

G. Casadei A.G.B. Teolis

# LA GESTIONE DEI PROGETTI INFORMATICI

## Prima parte: i progetti

### 1. La definizione

#### 1.0 Introduzione

#### 1.1 WBS, schema di progetto, piano di progetto e Gantt

#### 1.2 Fasi di un progetto

#### 1.3 Risorse costi vincoli e percorso critico

#### 1.4 Verifica e controllo

#### 1.5 CPM e Pert

#### 1.6 La pianificazione come compromesso fra tempo e risorse

REVISIONE DEL 10/11/2003

## Prima parte: i progetti

In questa prima parte viene descritta l'anatomia dei progetti, cioè vengono illustrati, nei singoli capitoli, i metodi per

- descrivere un progetto,
- valutarne la fattibilità (anche dal punto di vista economico) e gestirlo temporalmente,
- valutare quantitativamente gli elementi significativi,
- organizzarne le attività in maniera da raggiungere gli obiettivi fissati.

La capacità di costruire programmi si basa su due competenze: l'una, acquisita *bottom up*, consiste nella conoscenza di linguaggi di programmazione e l'altra, acquisita *top down*, consiste nel saper scomporre problemi in sottoproblemi. Analogamente, la capacità di gestire progetti (*software*) si basa su due competenze: una *bottom up* che parte dalle attività, ne valuta tempi e costi e le aggrega in un progetto; l'altra *top down* parte dagli obiettivi del progetto, li scompone in sotto obiettivi formalizzando le fasi per raggiungerli sino a individuare le singole attività (con i relativi tempi di svolgimento, risorse da impiegare e costi).

Questa prima parte del libro si articola in quattro capitoli: la definizione, la gestione, le metriche e i modelli.

Nel primo vengono illustrati i concetti, gli strumenti e i metodi necessari per costruire la documentazione relativa ai progetti e per la loro conduzione ottimale..

Nel secondo vengono espone le azioni rilevanti per la gestione dei progetti, gli attori coinvolti (da parte del committente e da quella del fornitore) e i rispettivi ruoli.

Nel terzo sono introdotte le tecniche quantitative per valutare analiticamente i parametri fondamentali in gioco (impegno, tempi, costi, etc).

Nel quarto, infine, sono illustrati i principali modelli utilizzati per articolare un progetto in fasi operative.

# 1. La definizione

In questo Capitolo viene data la definizione di progetto e vengono illustrati i metodi per costruire la documentazione relativa ai progetti e le operazioni formali connesse. Il primo argomento è il concetto di attività e gli attributi che consentono di organizzarle in un Gantt. Il secondo argomento consiste nel descrivere un progetto come un insieme strutturato di attività, mediante la scomposizione del progetto in fasi. Il terzo argomento illustra come attribuire risorse e costi alle singole attività e come individuare il percorso critico. Il quarto argomento è costituito dalla verifica e dal controllo dell'avanzamento di un progetto e dai metodi per la ripianificazione. Il quinto argomento riguarda il Pert, un metodo alternativo al Gantt, per descrivere le attività nel tempo.

## 1.0 Introduzione

Nell'ambito degli argomenti trattati in questo testo, un insieme di attività si chiama progetto quando sono soddisfatte tutte le seguenti caratteristiche

1. è necessaria una **pianificazione**;
2. esiste un **obiettivo specifico** e raggiungibile;
3. si può individuare una **durata predeterminata** (almeno in linea di principio);
4. esistono **risorse disponibili** (con vincoli predefiniti);
5. l'insieme delle attività è sufficientemente **complesso**;
6. le attività **non** siano la **ripetizione** (quasi esatta) di esperienze già fatte.

Per esempio, il così detto "Progetto Apollo" (il piano varato dal governo americano negli anni sessanta per portare un uomo sulla luna) non è in realtà un progetto in questa accezione in quanto mancano le condizioni 2, 3 e 4; infatti:

- non aveva un obiettivo specifico e raggiungibile al momento del suo varo (il piano aveva come scopo quello di rendere raggiungibile un obiettivo);
- non era possibile predeterminare la durata (esisteva solo l'esigenza di carattere generale di raggiungere l'obiettivo entro il decennio);
- non era possibile valutare le risorse in maniera analitica.

Altro esempio che non soddisfa la definizione data in questo contesto è il così detto "Progetto Manhattan" che aveva come obiettivo la costruzione della prima bomba atomica.

D'altra parte, neppure l'insieme delle attività svolte da una catena di montaggio è un progetto (nonostante soddisfi le prime cinque condizioni) perché consiste in una **ripetizione** seriale di azioni per raggiungere ripetutamente uno stesso obiettivo specifico.

Questi esempi mostrano che il concetto di progetto, così come definito, richiede che siano soddisfatti dei vincoli abbastanza stringenti e non si adatta né alle imprese troppo grandi né a quelle troppo semplici.

Si sottolinea comunque che non esiste progetto senza limiti temporali predefiniti e senza la disponibilità delle risorse finanziarie e tecnico-scientifiche necessarie. In particolare, non possono essere chiamati progetti le iniziative che prevedono attività per la messa a punto di tecniche non ancora esistenti (o puramente sperimentali) o presuppongono di utilizzare uno strumento non ancora disponibile sul mercato.

Inoltre non costituiscono un progetto le attività di una sola persona, oppure le attività di più persone che non interagiscono, oppure ancora fare la decimilionesima bottiglia, mentre costituisce un progetto fare la decima centrale elettrica o la quinta enciclopedia multimediale perché nessuna di queste può essere la esatta ripetizione di una delle precedenti.

In sintesi, supponendo che l'insieme delle attività sia sufficientemente complesso e non costituisca la ripetizione esatta di cose già fatte e che esistano le risorse tecnologiche, economiche e di conoscenza, un progetto è caratterizzato da:

1. **obiettivo,**
2. **tempi e**
3. **costi**

che devono essere integrati nella **pianificazione**.

Nel resto del capitolo vengono esposti essenzialmente i metodi formali per svolgere la pianificazione (e la ripianificazione) di un progetto, cioè per redigere il piano (e mantenerlo aggiornato in corso d'opera).

### **1.1 WBS, schema di progetto, piano di progetto e Gantt**

Sia  $P$  un insieme finito; una *relazione binaria* su  $P$  è un sottoinsieme  $E \subset P \times P$ ; per indicare che due elementi  $x, y \in P$  stanno nelle relazione  $E$  si scrive sia  $E(x, y)$  (notazione *prefissa*) sia  $xEy$  (notazione *infissa*).

Una relazione di *preordine* su  $P$  è una relazione binaria  $E$ , che verrà indicata con  $\preceq$  in forma infissa, che soddisfa le due proprietà:

*riflessività*  $\forall x \in P: (x, x) \in E$  ovvero  $x \preceq x$ ,

*transitività*  $\forall x, y, z \in P: (x, y) \in E \text{ e } (y, z) \in E \Rightarrow (x, z) \in E$  ovvero  $x \preceq y, y \preceq z \Rightarrow x \preceq z$ .

Un preordine si dice *ordine (parziale)* se gode della seguente proprietà

*antisimmetria*  $\forall x, y \in P: (x, y) \in E \text{ e } (y, x) \in E \Rightarrow x = y$  ovvero  $x \preceq y, y \preceq x \Rightarrow x = y$ .

Data una relazione qualunque  $E \subset P \times P$ , si dice *chiusura riflessiva* il più piccolo sottinsieme  $E' \subset P \times P$  che contiene  $E$  e gode della proprietà riflessiva; è facile vedere che  $E' = E \cup \Delta$ , dove  $\Delta$  è l'insieme delle coppie  $(x, x)$ ,  $\forall x \in P$ .

