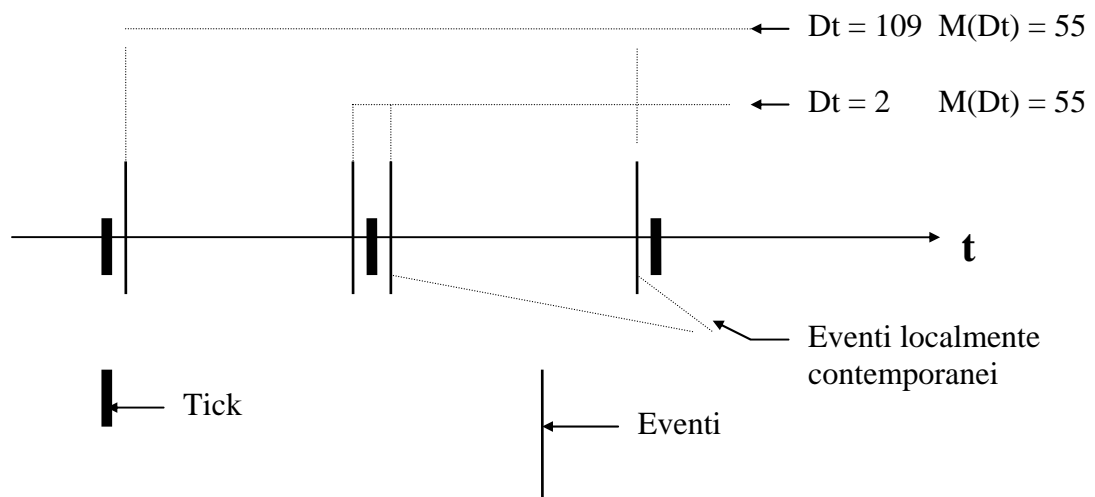
(Si ricorda che l'operatore *floor*  $\lfloor x \rfloor$  indica l'intero ottenuto per troncamento di x)

**Es.** eventi a distanza reale di 57 ms ( $N = 1$ ) possono apparire, in base al loro *time-stamping*, a distanza di 55 ms o di 110 ms.

- Viceversa eventi a distanza temporale  $N \cdot BTL$  nel tempo locale (con N intero  $> 0$ ) possono presentare nel tempo assoluto una distanza  $Dt$  che soddisfa la relazione

$$(N-1) \cdot BTL < Dt < (N+1) \cdot BTL$$

**Es.** eventi a distanza locale di 55 ms possono distare in realtà da 0+ ms fino a 110-



Esempio di possibili collocazioni sull'asse dei tempi reali di *tick* ed eventi.



## 2.3 CARATTERIZZAZIONE DEL COMPORTAMENTO

Il comportamento di un sistema può essere caratterizzato mediante l'andamento in funzione del tempo di proprietà (stati) le cui variazioni costituiscono degli eventi.

Nelle interazioni tra sottosistemi si possono considerare le attività e le azioni come effetto di alcune variazioni di stato e causa di altre variazioni di stato. Attività ed azioni costituiscono cioè, con questa visione, il supporto delle interazioni intese come concatenamenti di cause ed effetti.

Chiamiamo **comportamento autonomo** l'evoluzione nel tempo considerata a prescindere dalle cause che la producono.

Chiamiamo **comportamento trasformatore** quello che crea una relazione funzionale tra uscite ed ingressi di un dispositivo  $U = F(I)$

ed è basato su **attività** che realizzano la funzione di trasferimento  $F$ .

Chiamiamo **comportamento reattivo** quello che in risposta a stimoli costituiti da eventi in ingresso produce, con opportune **azioni** e dopo un certo ritardo, delle risposte che costituiscono gli eventi in uscita.

Consideriamo inizialmente gli aspetti che caratterizzano un comportamento considerato come autonomo.

### 2.3.1 CONCETTI DI STATO E DI EVENTO

I concetti di stato e di evento sono relativamente intuitivi ma meritano un approfondimento in relazione alle informazioni ad essi associate, ed in particolare tenendo conto del tipico modello di esecuzione dei "normali" calcolatori i quali, come è noto, presentano una funzionalità prodotta "volontariamente" dall'interno (esecuzione dei programmi) e non "subita" dall'esterno come accade invece per i dispositivi a funzionalità intrinseca.

In particolare, come vedremo nei prossimi capitoli, questi concetti ricorrono in varie occasioni nella caratterizzazione del comportamento del mondo esterno, nelle problematiche di interfacciamento e nell'implementazione di sistemi di elaborazione in tempo reale.

Vediamo ora brevemente i vari tipi di **stati** continui o discreti nei **valori** e continui o discreti nel **tempo**.

### 2.3.1.1 STATI CONTINUI A TEMPO CONTINUO

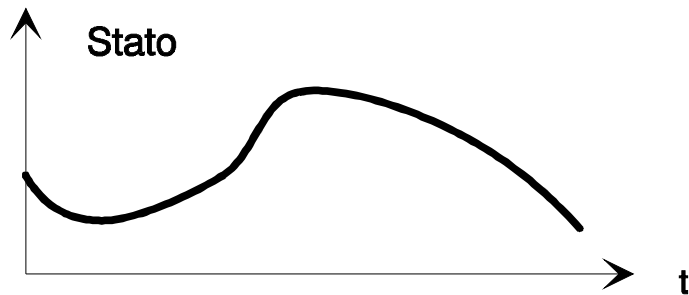


Fig. 2.8 - Andamento temporale di uno stato continuo

Entità che possono assumere tutti gli **infiniti valori in un intervallo**  $V_{MIN}..V_{MAX}$

$$V_{MIN} \leq STATO(t) \leq V_{MAX}$$

e che variano con continuità nel tempo; esiste cioè un intervallo  $Dt$  abbastanza piccolo per cui, per qualunque (piccolo)  $\epsilon$ :

$$abs(STATO(t) - STATO(t+Dt)) < \epsilon$$

Sono le tipiche informazioni **analogiche** studiate nella teoria dei sistemi continui e generalmente ben caratterizzate dal campo di valori assumibili e dalla **banda di frequenze** significative (nei casi più raffinati dallo spettro di frequenze), tipicamente da 0 a  $F_{max}$ .

Altri modi per caratterizzare gli stati continui a tempo continuo possono essere basati sulle costanti di tempo o sullo "slew rate".

La **costante di tempo** di un fenomeno del primo ordine è il tempo necessario perchè lo scostamento dal valore asintotico di regime passi da  $S$  a  $S/e$  ( $e = 2.71828$ ). Questo parametro dà un'idea del tempo necessario per *l'esaurirsi dei transitori*, normalmente assunto pari a 4..5 volte la costante di tempo.

Lo **slew rate** è il massimo valore assoluto della derivata rispetto al tempo di una grandezza fisica. Questo parametro esprime la massima *velocità* che il fenomeno può assumere nel contesto considerato.

Per questi stati continui a tempo continuo *non esistono eventi intrinseci*. Si noti che però possono essere introdotti degli eventi se si considerano asserzioni di verità sugli attributi degli stati (detti predicati di stato), come ad esempio che *una grandezza supera una soglia  $S$*  oppure che *è compresa in una fascia di valori ( $V1...V2$ )*. In tal modo si crea un mappaggio, certamente non biunivoco e talora anche non univoco, degli infiniti valori continui su un insieme finito di attributi. Si noti che così facendo si è in realtà introdotto un insieme finito di stati (astratti) discreti (v. seguito).

Gli stati continui a tempo continuo costituiscono l'unico tipo di stato **concreto** fisicamente esistente, almeno a livello macroscopico (natura non facit saltus).

Da un punto di vista concettuale, ma anche pratico, è però spesso molto appropriato considerare anche altri tipi di stati, che presentiamo nel seguito, e che rappresentano

astrazioni semplificative che tendono a cogliere aspetti informaticamente significativi della realtà.

Si noti che questi tipi di astrazioni sono tipicamente modellistici e informatici.

### 2.3.1.2 STATI CONTINUI A TEMPO DISCRETO

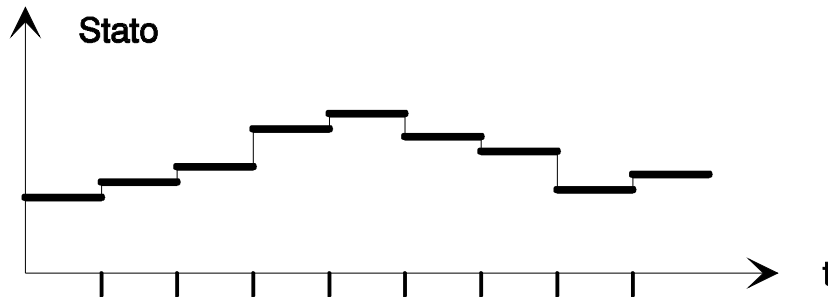


Fig. 2.9 - Andamento temporale di uno stato continuo a tempo discreto

Qui il termine "continui" è un pò improprio e da intendersi "quantitativi", cioè mappabili su un intervallo di numeri reali (in senso matematico).