Data una relazione qualunque  $E \subset P \times P$ , si dice *chiusura transitiva* il più piccolo sottinsieme  $E'' \subset P \times P$  che contiene  $E$  e gode della proprietà transitiva.

#### **Esercizi**

E1. Dato un insieme finito  $P$ , scrivere l'algoritmo che per ogni relazione  $E$  su  $P$  costruisce la chiusura transitiva; mostrarne la terminazione.

Le relazioni su insiemi finiti spesso si raffigurano con grafi orientati: i nodi del grafo sono gli elementi dell'insieme e le frecce rappresentano una coppia (ordinata) della relazione.

Per esempio, la figura 1.1.a mostra una relazione la cui chiusura riflessiva e transitiva è un preordine che non è un ordine (parziale) perché contiene un ciclo; la figura 1.1.b mostra una relazione la cui chiusura riflessiva e transitiva è un ordine (parziale).

Sia  $P$  un insieme ordinato (mediante  $E$ ); un sottinsieme  $P' \subset P$  eredita una struttura d'ordine mediante  $E' = E \cap P' \times P'$ .

Sia  $P$  un insieme ordinato;  $P' \subset P$  si dice una *catena* se per ogni  $x, y \in P'$  si ha o  $x \preceq y$  oppure  $y \preceq x$ .

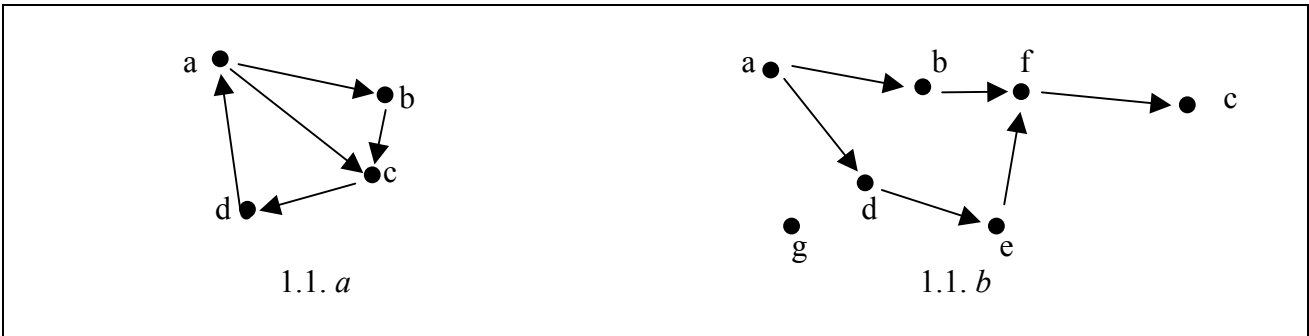


Figura 1.1

Sia  $P$  un insieme ordinato e sia data una funzione  $P \rightarrow N$  (insieme dei numeri naturali) che sarà detta *durata*; gli elementi di  $P$  preimmagini di 0 si dicono *eventi* e gli altri si dicono *attività*.

Una coppia  $(P, \preceq)$  si dice *WBS (Work Breakdown Structure)*; questa struttura rappresenta un elemento fondamentale nella descrizione di un progetto e cattura l'idea dell'elenco delle "cose" importanti nel contesto considerato (gli eventi o le attività, cioè gli elementi dell'insieme  $P$ ); è caratterizzata dall'ordine  $\preceq$ , che cattura l'idea della precedenza tra attività ed eventi.

Una terna  $(P, \preceq, d)$  si dice *schema (di progetto)* se in  $P$  esiste un primo e un ultimo elemento, tipicamente indicati con  $\alpha$  e  $\omega$ . Un qualunque WBS può sempre essere completato aggiungendo  $\alpha$  e  $\omega$  ( $\notin P$ ) ed estendendo opportunamente l'ordine (ponendo per ogni  $x \in P$ ,  $\alpha \preceq x$  e  $x \preceq \omega$ ) e la funzione  $d$  (ponendo  $d(\alpha)=d(\omega)=0$ ).

Dato uno schema  $(P, \preceq, d)$ , una funzione  $t: P \rightarrow Z$  (insieme dei numeri interi) si dice *assegnazione temporale* se

$$\forall x, y \in P, x \preceq y \Rightarrow t(x) + d(x) \leq t(y).$$

La funzione  $t$  definisce il tempo in cui inizia una attività o si verifica un evento ed esprime l'idea intuitiva che una certa attività o un evento successivi ad una attività possono accadere solo quando quest'ultima è terminata. Si noti come due elementi  $x$  ed  $y$  non confrontabili nell'ordine (cioè né  $(x, y)$ , né  $(y, x)$  appartengono alla relazione) possono essere "contemporanei": cioè non esiste nessun vincolo tra  $t(x)$  e  $t(y)$ .

Uno schema con una assegnazione temporale si dice *piano di progetto*.

Esiste un metodo grafico particolarmente efficace per descrivere gli schemi che abbiano una assegnazione temporale: i *diagrammi di Gantt*.

Esempio. Dato  $P=\{a, b, c, d, e\}$ , siano date le funzioni  $d$  e  $t$  definite dalle seguenti tabelle

	$d$
$a$	0
$b$	3
$c$	3
$d$	3
$e$	0

	$t$
$a$	3
$b$	4
$c$	5
$d$	8
$e$	12

e sia l'ordine su  $P$  quello mostrato in figura 1.2.