Questi stati sono entità che possono assumere tutti gli **infiniti valori in un intervallo** ma tali valori variano "istantaneamente" (anche di un "delta" finito) solo negli intorno di istanti di tempo prefissati e a distanza temporale finita.

In corrispondenza di questi istanti di tempo si può parlare di **eventi** periodici (v. seguito).

L'esempio tipico è costituito dalle informazioni costituite dai **campioni di misure analogiche** con mantenimento di ordine zero.

Con buona *approssimazione* possiamo considerare rappresentative di stati di questa categoria, interni al calcolatore, anche le variabili di tipo "real" (o `float`).

Una caratterizzazione di questi stati può essere basata sul campo dei valori assumibili dallo stato, sulla descrizione degli istanti di variazione e sul campo di valori che può assumere il rapporto incrementale in ogni intervallo (*slew rate equivalente*).

Quando (come spesso avviene) questi stati sono assunti come approssimazione di stati continui, li si può caratterizzare con lo spettro di frequenze dello stato continuo da essi *idealmente* rappresentato.

### 2.3.1.3 STATI DISCRETI A TEMPO CONTINUO

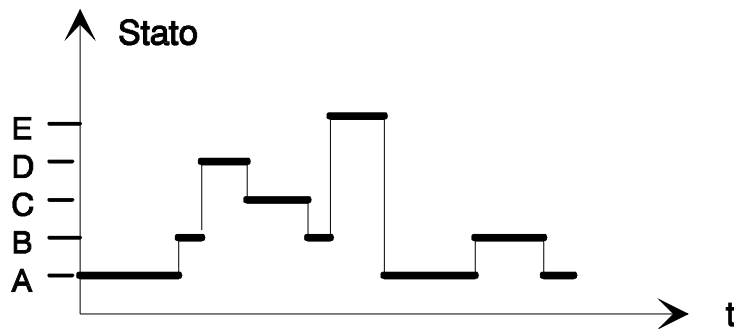


Fig. 2.10 - Andamento temporale di uno stato discreto a tempo continuo

Questi stati sono entità che possono assumere solo **valori appartenenti ad un insieme finito** enumerativo e che possono effettuare transizioni da un valore all'altro in **qualsiasi istante**.

Una caratterizzazione di questo tipo di stati può essere costituita dall'insieme finito dei valori assumibili, mentre per quanto riguarda il comportamento temporale la caratterizzazione può essere limitata, per semplicità, alla **minima durata** significativa di uno stato ("tempo morto").

Ulteriori precisazioni possono essere necessarie per specificare come si presentano gli spurii e come vadano considerati.

**Spurii** sono gli stati con durata minore di quella precisata e possono essere specificati come:

- "irrilevanti" se possono **indifferentemente** essere considerati o ignorati
- da "ignorare" se **devono** essere ignorati
- "malfunzionamenti" significativi di cui si **deve** tener conto.

Le transizioni costituiscono degli **eventi** ed una loro gestione "fisica" richiede che la distanza temporale tra due successivi eventi sullo stesso stato sia superiore al tempo morto (nel seguito definiti come eventi **sporadici**).

Il tempo morto può dipendere da proprietà fisiche dell'entità considerata (inerzia, velocità, ecc.) o da caratteristiche dell'osservatore da cui dipende la sua prontezza di percezione (come ad esempio la frequenza di campionamento, ecc.).

Per eventuali successioni di "microeventi" che non rispettino il vincolo del tempo morto in genere è possibile definire un *evento equivalente* (Transizione da stato iniziale a stato finale) e quindi "collassare" la successione in un solo evento. Questa trasformazione è tipica del passaggio da una visione *microscopica* ad una *macroscopica* di un fenomeno. Tipico esempio è costituito dai rimbalzi alla chiusura di un contatto meccanico.

Ad esempio per un contatto azionato a pulsante il tempo morto è il minimo tempo che l'operatore impiega tra due successive pressioni del pulsante (ad es. 100 ms), mentre i rimbalzi possono susseguirsi a distanza molto più breve (ad es. ogni 1 ms).

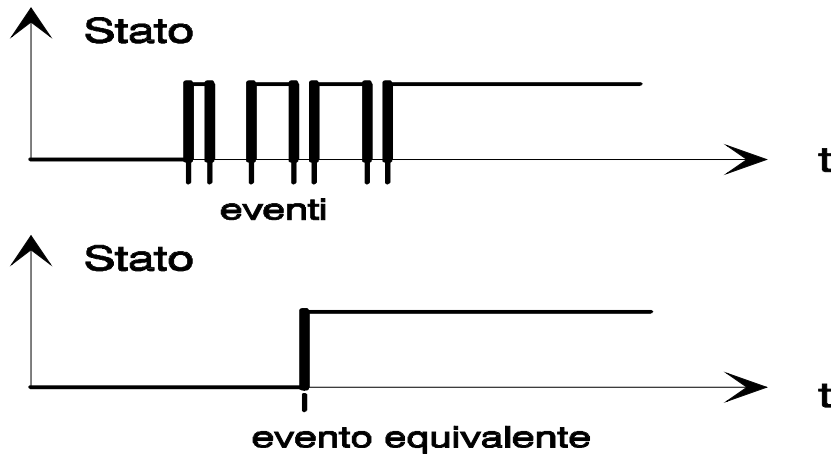


Fig. 2.11 - Sequenza di microeventi ed evento equivalente.

Un esempio generalmente basato su un modello con stati discreti a tempo continuo, è costituito dalle *reti logiche sequenziali asincrone*, per le quali i microeventi da collapsare sono le transizioni tra stati *instabili*.

#### 2.3.1.4 STATI DISCRETI A TEMPO DISCRETO

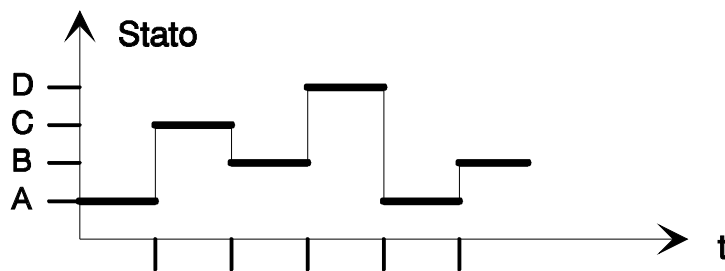


Fig. 2.12 - Andamento temporale di stato discreto a tempo discreto.

Il caso di stati discreti a tempo discreto è caratteristico di entità che possono assumere solo valori di un **insieme finito** enumerativo e che possono effettuare transizioni da un valore all'altro solo in **istanti di tempo prefissati e a distanza finita**.

Le transizioni costituiscono degli **eventi**.

La caratterizzazione di questi stati può essere basata sull'insieme dei valori che essi possono assumere e dalle proprietà temporali delle loro transizioni (v. seguito: eventi periodici, aperiodici, ecc.).

Un esempio tipico è costituito dalle *reti logiche sequenziali sincrone*.

Un altro esempio, all'interno dei calcolatori, è costituito dalle variabili di tipo "ordinale" (interi, caratteri, enumerativi, ecc.).

#### 2.3.2 FLUSSI DI EVENTI

Chiamiamo **flusso di eventi** un fenomeno caratterizzato da **stati discreti** nel tempo e/o nei valori. Ogni transizione tra stati significativi costituisce un "evento" generato dal fenomeno.

Per definizione di stati discreti gli eventi sono da considerare di *durata temporale nulla*, quindi associabili ad **istanti**.

Ogni variabile di stato discreto costituisce un potenziale flusso di eventi, e lo diventa di fatto se sono previsti, almeno concettualmente, meccanismi di "percezione" delle transizioni, ed è proprio a tali transizioni che si associano informazioni significative. Sul piano realizzativo ciò comporta l'impiego di dispositivi in grado di trasformare eventi in **azioni** (per esempio generare una richiesta di interruzione).

Gli eventi sono **entità concettuali**, dato che in realtà non si può verificare una transizione (finita) in un tempo infinitesimo. Tuttavia si può adottare la solita assunzione di considerare infinitesimo un intervallo di tempo che sia *molto minore della granularità temporale caratteristica del contesto di fenomeni considerato*.

### 2.3.2.1 SORGENTI DI EVENTI

- EVENTI TEMPORALI

Un modello di tempo discreto costituisce un flusso **astratto** ed **implicito** di eventi detti "eventi temporali".

Concretamente si tratta di eventi associati alle transizioni di una variabile di stato di un fenomeno di tipo "orologio".

In questo caso lo stato è discreto a tempo discreto e le sue transizioni avvengono in corrispondenza dei "tick" dell'orologio.

- EVENTI INTERNI ED ESTERNI

Quando si considerano (sotto)sistemi tra loro interagenti, rispetto ad ogni sottosistema gli eventi (o, meglio, i flussi di eventi, cioè gli stati) sono classificabili come "interni" ed "esterni".

Le interazioni tra i sottosistemi si manifestano come azioni e si traducono in rapporti "causa-effetto" tra eventi dei flussi esterni ed eventi dei flussi interni.

Compito delle **interfacce** è tipicamente quello di creare catene di cause-effetti che in ingresso generano stati ed eventi interni che rappresentino correttamente gli aspetti **significativi** di stati ed eventi esterni, e viceversa in uscita.

### 2.3.2.2 CONTENUTO INFORMATIVO DEGLI EVENTI

Rispetto al contenuto informativo gli eventi possono essere classificati in "incrementali" ed "assoluti".

## • EVENTI INCREMENTALI - EI

Sono descrivibili mediante 3-tuple:

$$EI_i = [FL, Ci, Ti]$$

$EI_i$  i-esima occorrenza dell'evento

FL flusso di eventi

$Ci$  simbolo qualificatore dell'evento - valore dell'evento

$Ti$  istante di tempo dell'occorrenza dell'evento.

Gli eventi incrementali sono rappresentati in una forma che consente di calcolare (cioè dedurre con opportuna funzione  $F(\ )$ ) lo stato  $S_{i+1}$  seguente alla transizione **solo** se è noto anche lo stato precedente  $S_i$ .

$$S_{i+1} = F(EI_i, S_i)$$

Esempi caratteristici di eventi incrementali sono ad esempio i seguenti.

- Il sopraggiungere di caratteri nella ricezione di un messaggio. L'automa riconoscitore del messaggio esegue transizioni ad ogni carattere ricevuto, e tali transizioni portano ad uno stato corretto solo se era corretto anche lo stato precedente.
- Gli impulsi di un encoder incrementale (o, in generale, impulsi destinati ad un conteggio), ognuno dei quali incrementa (o decrementa) una variabile di stato che rappresenta una posizione angolare (o un conteggio di pezzi, ecc.) e che ad ogni evento porta ad un nuovo valore corretto solo se lo era il precedente.

La "**perdita**" di eventi incrementali costituisce un **errore "fatale"**, da recuperarsi eventualmente con opportune tecniche di *risincronizzazione* basate su eventi assoluti, come verrà esemplificato tra breve.

## • EVENTI ASSOLUTI - EA

Sono descrivibili mediante 4-tuple:

$$EA_i = [FL, S_i, S_{i+1}, Ti]$$

$EA_i$  i-esima occorrenza dell'evento

FL flusso di eventi

$S_i$  stato precedente alla transizione | valore

$S_{i+1}$  stato seguente alla transizione | dell'evento

$Ti$  istante di tempo dell'occorrenza dell'evento.

Si noti che in molti casi gli eventi significativi sono caratterizzati dal fatto che  $S_i$  risulta essere sempre lo stato *di riposo* (oppure è considerato irrilevante), e in questi casi può essere sottinteso.

Gli eventi assoluti sono rappresentati in una forma che consente di calcolare correttamente lo stato seguente alla transizione *anche senza conoscere lo stato precedente*.

$$S_{i+1} = F(EA_i)$$

Esempi di eventi in forma assoluta sono i campionamenti di segnali (analogici o digitali), cioè successioni di "fotografie" di stati.

La **perdita** di eventi assoluti produce un **degrado informativo** della storia temporale, ma non la perdita definitiva di sincronizzazione. Infatti al prossimo evento assoluto recepito il sistema è in grado di portarsi nello stato corretto ponendo fine alla situazione di inconsistenza. Una perdita di eventi assoluti è quindi accettabile entro certi limiti dipendenti dalla gravità dell'instaurarsi e del perdurare delle situazioni inconsistenti.

Come si è accennato, gli eventi assoluti possono essere usati per effettuare risincronizzazioni.

Ad esempio consideriamo la trasmissione di

- **messaggi costituiti da caratteri ASCII.**

Un carattere SOH (Start Of Header, codice ASCII = 01) costituisce un **evento assoluto** perchè, non comparando in alcun altro contesto, identifica univocamente l'inizio di un nuovo messaggio, consentendo così la sincronizzazione del ricevente che si porta nello stato "inizio messaggio".

Consideriamo ora invece una comunicazione di

- **valori binari trasmessi come successione di byte.**

Un byte di valore pari al codice di SOH può ancora costituire l'inizio di un messaggio, ma in questo caso lo stesso valore può essere assunto anche da altri byte interni al messaggio, e costituisce quindi un **evento incrementale** che il ricevente interpreta correttamente solo se si trova già in uno stato corretto. La perdita di un byte rende non solo inutile, perchè incompleto, il messaggio ricevuto, ma rende impossibile la certezza di riconoscimento dell'inizio di un successivo messaggio.

In questo caso una tecnica robusta per recuperare perdite di sincronizzazione consiste nel progettare il trasmettitore in modo che lasci tra un messaggio ed il successivo una pausa di durata maggiore di un opportuno valore temporale TO e nell'adottare presso il ricevente l'evento assoluto **scadenza di un time-out** (cioè di un intervallo di TO ms senza ricezione di byte) come meccanismo che forza nel ricevente lo stato "attesa di SOH".

Analogamente, nel caso di encoder incrementali, si fa in modo che periodicamente si verifichino eventi assoluti detti in gergo *impostazione dello zero-macchina*, che garantiscano che ogni operazione rilevante inizi da uno stato corretto, recuperando eventuali errori dovuti a precedenti perdite di eventi incrementali.

---

- **NOTA - Scelta tra rappresentazione incrementale e assoluta di eventi.**

In alcuni casi il progettista può scegliere se rappresentare degli eventi in forma incrementale o assoluta. Il principale, o forse unico, vantaggio della forma incrementale è costituito dalla maggior compattezza della rappresentazione delle informazioni.

Ad esempio consideriamo il caso di eventi periodici costituiti dalla successione di campioni di una grandezza analogica lentamente variabile e rappresentata con interi di 16 bit..



La soluzione ad *eventi assoluti* comporta la rappresentazione di ogni campione con 2 byte. La perdita di un campione costituisce un degrado non catastrofico che verrà recuperato al ciclo seguente con il campione successivo.

La soluzione con *eventi incrementali* può essere basata sulla rappresentazione delle **variazioni**, con segno, di ogni valore rispetto a quello precedente. Tali variazioni richiedono un numero di bit dipendente dalla massima velocità di variazione della grandezza e con le usuali scelte della frequenza di campionamento è in genere sufficiente un solo byte, dimezzando così la quantità di informazione da comunicare. In questo caso però la perdita anche di un solo evento porta ad un errore, nel valore ricostruito, che si mantiene fino alla prossima eventuale operazione di recupero.

Le operazioni di recupero, a loro volta, possono essere spontanee e basate su una periodica generazione di eventi assoluti di sincronizzazione, oppure sollecitate dal processo consumatore di eventi che quindi deve essere in grado di riconoscerne l'avvenuta perdita e di comunicare a ritroso verso il processo produttore per sollecitarne l'invio di un evento assoluto di riconoscimento.

E' il classico problema di scegliere se comunicare i **valori** o solo le loro **variazioni**: la comunicazione dei valori comporta una **ridondanza** che però costituisce un elemento di **robustezza**, mentre la scelta di comunicare le variazioni privilegia la **compattezza**.

Si preferisce quindi la soluzione di ripetere ciclicamente la comunicazione dei valori nelle applicazioni che richiedono robustezza e predicibilità, quando gli eventi sono periodici e frequenti ed il meccanismo di comunicazione è piuttosto "falloso".

Si potrà invece adottare la soluzione incrementale, cioè di comunicare solo le variazioni, nei casi meno critici e per i quali è importante un risparmio di risorse, in presenza di eventi poco frequenti e con canale di comunicazione corretto o almeno dotato di meccanismi di risincronizzazione.

---

### 2.3.2.3 PROPRIETA' TEMPORALI DEGLI EVENTI

I flussi di eventi sono classificabili in base alle proprietà temporali dei corrispondenti eventi.

#### 2.3.2.3.1 FLUSSI DI EVENTI APERIODICI

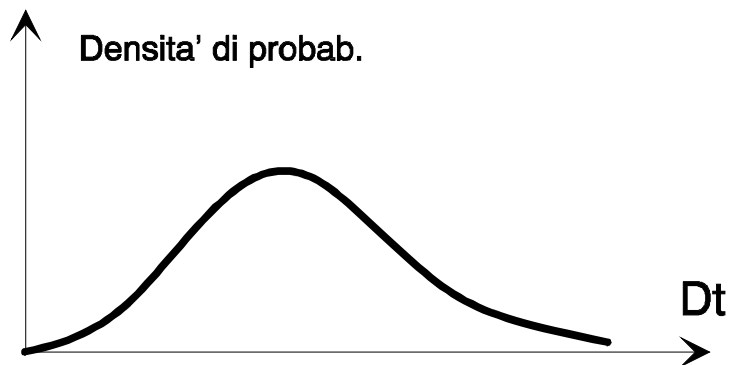


Fig. 2.13 - Densità di probabilità di eventi aperiodici consecutivi a distanza  $Dt$ .