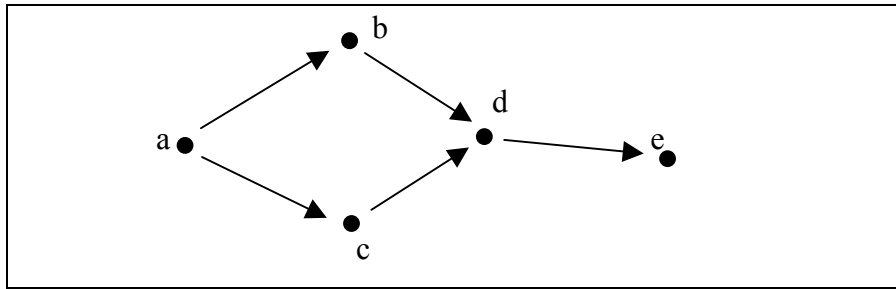


Figura 1.2

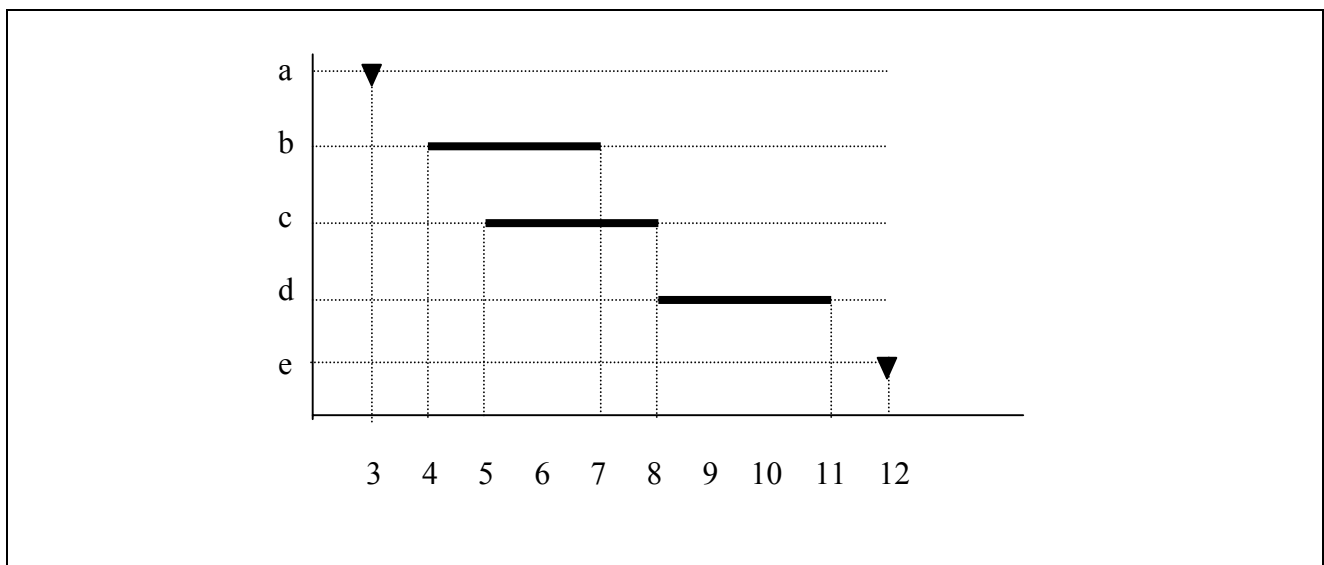


Figura 1.3

Oltre alla rappresentazione della pura relazione d'ordine di figura 1.2, è particolarmente significativo il diagramma di Gantt (riportato in figura 1.3) che descrive anche la particolare funzione  $t$  scelta. Come si vede, un diagramma di Gantt ha sulle ordinate tante "righe" quanti sono gli elementi del WBS (eventi e attività); sulle ascisse viene riportato il *range* della funzione  $t$  (l'inizio di una attività o l'istante in cui accade un evento); convenzionalmente l'unità di tempo viene scelta in modo tale da avere a che fare solo con numeri interi (giorni settimane o mesi). Di norma, gli eventi sono contrassegnati da un simbolo "puntiforme" (per esempio un triangolino) e le attività da un segmento (la cui lunghezza ne indica la durata).

In uno schema, la funzione  $d$  è essenzialmente intrinseca (cioè legata alle caratteristiche delle attività da svolgere e alle risorse disponibili), mentre assegnare la funzione  $t$  è in larga misura discrezionale (perché l'inizio delle attività può anche essere posposto "a piacere").

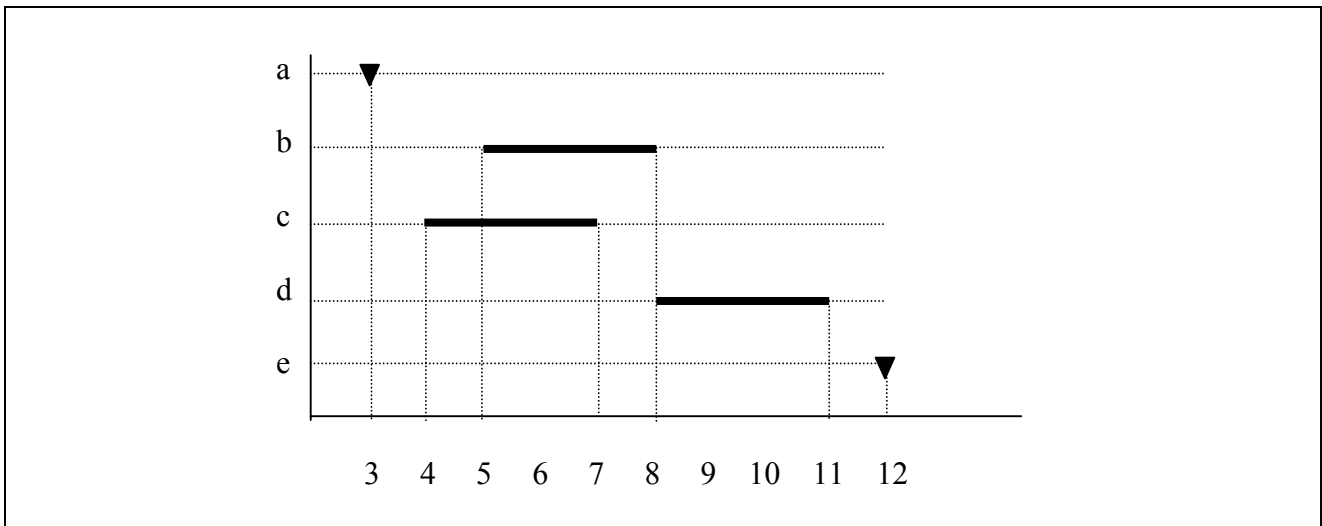


Figura 1.4

Come esempio si possono considerare i diagrammi di Gantt di figura 1.4 e figura 1.5 che utilizzano l'ordine di figura 1.2 e la durata della tabella precedente fanno riferimento rispettivamente alle due seguenti tabelle per  $t$ .

	$t$
$a$	3
$b$	5
$c$	4
$d$	8
$e$	12

	$t$
$a$	3
$b$	4
$c$	4
$d$	9
$e$	12

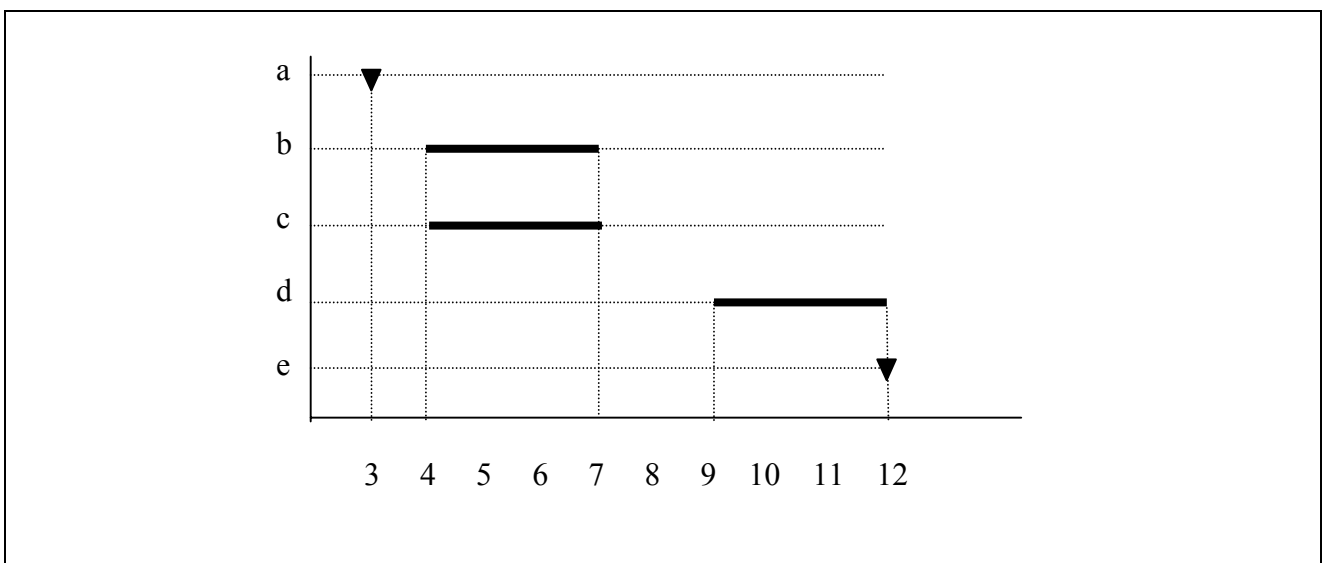


Figura 1.5

La tabella seguente mostra una funzione  $t$  che non è una assegnazione temporale per l'ordine di figura 1.2..

	$t$
$a$	5
$b$	3
$c$	4
$d$	10
$e$	11

Infatti, per esempio, i tempi di inizio di  $b$  e di  $c$  non sono compatibili con l'ordine rispetto ad  $a$  e il tempo di inizio di  $e$  non è compatibile con la durata di  $d$ .