Sono flussi i cui eventi si verificano in **istanti di tempo imprevedibili**, talvolta caratterizzabili solo con una densità di probabilità in funzione dell'intervallo  $Dt$  tra un evento ed il successivo.

Questi flussi di eventi sono significativi sul piano concettuale per la completezza della classificazione (costituiscono il caso più generale) e come limite del caso seguente (eventi sporadici).

Un esempio di flussi di eventi aperiodici è costituito dalle emissioni di particelle di corpi radioattivi, nelle quali non esiste un estremo inferiore finito  $> 0$  per l'intervallo tra un'emissione e la successiva.

#### 2.3.2.3.2 FLUSSI DI EVENTI SPORADICI

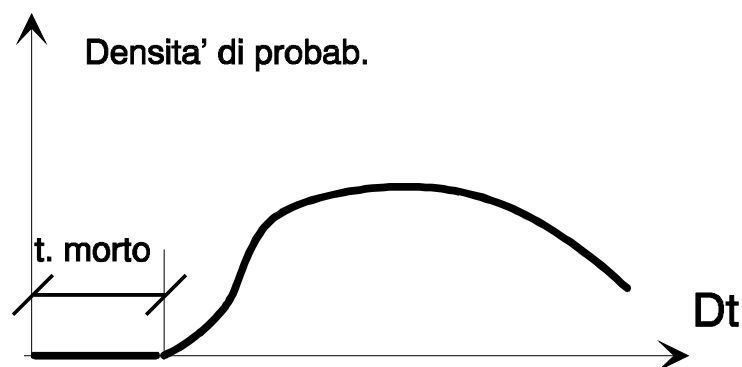


Fig. 2.14 - Densità di probabilità di eventi sporadici consecutivi a distanza  $Dt$ .

I flussi di eventi sporadici sono un caso particolare di flussi di eventi aperiodici, in cui si ha densità di probabilità nulla per intervalli minori di un valore  $DT_{min}$  (*tempo morto*).

Per questi flussi è garantito che dopo il verificarsi di un evento non se ne verificherà uno successivo per un intervallo di tempo  $DT_{min}$ . Il mancato rispetto di questo vincolo temporale qualifica un evento come spurio, cioè generato da un disturbo e non dal fenomeno in esame.

Flussi di eventi sporadici sono gli **stati discreti a tempo continuo**.

### 2.3.2.3 FLUSSI DI EVENTI PERIODICI

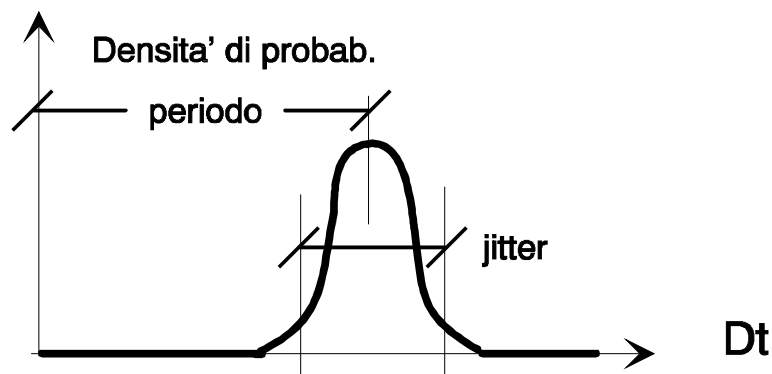


Fig. 2.15 - Densità di probabilità di eventi periodici consecutivi a distanza  $Dt$ .

Sono flussi i cui eventi si verificano a **cadenza regolare** e sono caratterizzati dal valore temporale del periodo ( $P$ ) ed eventualmente dalla irregolarità ("jitter") di tale periodo.

Questi flussi possono essere concettualmente considerati un caso particolare dei flussi di eventi sporadici per cui la densità di probabilità è un impulso per  $Dt = P$  che li caratterizza come "fortemente prevedibili".

Anche per questi eventi è opportuno che venga specificato il significato del mancato rispetto delle caratterizzazioni temporali nominali (spurii).

Flussi di questo tipo sono gli **stati a tempo discreto**.

Come si è già notato, il campionamento di uno stato (tipicamente, ma non solo, continuo) costituisce un flusso di eventi. A seconda che il campionamento sia regolare o meno si avrà un flusso periodico o sporadico.

#### 2.3.2.4 CORRELAZIONI TRA DIVERSI FLUSSI DI EVENTI

- EVENTI MUTUAMENTE APERIODICI

Sono eventi appartenenti a diversi flussi che possono essere considerati "scorrelati" in quanto i fenomeni che li generano non sono (facilmente) correlabili tra loro.

Possono essere mutuamente aperiodici eventi di flussi aperiodici, sporadici o periodici.

- EVENTI MUTUAMENTE CORRELATI

Sono eventi di diversi flussi (aperiodici o più spesso sporadici o periodici) tra i quali si può individuare una correlazione tale da consentire un elevato grado di predicibilità. Ciò si verifica tipicamente quando esiste un rapporto *causa-effetto* tra due fenomeni, così da instaurare una relazione temporale tra gli eventi dell'uno e quelli dell'altro.

E' molto importante notare che spesso non si forniscono descrizioni delle relazioni temporali tra eventi di diversi flussi. Tali eventi sono quindi considerati per difetto "mutuamente aperiodici".

In alcuni casi è invece relativamente semplice descrivere la correlazione temporale tra eventi appartenenti a diversi flussi, e tale conoscenza può essere utilmente usata per migliorare le prestazioni del sistema di elaborazione sfruttando la maggior prevedibilità di alcuni eventi in modo da prevenire situazioni non volute.

Ad esempio gli eventi (non desiderabili) *sovracorrente in un motore* e *intervento della protezione* sono eventi singolarmente sporadici, ma mutuamente correlati a meno che, come è opportuno che si faccia, non venga aperto in tempo il teleruttore di alimentazione del motore stesso.

## 2.4 AZIONI ED ATTIVITA'

### 2.4.1 AZIONI - COMPORTAMENTO REATTIVO

AZIONE : EVENTO --> EVENTO

Chiameremo "**azione**" una unità di "lavoro" di durata finita, attivata (*triggered*) da un evento e contrassegnata da eventi di inizio (*start*) e di fine (*stop*), la cui distanza temporale corrisponde alla durata dell'azione. Tale durata è in genere considerata piccola, spesso trascurabile, rispetto alla scala temporale dei fenomeni significativi (granularità temporale caratteristica), e quindi a livello temporalmente macroscopico le azioni sono spesso considerabili come istantanee e, come tali, associate all'istante in cui si verifica l'evento di stop. Le durate delle azioni non sono però trascurabili quando si entra nel merito della valutazione dei tempi di risposta, del carico di lavoro della CPU e quindi dello *scheduling*.

Si dice **scopo temporale** di un'azione l'intervallo di tempo tra l'istante in cui si verifica l'evento attivante (trigger) e l'istante in cui si verifica l'evento di stop.

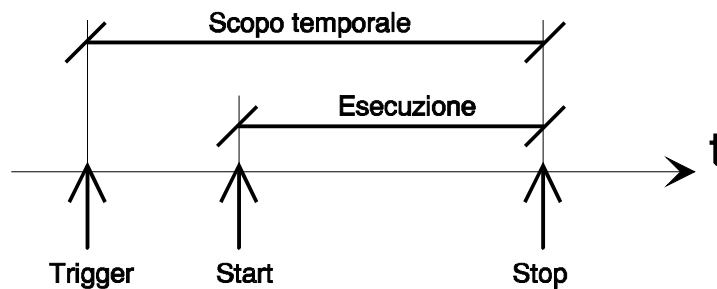


Fig. 2.16 - Scopo temporale e tempo di esecuzione di un'azione sull'asse dei tempi assoluti.

Le azioni possono essere considerate "primitive" o "composite", a seconda del dettaglio che si è interessati ad evidenziare. Le azioni composite sono costituite da ordinamenti parziali di altre azioni, a loro volta primitive o composite.

Le azioni provocano transizioni di stato nel sistema su cui agiscono. Si può quindi considerare un rapporto causa-effetto tra l'evento associato all'azione (stop) e l'evento transizione di stato del sistema.

Un'azione è cioè associabile ad un evento che la "produce" (evento di trigger), di cui può essere considerata effetto, e ad un evento che essa "genera", di cui può essere considerata causa.

Sono considerabili tipicamente come azioni le esecuzioni di istruzioni da parte dei calcolatori, rispetto ai quali costituiscono eventi "interni".