Dato un piano di progetto, si dice “durata totale del progetto” la quantità  $t(\omega) + d(\omega) - t(\alpha)$ .

### Esercizi

E2. Dato  $P = \{a, b, c, d, e, f, g\}$ ; l'ordine è la chiusura riflessiva e transitiva di  $\{(a, b), (a, g), (b, c), (g, f), (g, e), (f, d), (e, d), (c, d)\} \subset P^2$ , le funzioni  $d$  e  $t$  sono date dalla seguente tabella:

	$d$	$t$
$a$	3	3
$b$	5	6
$c$	7	12
$d$	2	25
$e$	8	15
$f$	4	12
$g$	6	6

- verificare che i valori assegnati per la funzione  $t$  siano una assegnazione temporale;
- in caso positivo, disegnare il diagramma di Gantt; altrimenti correggerla in modo che lo sia e disegnare il diagramma di Gantt.

E3. Con riferimento ai dati dell'esercizio precedente come esempio, costruire un programma per rispondere alle domande a) e b). Particolare cura deve essere posta nel determinare la struttura per rappresentare la conoscenza sul problema, in modo da ottenere una soluzione efficiente.

E4. Con riferimento agli esercizi precedenti scrivere un programma che, partendo da uno schema di progetto, determini la funzione  $t$  che minimizza la durata complessiva del progetto.

E5. Con riferimento agli esercizi precedenti scrivere un programma che, partendo da uno schema di progetto, determini una assegnazione temporale  $t$  che implica lo svolgimento di una sola attività alla volta.

### Il tempo

L'asse delle ascisse di un diagramma di Gantt misura il tempo. Nella schematizzazione vista (per la definizione della funzione  $t$ ) il tempo è essenzialmente riducibile all'insieme degli interi (con la sua struttura d'ordine); questo presuppone che è possibile fissare un intervallo di tempo che sia “elementare” nel contesto in considerazione, cioè rappresenti il (massimo) potere risolutivo con cui si vogliono osservare i fenomeni (per quello che riguarda sia il tempo di inizio delle attività, sia la loro durata); esempi sono: l'ora, il giorno, la settimana o il mese.

Particolare attenzione va posta quando le unità temporali di lavoro effettivo devono essere sovrapposte al calendario reale a causa della presenza in quest'ultimo delle giornate festive (e prefestive) in uso nei paesi coinvolti nel progetto.

Usualmente i prodotti commerciali che consentono di preparare, editare e mantenere i Gantt (e la documentazione di progetto) si prendono cura di questo problema.

### Il Tracking

Il diagramma di Gantt, che presuppone il WBS e la valutazione delle funzioni  $d$  e  $t$ , è uno dei documenti fondamentali della pianificazione di un progetto. Oltre che nella fase preliminare di pianificazione, ha anche un ruolo determinante nella gestione del progetto, in particolare in quella attività che si chiama *tracking*. Il Gantt mostra le “cose” da fare e indica quando devono essere fatte; il *tracking* è la registrazione delle “cose” che sono state effettivamente fatte riportando le date reali.

Mentre il Gantt è una descrizione a priori, il *tracking* è una registrazione a posteriori (si veda la figura 1.6) e, riportata sullo stesso grafico del Gantt, permette di valutare anche visivamente gli scostamenti fra previsione e attuazione e quindi costituisce uno strumento essenziale per mantenere il controllo del progetto.

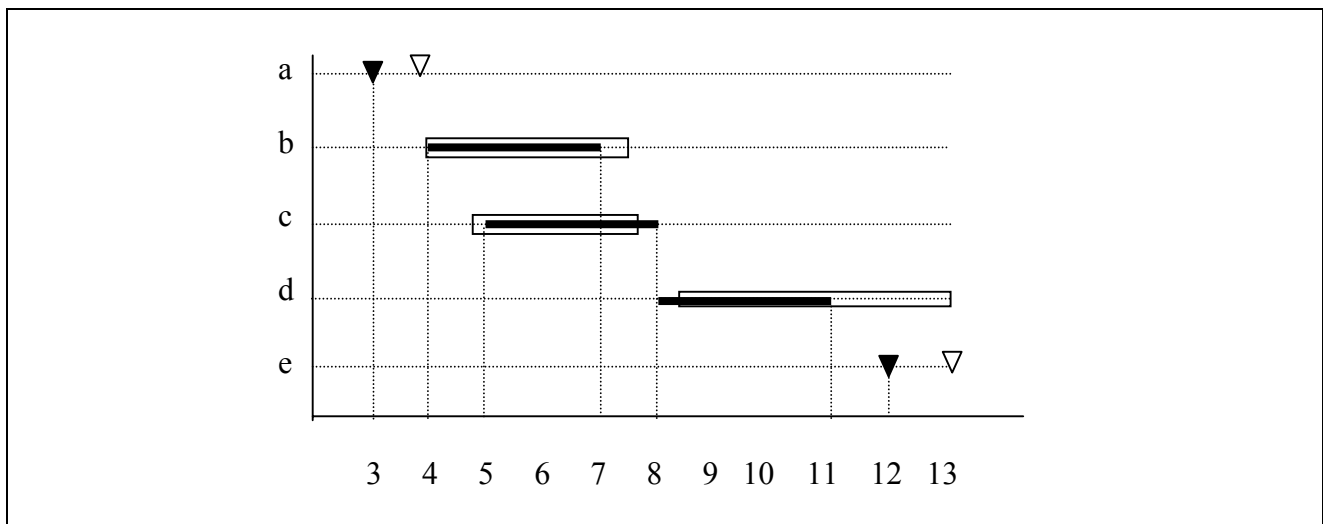


Figura 1.6

### PROGETTO

La parola progetto viene spesso usata in due accezioni diverse.

Nella prima, progetto indica un insieme di attività ordinate, dotate di durata e costo e sistemate opportunamente nel tempo; è caratterizzato anche da una durata e un costo totali. Un esempio d'uso di questa accezione è per esempio la frase “il progetto è in ritardo” che indica che la durata reale è superiore a quella pianificata.

Nella seconda accezione la parola progetto indica un documento che descrive l'insieme delle attività ordinate (v. accezione precedente) in vario modo (per esempio solo con un Gantt oppure con la documentazione di progetto che verrà descritta successivamente). Un esempio d'uso è la frase seguente: il progetto deve essere preparato entro la settimana prossima. Gli ingegneri usano spesso questa accezione: il “progetto di una casa” indica quasi sempre un insieme di disegni e documentazione tecnica.

Esistono anche delle frasi ancipiti come per esempio “il progetto è troppo costoso” che può riferirsi sia al documento che alla realizzazione.

**In linea di principio, nel presente contesto la parola “progetto” assume (quasi sempre solo) il primo significato.**

## 1.2 Fasi di un progetto

Dato uno schema  $(P, \alpha, d)$ , sia  $F \subseteq P$  con

$$\exists x, y \in F: \forall z \in F: x \alpha z \text{ e } z \alpha y.$$

La terna  $(F, \alpha', d')$ , con il  $'$  che indica la restrizione a  $F$ , si dice *sottoschema*. Gli elementi  $x$  e  $y$  si dicono *inizio* e *fine* del sottoschema; in maniera informale un sottoschema  $F$  è un insieme di attività ed eventi che ha un “primo” e un “ultimo” elemento, quindi può essere “collassato” (o contratto) in un solo punto; ci possono essere altri elementi di  $P$  non confrontabili con gli elementi di  $F$ .

Se per  $F \subseteq P$  oltre alla condizione precedente vale anche che

$$\forall z \in P-F: \forall z \in F: y \alpha z \text{ oppure } z \alpha x,$$

cioè tutte le altre attività o eventi sono precedenti o successivi, allora la terna  $(F, \alpha', d')$  si dice *fase (di progetto)*.

Può accadere che l'insieme  $P$  possa essere partizionato in fasi di progetto (disgiunte)  $F_1, F_2 \dots$  che costituiscono una catena.

### **Esercizi**

E6. Definire l'ordine rispetto cui le fasi di progetto costituiscono una catena.

In ogni area disciplinare (ingegneria civile, informatica, ingegneria meccanica) si sono evoluti metodi diversi per condurre con successo i vari progetti. In ciascuna di queste aree sono state poi individuate metodologie diverse per specifiche classi. Le varie metodologie per condurre con successo i progetti nell'area informatica sono descritte nel quarto capitolo. Queste metodologie differiscono (intrinsecamente) non solo per gli obiettivi e i risultati che vogliono raggiungere, ma (formalmente) per le varie *fasi* in cui possono essere partizionati i diversi progetti. Ad esempio in un progetto di edilizia possono essere determinanti le fasi di prospezione geologica, scavo per le fondamenta o approvvigionamento dei materiali da costruzione; in progetti di ingegneria meccanica invece sono determinanti le fasi di indagine di mercato, localizzazione degli stabilimenti per la produzione e individuazione delle catene di fornitura.

Come si è visto, per descrivere uno schema di progetto è necessario elencare tutti gli eventi e le attività coinvolti, con la indicazione delle rispettive relazioni di precedenza e le durate. Queste attività sono spesso centinaia e per individuarle viene sfruttata la possibilità di ripartire il progetto in fasi, in modo che ogni fase del progetto abbia associato un insieme caratteristico di attività che devono essere prese in considerazione (o per escluderle o per ripeterle opportunamente al fine di adeguarle alle esigenze del progetto specifico).

Per fissare le idee e per illustrare degli esempi, si fa riferimento ad uno dei modelli di progetto informatico, più semplice e di uso più frequente tra quelli che sono illustrati nel seguito (Capitolo 4), che si chiama a “cascata” o *waterfall* (Vedi figura 1.7).

Poiché attualmente gli scopi sono solo illustrativi, le fasi non sono esattamente quelle classiche, che verranno esaminate successivamente.

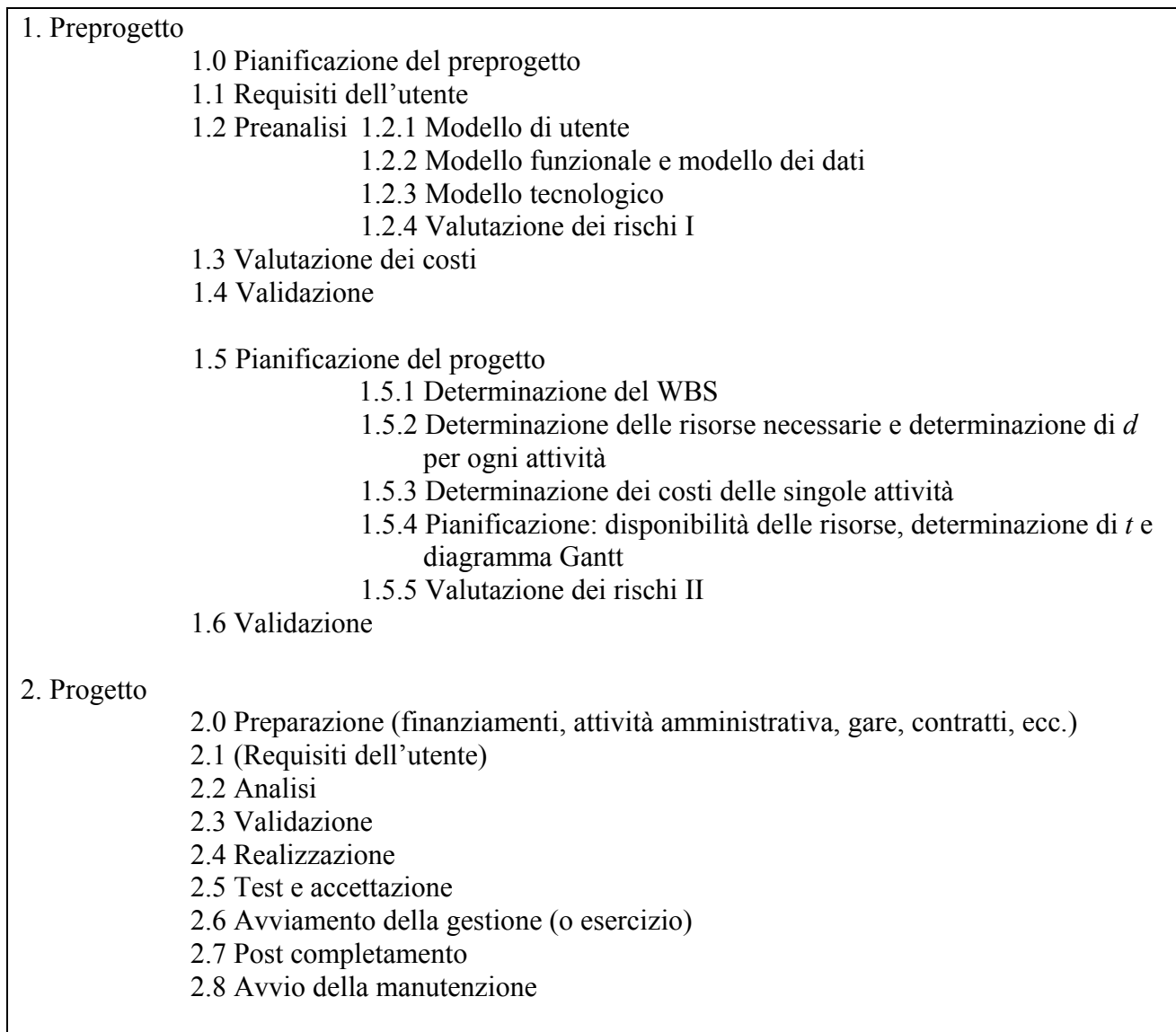


Figura 1.7

La (macro)fase di preprogetto è abbastanza indipendente dal modello di progetto adottato; è sostanzialmente divisa in due parti: nella prima (da 1.1 a 1.4) occorre fare una preanalisi, cioè una descrizione, a livello alto, delle caratteristiche generali della realizzazione; questa parte, che in altri contesti è chiamata studio di fattibilità (vedi capitolo 2), deve raccogliere in maniera organica i requisiti degli utenti e conseguentemente trattare almeno i seguenti punti:

1. gli utenti (che spesso sono anche migliaia) classificati per tipologie ed esigenze logistiche,
2. le funzioni (come percepite dalle varie tipologie utenti) e i dati, come è usuale in questo contesto,
3. le apparecchiature presumibilmente necessarie per l'esercizio dell'applicazione,
4. i punti critici e i rischi prevedibili.

Questa prima parte termina con la produzione di un documento formale che contiene le quattro parti (relative ai punti precedenti) e una prima valutazione (anche approssimata) dei costi: il documento deve essere approvato esplicitamente (validazione).

La seconda parte di questa prima fase, è sostanzialmente **la pianificazione del progetto** che consiste in:

- 1.5.1 fare l'elenco delle attività del progetto con la struttura delle precedenze (cioè definire il WBS),

- 1.5.2 valutare, per ogni attività, l'impegno necessario per portarla a termine e quindi la durata,
- 1.5.3 stimare le risorse disponibili e definire, per ogni attività, il costo,
- 1.5.4 in base alla disponibilità globale delle risorse e dei vincoli (per quanto può essere noto), definire una funzione  $t$  (provvisoria) e quindi un primo Gantt,
- 1.5.5 valutare i punti critici e rideterminare i rischi con riferimento anche allo sviluppo temporale e al costo (analitico) del progetto.