## 2.4.2 ATTIVITA' - COMPORTAMENTO TRASFORMAZIONALE

ATTIVITA' : STATO -> STATO

Chiameremo “**attività**” un comportamento che presenta una durata concettualmente non limitata, e comunque lunga rispetto ai tempi in gioco, durante la quale tale comportamento è caratterizzabile da una "funzione di trasferimento" (in senso lato di funzione). Le attività sono caratteristiche dei sistemi a "funzionalità intrinseca".

L'effetto di un'attività è l'evoluzione di uno o più stati in uscita in funzione dell'evoluzione di uno o più stati in ingresso.

E' significativo assumere un'analogia con il teorema di Shannon del campionamento e considerare che una successione di azioni ciclicamente ripetute sia assimilabile ad un'attività, quando il periodo di ripetizione di tali azioni sia molto breve rispetto alla scala temporale significativa per i fenomeni considerati.

Con questa assunzione anche una macchina programmabile ad istruzioni, e quindi a comportamento discreto, può essere considerata svolgere un'attività “continua”.

In altre parole è considerabile come attività anche una ripetizione ciclica di azioni operanti ripetutamente sulle stesse uscite, quando si sia interessati agli effetti globali e non a quelli di ogni singola azione.

### **Esempio. - Regolatore.**

Un regolatore realizzato a software può essere basato sulla ripetizione ciclica delle seguenti azioni:

Acquisizione del valore del riferimento.

Acquisizione del valore corrente della misura.

Calcolo del valore di stato da emettere.

Emissione sulla grandezza manipolata del valore calcolato.

L'effetto “visibile” di una sufficientemente frequente ripetizione di tali azioni è una regolazione “continua”.

## 2.5 CARATTERIZZAZIONE DEL MONDO ESTERNO

Per caratterizzare il mondo esterno:

- Si evidenziano I MODI e le FASI dell'evoluzione del sistema.
  - Presentazione organica ed aggregata delle singole caratterizzazioni di stati ed eventi.
  - Si evidenziano le CORRELAZIONI tra diversi stati ed eventi.
- Si descrivono le interfacce, con informazioni e relativi segnali.
- Si forniscono caratteristiche GLOBALI di tipo ambientale (disturbi meccanici, chimici, fisici (temperatura, umidità) ed elettrici)

## 2.5.1 MODI FASI E PASSI DEI SISTEMI

### MODI

Per **modo** di un sistema possiamo intendere un insieme aggregato di stati caratterizzati dall'essere finalizzati ad un particolare obiettivo e dal fatto che in ognuno di essi sono abilitate o meno diverse operatività, allarmi, ecc.

I modi evidenziabili nel comportamento di un sistema non sono universali, ma dipendono dal particolare sistema e dall'impostazione di progetto. In molti casi si adotta una visione basata sui seguenti modi.

#### FERMO

Il sistema è a riposo ed è caratterizzato da un unico stato.

#### MARCIA MANUALE

Il sistema è asservito ai comandi di un operatore che ne guida a piacere l'evoluzione, generalmente per effettuare delle prove di funzionamento o per predisporlo ad altri modi o per gestire situazioni anomale.

#### MARCIA AUTOMATICA

Il sistema evolve interagendo con i sistemi di controllo, secondo le funzionalità previste per il normale funzionamento.

#### MANUTENZIONE

Il sistema è predisposto per facilitare operazioni di diagnosi di corretta funzionalità o sostituzioni e riparazioni per eliminare guasti e malfunzionamenti.

### FASI

Possiamo chiamare **fasi** di un sistema gli stati, all'interno di un particolare modo, che sono generalmente raggiunti seguendo una sequenza relativamente rigida. La suddivisione di un modo in fasi è caratteristica dei processi discreti, mentre nei processi continui si possono generalmente distinguere solo le fasi di avviamento, regime e arresto.

### PASSI

Infine può essere significativo evidenziare, all'interno delle fasi più articolate dei processi discreti, delle successioni di **passi**.

Naturalmente si potranno individuare "sottofasi" e "sottopassi" introducendo nella descrizione del comportamento del sistema il numero opportuno (non necessariamente pari a 3, come qui proposto) di livelli gerarchici di raffinamento.

## 2.5.2 INFORMAZIONI DI INTERFACCIA

Aspetto essenziale per una corretta progettazione di ogni applicazione è la descrizione delle interfacce tra i vari elementi. In particolare nelle applicazioni informatiche per l'automazione, la grande varietà di comportamenti e di requisiti dei vari dispositivi interessati richiede una particolare attenzione alla completezza, rigorosità e consistenza di tali descrizioni.

Mentre si rimanda al capitolo dedicato all'interfacciamento per un'analisi più dettagliata e per le tecniche implementative, nel seguito si delineano sinteticamente gli aspetti più generali e globali di cui tener conto in sede di specifica e progetto di un (sotto)sistema informatico di automazione.

### 2.5.2.1 INFORMAZIONI DESTINATE AL CALCOLATORE

Questa caratterizzazione consiste nell'elenco delle informazioni prodotte dal sottosistema da controllare ed in ingresso verso il sistema di calcolo, con la precisazione delle caratteristiche ad alto livello di astrazione:

- eventi, significato e loro successione temporale;
- stati, significato, durata, tipo, campo di valori.

Scopo principale di questa descrizione è verificare la completezza e congruenza delle informazioni disponibili rispetto alle elaborazioni da eseguire, ed una valutazione del "carico di lavoro" (workload) imposto ai sottosistemi di calcolo.

### 2.5.2.2 COMANDI EMESSI DAL CALCOLATORE

Il quadro delle interazioni tra sottosistema da controllare e sottosistema di calcolo va completato con le informazioni in uscita da quest'ultimo, spesso dette "comandi".

E' importante inoltre caratterizzare il comportamento del sottosistema da controllare rispetto ai comandi, mediante qualche opportuna forma di modellizzazione.

### 2.5.2.3 SEGNALI FISICI

Ad un successivo livello di dettaglio occorre precisare i fenomeni fisici che costituiscono il veicolo delle informazioni scambiate dai due sottosistemi nei due sensi. Tali fenomeni fisici sono detti "**segnali**" e si deve specificare la **forma** con cui essi **rappresentano** le informazioni, che possono essere associate agli stati o agli eventi.

## 2.5.3 SPECIFICHE DI PRECISIONE

Precisioni, disturbi e tolleranze della strumentazione e dei modelli dei processi fisici.

Un ulteriore livello di dettaglio riguarda:

- le precisioni fornite dalla strumentazione prevista;
- la caratterizzazione dei disturbi previsti sui segnali;
- la precisione con cui sono noti i modelli dei sottosistemi controllati;
- le tolleranze accettabili sui comandi.

Va sottolineata l'importanza anche di queste ultime caratterizzazioni per una corretta impostazione del progetto ingegneristico dei sottosistemi di controllo perchè condiziona la scelta di granularità adatte, nelle operazioni di discretizzazione, e di tecniche e dispositivi congrui.



Le caratterizzazioni sopra citate sono generalmente espresse dall'esperto del sistema da controllare che però deve possedere attitudini interdisciplinari in particolare sulla strumentazione e relativo interfacciamento con elaboratori digitali.

## 2.6 CARATTERIZZAZIONE DELLE PRESTAZIONI TEMPORALI RICHIESTE

Descrivono i requisiti temporali che una corretta automazione impone al sottosistema di elaborazione.

Funzioni Validità (t) della risposta a flussi di eventi che consentono di definire Urgenza, Importanza, Criticità

**L'esperto dell'applicazione** deve tipicamente fornire una descrizione dei requisiti temporali che il sottosistema di elaborazione deve rispettare per ottenere dal sistema complessivo il comportamento che costituisce, in ultima analisi, lo scopo finale della realizzazione.

Tali requisiti saranno dedotti dal comportamento desiderato del sistema applicativo globale e pongono vincoli temporali di tipo "end-to-end" alle elaborazioni svolte dal calcolatore di automazione. In altre parole i requisiti sul (sotto)sistema di automazione riguardano la correlazione temporale tra stimoli in ingresso al calcolatore e risposte finali emesse dal calcolatore, senza entrare nel merito della scomposizione delle elaborazioni complessive in diverse azioni parziali all'interno dell'elaboratore.

Per maggiore semplicità nel seguito considereremo la caratterizzazione "disaggregata", cioè considerando indipendenti tra loro i diversi eventi di stimolo. Con questa semplificazione il comportamento temporale globale richiesto consiste in un insieme di requisiti che correlano ogni particolare risposta al suo stimolo.

Ognuno di questi requisiti può assumere una delle forme seguenti.

a) - **Viene specificata la frequenza di campionamento** di uno stato, **irregolarità** massima nel periodo e **ritardo** massimo di emissione della risposta. In molti casi si assume convenzionalmente come accettabile che il ritardo massimo della risposta sia pari al periodo di campionamento specificato.

b) - **Viene specificata la scadenza temporale** (*deadline*) che limita l'intervallo temporale entro cui deve essere fornita la risposta allo stimolo. Questa caratterizzazione è tipicamente fornita per le azioni attivate da eventi sporadici, e costituisce una forma semplificata di specifiche del tipo seguente (c).

c) - **Viene specificata la funzione "validità temporale"**, che esprime l'"utilità" della risposta ad un evento in funzione del tempo intercorrente tra l'istante in cui si è verificato l'evento di "stimolo" e l'istante dell'evento di *stop* dell'azione che fornisce tale risposta.

NOTA. Se consideriamo le scadenze di intervalli regolari di tempo, cioè i *tick dell'orologio*, come se fossero eventi esterni che attivano azioni di campionamento, i requisiti di tipo a) possono assumere una forma analoga a quelli di tipo b).

La forma c), più articolata, merita qualche ulteriore commento.

La motivazione di questa forma è la seguente: precisare meglio come decade il valore di una risposta allo stimolo, sia per piccoli ritardi rispetto alla scadenza nominale, che per ritardi più elevati.

Nell'ipotesi semplificativa, ma spesso ragionevolmente applicabile, che lo stimolo che richiede una risposta sia costituito da ogni singolo evento di un unico flusso, la funzione **validità temporale** risulta una funzione di una sola variabile tempo, la cui *origine* è collocata nell'istante in cui si verifica l'evento.

Per una breve analisi dei tipi di funzioni validità che caratterizzano diversi requisiti temporali assumeremo la semplificazione di funzioni costituite da spezzate.

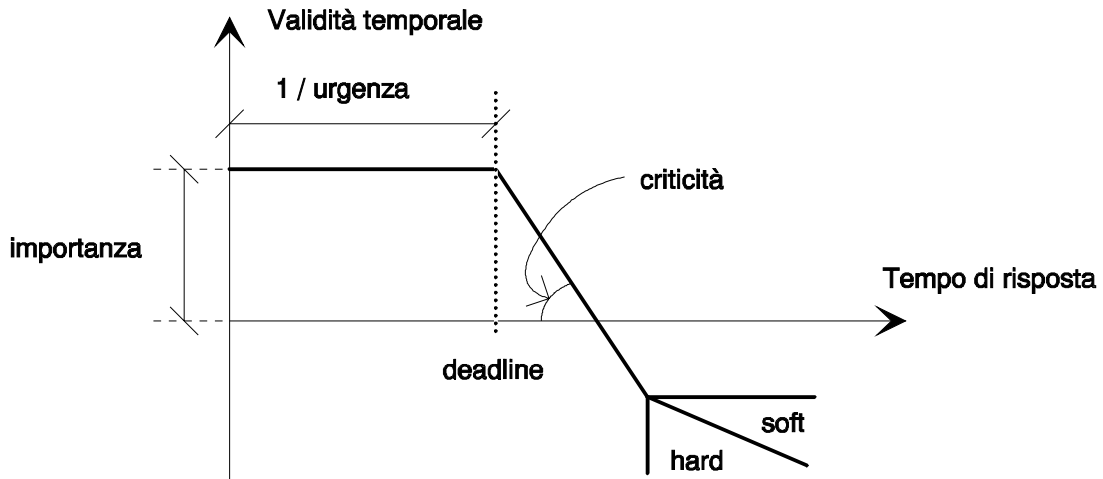


Fig.2.17 - Validità temporale di una risposta allo stimolo verificatosi nell'origine.

Nella fig. 2.17 si possono individuare gli elementi seguenti.

L'ordinata del primo tratto orizzontale esprime l'**importanza** della risposta.

La durata di questo tratto rappresenta l'inverso dell'**urgenza**, ed il suo estremo destro è detto **deadline** (scadenza).

La pendenza del tratto seguente esprime la **criticità** temporale della risposta e descrive come decade il valore della risposta per ritardi che superano "marginalmente" la *deadline*.

I valori negativi rappresentano il danno, che può stabilizzarsi (*soft deadline*), crescere o essere catastrofico (*hard deadline*), e questo tratto finale descrive l'entità del danno in caso di risposta con notevole ritardo rispetto alla *deadline*.

In alcuni casi un anticipo nell'emissione della risposta rispetto al tempo ideale può essere tanto dannoso quanto un ritardo. Un esempio è costituito dal comando per la scintilla di accensione in un motore a scoppio, correlato all'istante di passaggio dell'albero motore per una posizione angolare di riferimento.

Per questi casi la funzione di Validità temporale è non monotona, e può essere rappresentata come nella figura seguente (fig. 2.18).

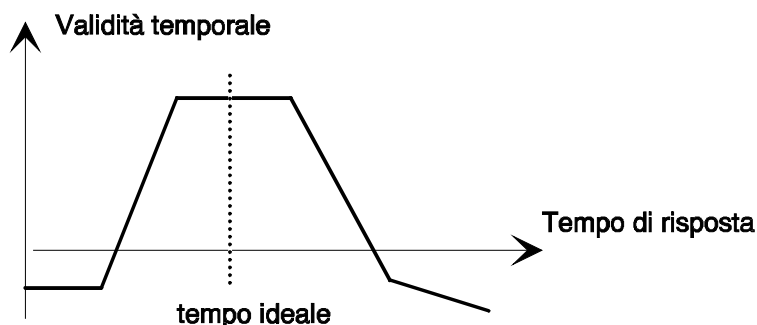


Fig. 2.18 - Funzione Validità temporale per il caso in cui un anticipo della risposta è dannoso come un ritardo.

In base alle funzioni “validità temporale” delle varie azioni è possibile effettuare una schedulazione più raffinata di tali azioni nel caso di sistema con elevato carico di lavoro (*overloaded*), in cui possono essere occasionalmente e marginalmente mancate alcune scadenze.

## 2.7 CARATTERIZZAZIONE DELLE PRESTAZIONI TEMPORALI OFFERTE DALLE AZIONI

Descrivono le proprietà temporali del comportamento di un particolare sistema di elaborazione. I risultati sono forniti da azioni che risultano dall'esecuzione di opportuni algoritmi

Si possono distinguere due diversi tipi di prestazioni relative a diverse velocità di elaborazione. Tale distinzione è molto importante e deve essere ben meditata.

### **PRESTAZIONI DI CADENZA** (*throughput, input/output rate*).

E' la capacità di accettare ed elaborare una elevata quantità di informazioni (eventi) per unità di tempo.

Costituisce una prestazione interessante quando si è interessati alla quantità di elaborazione effettuabile nel tempo. Può essere incrementata con tecniche di *pipeline*.

### **PRESTAZIONI DI RAPIDITA' DI RISPOSTA** (*response time*).

E' la capacità di emettere rapidamente le risposte a stimoli provenienti dall'esterno anche in istanti non prevedibili.

Questa prestazione è significativa per i sistemi real-time, che devono interagire con fenomeni esterni dotati di una propria evoluzione temporale. Dato che questi sono i casi che qui più ci interessano, li approfondiamo ulteriormente.

E' interessante descrivere, anche qui per semplicità in forma "disaggregata" per ogni singola azione, le prestazioni offerte dal sistema di elaborazione, sia in termini di "bontà del valore offerto" del risultato calcolato, sia in termini di istanti temporali in cui tali risultati sono resi disponibili.

Queste descrizioni sono tipicamente di **competenza dell'esperto di applicazioni informatiche** e sarebbe auspicabile che possano essere prodotte automaticamente da opportuni strumenti informatici in grado di rilevare ("profiler") o calcolare (analizzatori di codice eseguibile) tali prestazioni.

Con la semplificazione che ogni elaborazione sia richiesta da un evento e fornisca al termine un risultato, potremo descrivere il comportamento di ogni elaborazione mediante una funzione che correla la bontà del risultato con il "tempo dedicato alla produzione" del risultato (si noti bene, il *tempo di elaborazione netto* non il tempo assoluto).

NOTA. Il tempo dedicato alla produzione del risultato coincide con il tempo assoluto solo nel caso ideale in cui l'elaborazione considerata sia l'unica attività svolta dal sistema durante quell'intervallo di tempo e quindi non sia ritardata o interrotta da altre elaborazioni o da interrupt (cioè non soffra di alcuna latenza), come invece avviene normalmente nei sistemi *multitask*.

### 2.7.1 ALGORITMI NORMALI SINGOLI

In questa categoria ricadono i normali algoritmi che forniscono un risultato significativo solo dopo una sequenza di operazioni univocamente o statisticamente determinata. Nel caso di distribuzione statistica dei tempi di esecuzione occorre che esista comunque un limite superiore (*upper bound*) che normalmente viene assunto come valore da utilizzare come indicativo delle prestazioni "garantite" e per la verifica di correttezza temporale (caso peggiore).