Anche questa seconda parte deve terminare con una approvazione esplicita dei documenti prodotti (validazione).

Nel punto 1.5.1 si deve utilizzare la scomposizione del progetto in fasi; ogni modello di progetto ha una propria partizione in fasi e ciascuna di queste prevede un elenco di possibili attività ed eventi che sono caratteristici della fase; un modello di progetto (come si vedrà nei capitoli 2 e 4) si caratterizza con un *WBS tipo*, che è una sorta di *menu* da cui si scelgono gli elementi da includere (alcuni opportunamente ripetuti) nel WBS del progetto.

Il progetto vero e proprio inizia con una fase che raggruppa tutte le attività (di carattere amministrativo e finanziario) che sono necessarie per rendere disponibili le risorse (ricerca di finanziamenti, espletamento di gare, stipula di contratti, ecc.). Quindi segue una serie di fasi che richiamano il ciclo di vita del *software*. La raccolta dei *requisiti dell'utente* (che deve essere rifatta se nel preprogetto non è stata portata a termine in maniera dettagliata) e l'*analisi* sono quelle "classiche" e terminano con un documento che, per le realizzazioni di cui si parla in questo contesto, è di dimensioni notevoli (anche parecchie centinaia di pagine). La *validazione* è un punto critico perché richiede la comprensione, da parte di chi la effettua, di tutti i documenti dell'analisi: per questo sono necessarie parecchie competenze diverse<sup>1</sup>.

La *realizzazione* (di quanto individuato dall'analisi) viene condotta con la metodologia classica che si studia in ingegneria del *software* e presenta due difficoltà principali: il coordinamento di parecchi sviluppatori che lavorano in parallelo (talvolta parecchie decine) e il vincolo di terminare entro un tempo prestabilito. Naturalmente, questa fase comprende, oltre la programmazione, anche il *system design*, il *program design* e il *debugging (unit test)*.

La fase di *test e accettazione* può essere anche molto complessa perché comporta il controllo della funzionalità del prodotto (che può coinvolgere anche decine di tipi di utenti diversi) e il controllo delle prestazioni; questi controlli sono difficili da eseguire perché presuppongono la ricostruzione esatta dell'ambiente in cui il sistema sarà utilizzato (che talvolta prevede migliaia di utenti contemporanei).

L'*avviamento in gestione* è una fase delicata e cruciale per il successo del progetto; la sua natura, più che informatica o tecnica, è di carattere logistico e organizzativo; comporta, per esempio, l'addestramento di parecchie centinaia di utenti con durate che vanno da una a parecchie giornate, l'installazione di nuove stazioni di lavoro o l'*upgrade* delle vecchie, l'installazione del *software* prodotto (spesso in più *server* geograficamente distribuiti) e infine l'assistenza agli utenti per lo *startup* (che può avvenire in varie strategie che vanno dall'avvio contemporaneo all'ora zero per tutte le funzionalità e per tutti gli utenti coinvolti alla partenza incrementale per gruppi di utenti distinti geograficamente o per funzionalità).

---

<sup>1</sup> La difficoltà di realizzare le condizioni per effettuare una affidabile validazione dell'analisi giustifica la criticità di questa fase e ha favorito la nascita di strategie alternative per la strutturazione in fasi e quindi per i modelli (vedi capitolo 4).

Il *postcompletamento* gioca un ruolo essenziale nell'arricchimento culturale sia del fornitore sia del committente. Questa fase comporta la valutazione critica dei parametri del progetto (ritardi, costi aggiuntivi ed errori di previsione in genere), il confronto fra la valutazione dei rischi e ciò che si è effettivamente verificato e la valutazione dei parametri di successo; questi possono essere la soddisfazione del cliente (gli utenti e il committente) e del fornitore (margini e impatto sul successo).  
 Successivamente occorre realizzare le azioni che consentono l'avvio della manutenzione del sistema prodotto.

### **1.3 Risorse costi vincoli e percorso critico**

#### **Risorse**

Una volta determinato l'insieme  $P$ , per un progetto, occorre determinare la durata di ogni attività. Per fare ciò è essenziale il concetto di *risorsa*: in linea di principio, la durata della maggior parte delle attività dipende dai mezzi (strumentali, finanziari, umani, ecc.) che possono essere impiegati per il loro svolgimento. Esistono essenzialmente due classi di risorse: quelle cosiddette *a consumo* e quelle dette *individuali* o *stabili*.

Le risorse del primo tipo (come per esempio carburanti, carta, energia elettrica) sono caratterizzate dalla ennupla

$$(\langle \text{nome} \rangle, \langle \text{tipo} \rangle, \langle \text{unità di misura} \rangle, \langle \text{quantità} \rangle, \langle \text{tariffa} \rangle, \dots)$$

quelle del secondo tipo (come per esempio le persone, le attrezzature, i siti) dalla ennupla

$$(\langle \text{nome} \rangle, \langle \text{tipo} \rangle, \langle \text{quantità} \rangle, \langle \text{tariffa} \rangle, \langle \text{intervalli di disponibilità} \rangle, \dots).$$

Il nome è un identificativo univoco della risorsa (per esempio il nome di una persona o il numero di inventario di una apparecchiatura; il tipo è la classe della risorsa come per esempio programmatore, capo progetto oppure *router* o *server*).

In questo contesto vengono prese in considerazione solo risorse individuali (che sono le più importanti per i progetti *software*).

È necessario ed essenziale tener conto delle risorse quando si definisce lo schema e poi il piano di progetto (e si traccia il Gantt) sia per valutare la funzione  $d$  sia, successivamente, per valutare la funzione  $t$ .

Ad ogni attività è associata (con metodi che saranno visti successivamente) una misura (che ne esprime in qualche modo la "quantità") chiamata *impegno* dell'attività; l'unità di misura dell'impegno è  $\langle \text{durata} \rangle \times \langle \text{tipo} \rangle$  come per esempio "giorno sviluppatore", "mese sistemista" o "anno operatore".

La durata di una singola attività, cioè la funzione  $d$ , dipende (almeno in prima approssimazione) dall'impegno associato a quella attività e dalla quantità di risorse ad essa specificamente assegnate, cioè

$$d(x) = \text{impegno}(x) / \text{quantità di risorse}(x),$$

come per esempio

$$d = 15 \text{ giorni sviluppatore} / 3 \text{ sviluppatori} = 5 \text{ giorni.}$$

Calcolare la funzione  $d$  significa, in maniera implicita o esplicita, attribuire le risorse alle singole attività (che nascono in modo "naturale" misurate in impegno). Inoltre, per semplicità, le risorse si ritengono fisse e disponibili per quella attività; in caso contrario si spezza l'attività in parti in ciascuna delle quali si può assegnare una quantità costante di risorse.

## Costi

Ad ogni attività sono associati dei costi che sono di due tipi:

- i *costi variabili* generati dalle risorse attribuite alla attività e dipendono dall'impegno associato alla attività;
- i *costi fissi* che sono indipendenti dall'impegno.

Per esempio, in una attività (della fase di) di realizzazione, esistono spesso dei considerevoli costi fissi che riguardano l'acquisizione di strumenti e la preparazione degli ambienti di sviluppo.