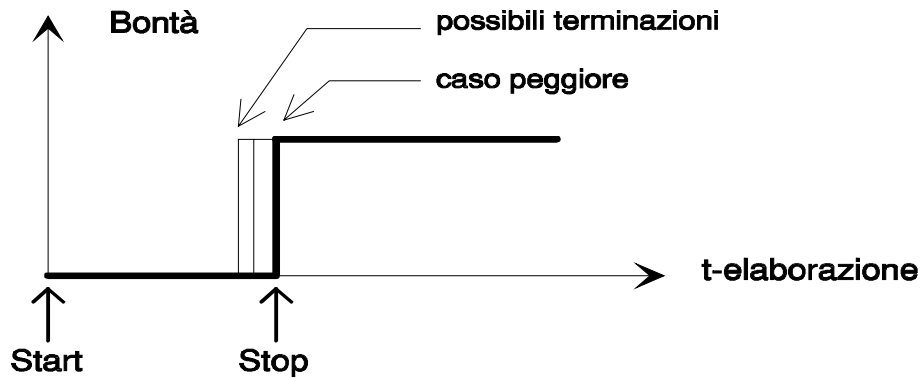


Fig. 2.19 - Bontà di un algoritmo normale in funzione del tempo di elaborazione.

## 2.7.2 ALGORITMI NORMALI MULTIPLI

Chiamiamo algoritmi multipli quelli per cui ad ogni occorrenza di uno stimolo è possibile scegliere, *prima di iniziare le elaborazioni* relative, tra diversi algoritmi semplici che si differenziano tra loro per la diversa precisione dei risultati ottenibili (bontà) e la diversa quantità di tempo di calcolo richiesto per ottenerli. Ovviamente in una collezione significativa gli algoritmi che forniscono risultati meno precisi saranno anche i più veloci.

Con questo approccio sarà possibile dinamicamente scegliere il miglior algoritmo da "attivare" compatibilmente con la disponibilità temporale della CPU al momento dell'evento di start.

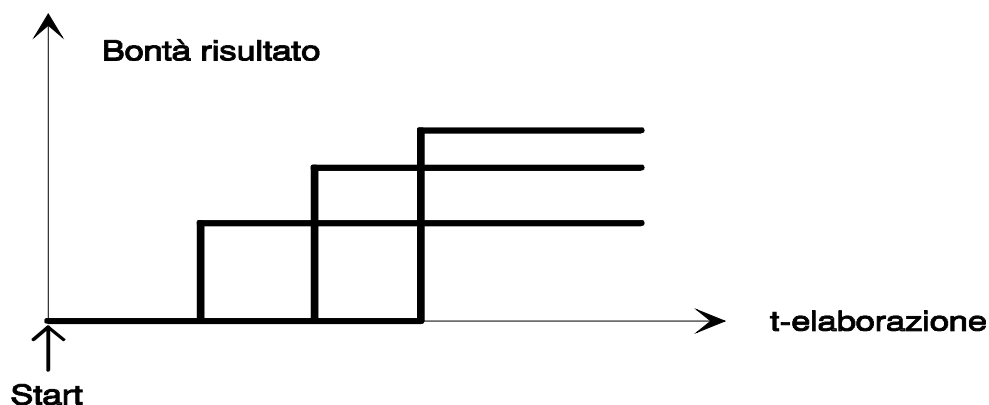


Fig. 2.20 - Bontà di algoritmi multipli in funzione del tempo di elaborazione.

## 2.7.3 ALGORITMI AD APPROSSIMAZIONI SUCCESSIVE

Alcuni algoritmi godono della proprietà di fornire risultati approssimati in tempi relativamente brevi e di procedere ad un progressivo miglioramento della precisione ad ogni successivo ciclo di esecuzione che venga loro concesso.

Un aspetto molto interessante di questi algoritmi è costituito dal fatto che è possibile ottenere dei risultati "meglio che niente" in tempi brevi e di poter scegliere *in ogni istante* se procedere a migliorare l'approssimazione o, se è scaduto il tempo disponibile, adottare come risultato l'ultimo valore calcolato, ottenendo così doti di robustezza e "*graceful degradation*" (degrado "morbido") delle prestazioni offerte, anche in situazioni critiche e di sovraccarico del processore (*overload*).

E' ovvio aspettarsi che un buon algoritmo ad approssimazioni successive presenti una funzione "bontà del risultato" di tipo monotono non decrescente.

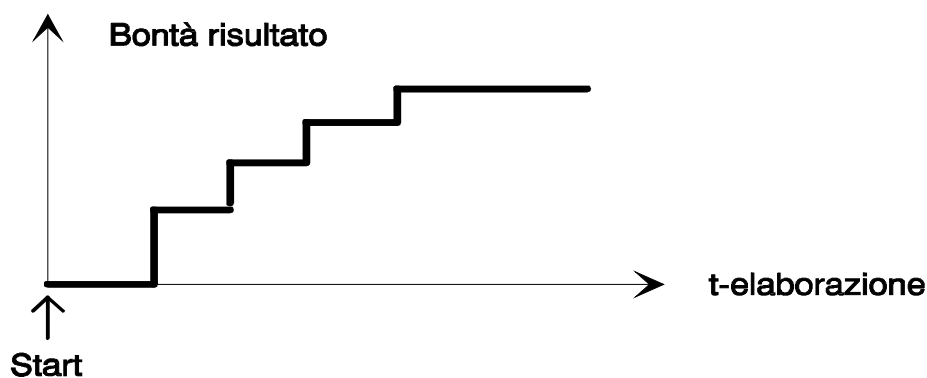


Fig. 2.21 - Bontà di un algoritmo ad approssimazioni successive in funzione del tempo di elaborazione.

## 2.8 CARICO DI LAVORO DELL'ELABORATORE

Per caratterizzare l'impegno richiesto al sistema di elaborazione è significativo considerare il carico di lavoro (*workload*) "aggregato", cioè globalmente derivante dalla frequenza di ripetizione delle azioni periodiche, dalla sequenza di eventi del mondo esterno, dai relativi requisiti di comportamento temporale e dagli impegni di elaborazione associati (tempi di esecuzione).

Si noti che il carico di lavoro deriva dalla composizione dell'insieme delle prestazioni richieste e delle relative prestazioni offerte, a cui va sommato un "*overhead*" di gestione interna, costituito tipicamente dall'esecuzione di azioni da parte del sistema operativo.

Il carico di lavoro globale di un sistema di elaborazione può quindi essere in generale considerato come somma delle seguenti componenti.

- **Componente periodica** costituita da un certo numero di attività di elaborazione basate sulla ripetizione di azioni con opportune, ed in generale diverse, frequenze.

Questa componente periodica produce un carico regolare, abbastanza ben quantificabile in sede di progetto, ed è caratteristica dei sistemi trasformativi, come quelli di acquisizione dati, monitoraggio e regolazione.

$$U = \sum_i \frac{C_i}{T_i}$$

$U$  = Utilization factor

$C_i$  = Computation time - tempo di elaborazione netto dell'azione i-esima

$T_i$  = Time period - periodo di ripetizione dell'azione i-esima.

Il carico di lavoro per la CPU dovuto alle azioni periodiche è

$$W_p = U$$

- **Componente aperiodica** così detta perchè le sue attività sono scatenate da eventi, esterni e/o interni, appunto di tipo sporadico e mutuamente aperiodico.

Questi eventi, e quindi il carico di lavoro da essi indotto, possono essere solo parzialmente caratterizzati in termini probabilistici, e nella realtà producono situazioni fortemente irregolari e di cui non è facile definire un "caso peggiore" (*worst case*) significativo e realistico.

Il carico di lavoro dovuto alle azioni attivate da eventi aperiodici è anche fortemente variabile nel tempo, e può essere caratterizzato con un valor medio  $W_{am}$  e con un valore massimo  $W_{aw}$  (caso peggiore).

$$W_{am} = \sum_i \frac{C_i}{T_{mi}} \quad T_{mi} \text{ è l'intervallo medio di attivazione dell'azione i-esima}$$

$$W_{aw} = \sum_i \frac{C_i}{T_{wi}} \quad T_{wi} \text{ è l'intervallo minimo (tempo morto) di attivazione dell'azione i-esima}$$

Questa componente è caratteristica dei sistemi detti "reattivi" perchè la loro attività consiste sostanzialmente nel reagire a stimoli esterni, come è il caso di automazione di macchine e processi discreti (a fasi).



- **Componente di "sottofondo"** (*background*) costituita da attività con scadenze molto lasche e tipicamente basate su un uso intensivo della CPU e piuttosto ridotto dell'I/O. Queste attività riguardano ad esempio analisi ed elaborazione "off-line" di dati raccolti, operazioni di configurazione da parte dell'operatore, attività di stampa periodica o su richiesta, ecc.