Il costo (totale)  $C$  di un progetto si calcola semplicemente sommando il costo delle singole attività:

$$C = \sum_j (\text{costo variabile(attività}_j) + \text{costo fisso(attività}_j) ) = \\ = \sum_j (\text{impegno(attività}_j) \times \text{tariffa risorse(attività}_j) + \text{costo fisso(attività}_j)).$$

Occorre osservare che, mentre è semplice e naturale calcolare il costo totale del progetto come somma dei costi delle singole attività, non lo è per la durata totale e per l'impegno totale. La prima non è affatto la somma delle durate delle attività, ma si deduce dal Gantt (cioè dalla funzione  $t$ ) che tiene conto del parallelismo, dei vincoli, ecc. Per l'impegno totale si noti che, in generale, ogni attività ha una sua unità di misura, a seconda del tipo di risorse che richiede e quindi, a rigore, gli impegni "parziali" non si possono sommare per dare origine a un impegno totale associato al progetto, nozione che, appunto, non è possibile definire.

La figura 1.8 illustra il rapporto logico tra le entità viste: a una attività è associato un impegno; se ad essa si assegnano delle risorse se ne può calcolare la durata e il costo

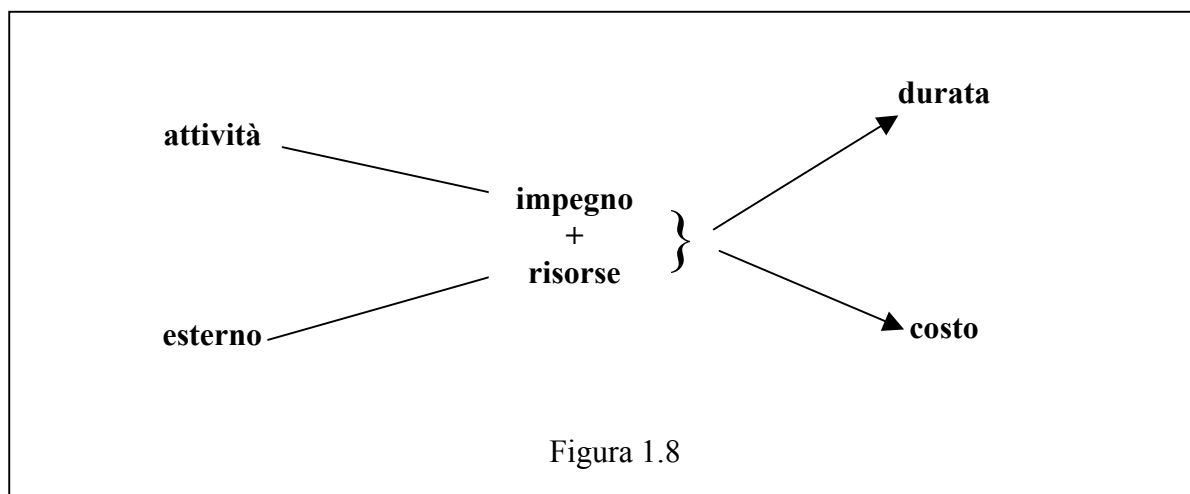


Figura 1.8

## Tipi di attività

Ricapitolando, tra l'impegno  $I$  associato a un attività, la sua durata  $d$ , e la quantità di risorse assegnate  $r$  esiste la relazione

$$I = d \times r.$$

Finora sono state considerate solamente attività a *impegno fisso*, cioè quelle in cui è dato  $I$ , vengono assegnate le risorse e viene calcolato  $d$ ; questo è il tipo più comune nei progetti informatici.

Esistono anche *attività a durata fissa* e *attività a risorse fisse*.

Un esempio di attività a durata fissa è il calcolo del tasso di errori del funzionamento di un impianto complesso; questo si ottiene per esempio controllando l'impianto per un periodo predeterminato (per esempio un settimana) con un numero opportuno di operatori. In questo caso il numero di operatori da impegnare dipende dalla accuratezza con cui si vuole valutare il fenomeno: è assegnato  $d$ , vengono allocate le persone e viene calcolato  $I$ .

Un esempio di attività a risorse fisse è il *test di stress* di un sistema *online* che può richiedere 5 operatori (reali e non simulati) che adoperano a pieno ritmo il sistema. In questo caso l'attività nasce con  $r$  assegnato, si alloca  $d$  (per esempio 2 giorni) e si calcola  $I$ .

In condizioni molto particolari, che possono tuttavia presentarsi negli esempi di progetti che saranno presi in considerazione, si presentano altri due tipi di attività: le *attività condizionate* e le *attività non valutabili*.

Le prime sono quelle la cui necessità è condizionata al verificarsi di eventi di cui si conosce solo la possibilità di accadere; le seconde sono quelle per cui non è possibile determinare, neppure in modo approssimato, l'impegno e quindi il valore della funzione  $d$ . Esempio tipico sono attività relative alla realizzazione di quei moduli *software* che, per la difficoltà o la novità, sono praticamente imprevedibili<sup>2</sup>.

In questi casi sono possibili due atteggiamenti:

- fare una valutazione provvisoria e trattare gli scostamenti durante il controllo,
- usare attività con parametri a valore in un intervallo e che determinano quindi durate totali e costi totali collocati in intervalli.

In pratica, il calcolo per intervalli è relativamente poco utilizzato per i progetti trattati in questo contesto e si preferisce usare un valore stimato.

### Vincoli

Sono stati proposti come esercizio la scrittura di programmi per allocare le attività nel tempo, cioè per calcolare  $t$ , tenendo conto della relazione di precedenza fra le attività e con durata totale minima (esercizio E4) o con una sola attività alla volta (esercizio E5). In realtà, nelle applicazioni esistono altre condizioni di cui tener conto:

- la disponibilità delle risorse,
- i vincoli sulle attività.

La prima condizione potrebbe imporre di spostare in avanti (rispetto alla soluzione standard dell'esercizio E4) l'inizio di una attività fino a quando non sono disponibili tutte le risorse necessarie per effettuarla.

Per descrivere la seconda condizione si associano ad alcune attività dei vincoli, che possono essere di tre tipi: *flessibili*, *deboli* e *forti*.

Vincoli flessibili: si riferiscono alla direzione del tempo;  
 - il più presto possibile: ASAP (*as soon as possible*),  
 - il più tardi possibile: ALAP (*as late as possible*);

Vincoli deboli: si riferiscono a intervalli aperti;  
 - iniziare non prima di  $t_0$ ,  
 - iniziare non oltre  $t_0$ ,  
 - finire non prima di  $t_1$ ,  
 - finire non oltre  $t_1$ ;

Vincoli forti: si riferiscono a punti o intervalli chiusi;  
 - iniziare a  $t_0$  (o tra  $t_1$  e  $t_2$ ),  
 - finire a  $t_1$  (o tra  $t_3$  e  $t_4$ ).

Naturalmente, in prima assegnazione, possono verificarsi incompatibilità tra vincoli o tra questi e le disponibilità delle risorse. In generale, queste incompatibilità non possono essere trattate da un programma, ma devono essere discusse e risolte con compromessi dai responsabili del progetto.

---

<sup>2</sup> Naturalmente l'imprevedibilità è solo "locale", cioè si riferisce al risultato di una singola attività e non ha influenza rilevante sul progetto nel suo insieme.

### Esercizi

E7. Costruire un programma che partendo dalla “soluzione *standard*” (dell’esercizio E4) prende in considerazione i vincoli su (alcune del)le attività e termina o con la segnalazione “vincoli incompatibili” o con una nuova funzione  $t$  che rispetta i vincoli. Esplicitare la struttura dei dati che descrive i vincoli associati agli elementi dell’insieme  $P$ .

### Macroattività (o attività di riepilogo)

Un Gantt reale può essere composto anche da parecchie centinaia di attività; per dominare questa complessità e rendere il diagramma maneggiabile e comprensibile si adotta il metodo di “riassumere” alcune attività in *macroattività*.