Il **carico di lavoro totale**  $W$  è dato dalla somma delle componenti:

$$W = W_p + W_a + W_b$$

Si noti che la componente  $W_b$  dovuta alle attività di *background* non è imposta, come le altre componenti  $W_p$  e  $W_a$ , ma costituisce un carico *assorbito* dal calcolatore rispettando la relazione

$$W_b < 1 - (W_p + W_a)$$

Il fattore di utilizzazione della CPU, che esprime la frazione del tempo per cui essa è attiva, non può ovviamente superare l'unità ed è pari al carico di lavoro finquando quest'ultimo si mantiene inferiore a 1.

Un carico di lavoro superiore all'unità corrisponde a situazioni di sovraccarico (*overload*) in cui il sistema non è in grado di rispettare le scadenze temporali delle azioni di risposta e di tenere il ritmo degli stimoli di ingresso.

Nelle applicazioni meno critiche possono essere accettati occasionali sovraccarichi per brevi periodi dopo i quali il sistema può riprendere il passo, mentre il caso in cui il sovraccarico sia permanente costituisce sempre un errore di dimensionamento, inaccettabile per sistemi real-time.

Se la stima del carico di lavoro per il caso peggiore delle componenti aperiodiche supera l'unità, l'applicazione dovrà essere organizzata per reagire in uno dei seguenti modi.

- Accettare il temporaneo e marginale mancato rispetto di alcune scadenze per le azioni meno critiche periodiche o aperiodiche.
- Per algoritmi multipli scegliere la variante più veloce e per gli algoritmi ad approssimazioni successive ridurre il numero di iterazioni, accettando la minor bontà dei valori calcolati.
- Abolire temporaneamente l'esecuzione di alcune attività periodiche meno critiche.

Le applicazioni industriali presentano diverse distribuzioni tra le componenti sopra citate. La situazione è poi complicata dal diverso "grado" di tempo reale richiesto per i singoli elementi di ogni componente (soprattutto periodica e aperiodica), rendendo così difficile individuare una metodologia univoca per il progetto ottimale dei sistemi informatici ed in particolare per la scelta delle risorse HW e SW di base che costituiranno la piattaforma implementativa.

## 2.9 ESERCIZI

-- Discutere se e in quali casi, sia preferibile avere misure con buona ripetibilità o con buona accuratezza. Pensare a possibili tecniche informatiche per correggere (parzialmente) gli errori nei due casi.

-- Discutere i motivi per cui per le grandezze tempo assoluto e spazio assoluto è preferibile una misura con errore assoluto costante, contrariamente alla grande maggioranza delle grandezze per cui sono preferibili misure con errore relativo costante.

-- Dire come varia l'errore di quantizzazione per ogni bit aggiuntivo di cui si possa disporre nella rappresentazione di un valore.

-- Gli errori di quantizzazione sono sempre per difetto, per eccesso o a valor medio nullo?. E in quali casi?.

-- Ricavare le espressioni che danno il campo di valori rappresentabili, e l'errore di quantizzazione relativo, con una rappresentazione a virgola mobile che disponga di  $N_e$  bit per l'esponente e  $N_m$  bit per la mantissa.

-- Discutere il fatto che pur essendo gli orologi (oscillatori al quarzo) tra gli strumenti più precisi generalmente disponibili, si deve ricorrere periodicamente ad un loro "aggiustaggio", contrariamente a quanto avviene per i normali strumenti.

-- Discutere gli errori che si commettono nella misura di intervalli temporali, nel caso normale di oscillatore indipendente dagli eventi iniziale e finale dell'intervallo stesso, e nel caso di oscillatore che possa essere attivato in corrispondenza dell'evento iniziale.

-- Quali tipi di stati caratterizzano i fenomeni fisici e quali i fenomeni informatici?

-- Discutere la relazione tra tempo morto del segnale e tempo morto del fenomeno, che consente un riconoscimento degli spurii del segnale.

-- Alcuni orologi digitali da polso possono essere impostati per emettere un "bip" ad ogni ora e/o per emettere un suono (alarm) ad un'ora prefissata. Dire quali di questi eventi sono assoluti e quali incrementali, motivando.

-- Le funzioni di Validità temporale di una risposta sono quasi sempre di tipo monotono, o almeno riconducibili a tale caso. Dire se l'affermazione è vera e motivare.

-- Discutere l'affermazione che il carico di lavoro per la CPU delle attività di background non è imposto da esse, ma "assorbito".

## 2.10 BIBLIOGRAFIA

### LIBRI

[LEAG90] S.T. Levi, A.K. Agrawala  
Real-Time System Design.  
McGraw-Hill Pub.Co. 1990

[BUW90] A. Burns, A. Wellings  
Real-time systems and their programming languages  
Addison-Wesley Pub. Co. 1990  
*Cap.12 Orologi in OCCAM, ADA, Modula.*

### ARTICOLI

[MRCM91] A. Montanari, E. Ratto, E. Corsetti, A. Morzenti Embedding Time Granularity in Logical Specifications of Real-time Systems.  
Proc. Euromicro-91 Workshop on real-time systems.  
IEEE Press.

[JAM86] F. Jahanian, A.K. Mok  
Safety Analysis of Timing Properties in Real-Time Systems.  
IEEE Trans. on Software Eng. Vol. SE-12 N.9 Sett.1986  
*Un interessante articolo sulla caratterizzazione di specifiche e proprietà di sistemi in tempo reale con stati, eventi, logica temporale (RTL = Real-Time Logic):*

[SCH90] F.A. Schreiber  
A breviary to time concepts for information systems.  
Rivista di INFORMATICA 1/90

<b>2. CARATTERIZZAZIONI.....</b>	<b>2-1</b>
2.1 CARATTERIZZAZIONE DEGLI ERRORI .....	2-2
2.1.1 ERRORI FORMALI.....	2-2
2.1.2 ERRORI DI COMPORTAMENTO.....	2-2
2.1.2.1 ERRORI DI VALORE .....	2-2
2.1.2.2 ERRORI TEMPORALI.....	2-3
2.1.3 ERRORI DI MISURA.....	2-4
2.1.3.1 Distribuzione degli errori con misure ripetute .....	2-4
2.1.3.2 Distribuzione degli errori per diversi valori V .....	2-6
2.1.4 ERRORI DI QUANTIZZAZIONE.....	2-7
2.1.4.1 MISURE RAPPRESENTATE CON INTERI .....	2-7
2.1.4.2 MISURE RAPPRESENTATE CON VIRGOLA MOBILE .....	2-10
2.2 CARATTERIZZAZIONE DEL TEMPO.....	2-11
2.2.1 IL MODELLO DEL TEMPO .....	2-11
2.2.1.1 ERRORI.....	2-13
2.3 CARATTERIZZAZIONE DEL COMPORTAMENTO .....	2-17
2.3.1 CONCETTI DI STATO E DI EVENTO .....	2-17
2.3.1.1 STATI CONTINUI A TEMPO CONTINUO .....	2-18
2.3.1.2 STATI CONTINUI A TEMPO DISCRETO .....	2-19
2.3.1.3 STATI DISCRETI A TEMPO CONTINUO .....	2-20
2.3.1.4 STATI DISCRETI A TEMPO DISCRETO.....	2-21
2.3.2 FLUSSI DI EVENTI.....	2-21
2.3.2.1 SORGENTI DI EVENTI.....	2-22
2.3.2.2 CONTENUTO INFORMATIVO DEGLI EVENTI .....	2-22
2.3.2.3 PROPRIETA' TEMPORALI DEGLI EVENTI .....	2-26
2.3.2.4 CORRELAZIONI TRA DIVERSI FLUSSI DI EVENTI.....	2-28
2.4 AZIONI ED ATTIVITA'.....	2-29
2.4.1 AZIONI - COMPORTAMENTO REATTIVO .....	2-29
2.4.2 ATTIVITA' - COMPORTAMENTO TRASFORMAZIONALE .....	2-30
2.5 CARATTERIZZAZIONE DEL MONDO ESTERNO.....	2-30
2.5.1 MODI FASI E PASSI DEI SISTEMI .....	2-31
2.5.2 INFORMAZIONI DI INTERFACCIA.....	2-31
2.5.2.1 INFORMAZIONI DESTINATE AL CALCOLATORE .....	2-32
2.5.2.2 COMANDI EMESSI DAL CALCOLATORE.....	2-32
2.5.2.3 SEGNALI FISICI .....	2-32
2.5.3 SPECIFICHE DI PRECISIONE .....	2-32
2.6 CARATTERIZZAZIONE DELLE PRESTAZIONI TEMPORALI RICHIESTE .....	2-34
2.7 CARATTERIZZAZIONE DELLE PRESTAZIONI TEMPORALI OFFERTE DALLE AZIONI...	2-37
2.7.1 ALGORITMI NORMALI SINGOLI .....	2-37
2.7.2 ALGORITMI NORMALI MULTIPLI .....	2-38
2.7.3 ALGORITMI AD APPROSSIMAZIONI SUCCESSIVE .....	2-38
2.8 CARICO DI LAVORO DELL'ELABORATORE .....	2-40
2.9 ESERCIZI .....	2-42
2.10 BIBLIOGRAFIA .....	2-43