Dato uno schema di progetto  $(P, \infty, d)$ , un insieme di attività  $A = \{a_1, a_2, \dots, a_n\}$  può essere riassunto in una macroattività se per ogni  $b \in P$ :

- se  $\exists j$  tale che  $b \leq a_j$  allora anche  $b \leq a_i$ , per ogni  $i$ ,
- se  $\exists j$  tale che  $a_j \leq b$  allora anche  $a_i \leq b$ , per ogni  $i$ .

In questo caso si introduce una macroattività  $a \equiv \{a_1, a_2, \dots, a_n\}$  detta anche *attività di riepilogo*. Si confronti questa definizione con quelle di sottoschema e fase.

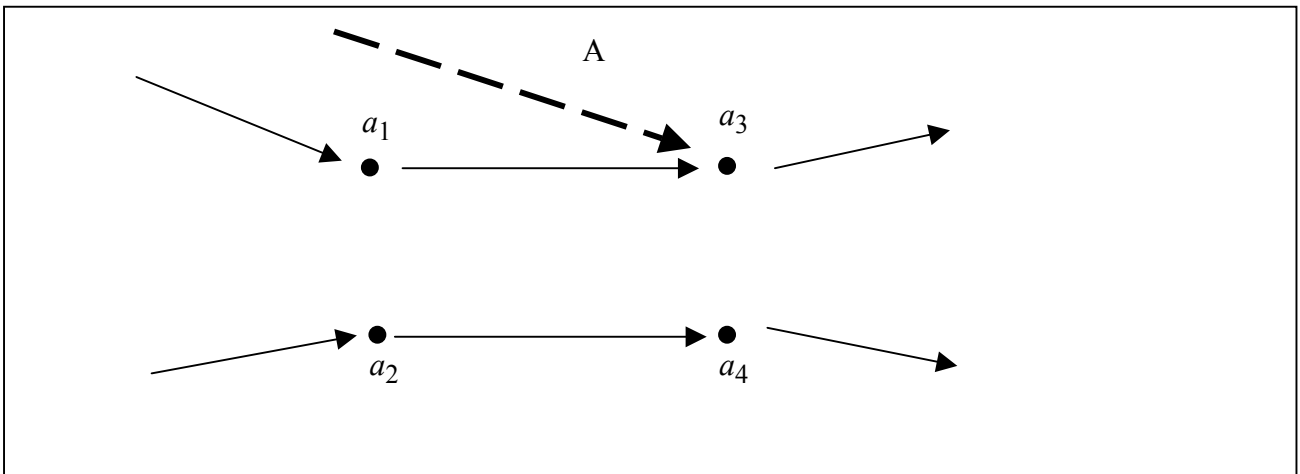


Figura 1.9

Un esempio è riportato in figura 1.9, in cui la freccia  $A$  è non ammessa perché le quattro attività  $a_1$ ,  $a_2$ ,  $a_3$ ,  $a_4$  possano costituire una macroattività; in figura 1.10a 1.10b sono riportati due differenti piani di progetto relativi a queste attività.

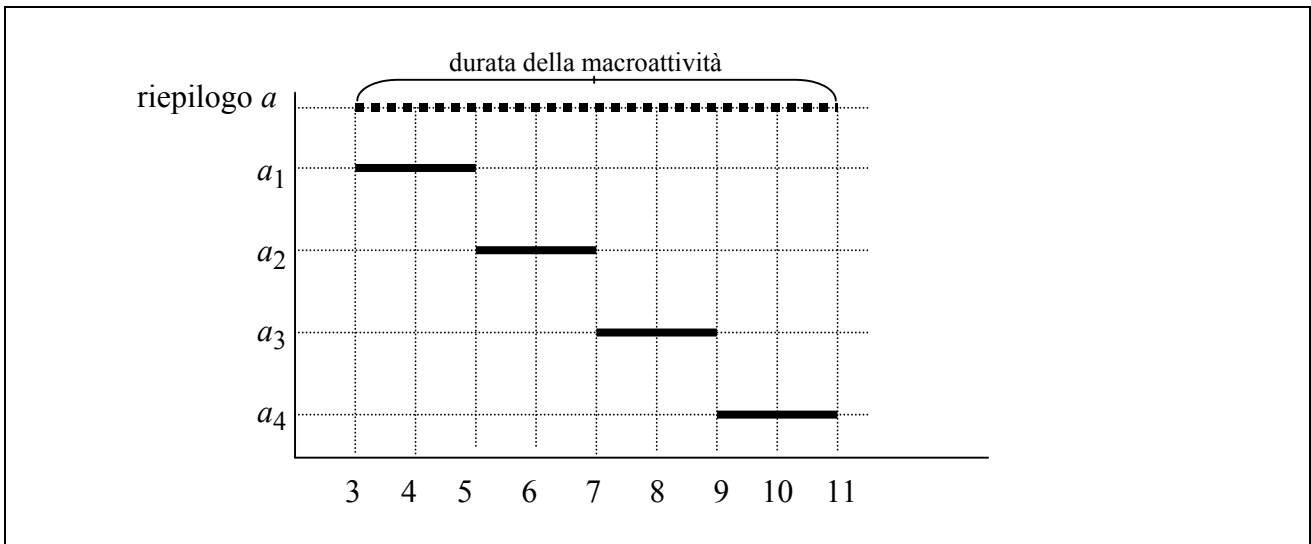


Figura 1.10a

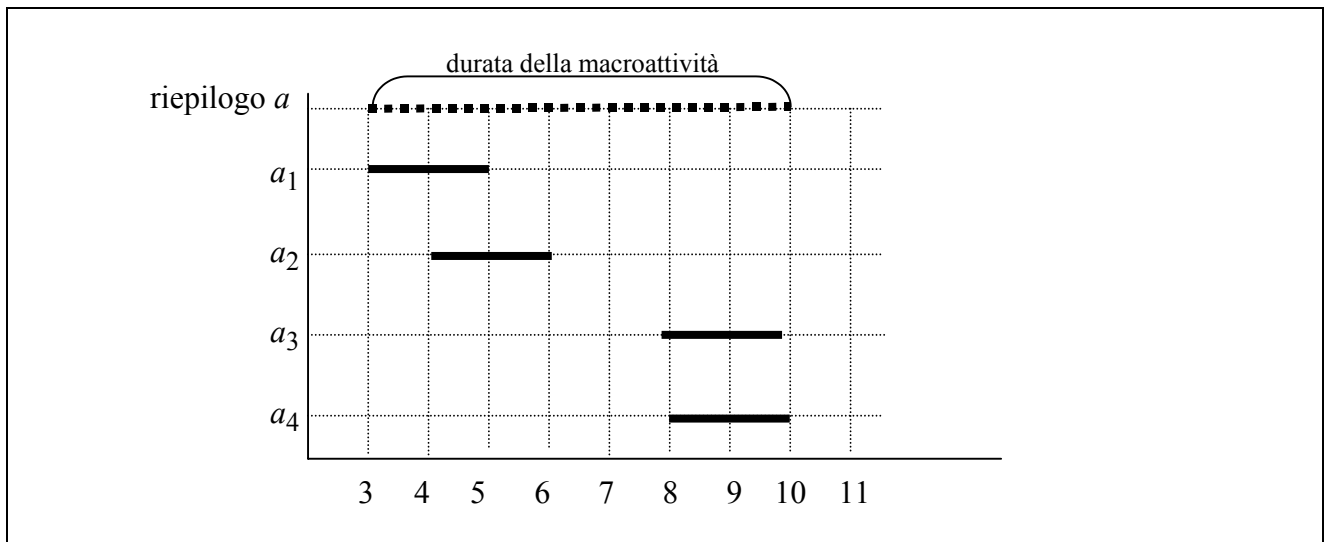


Figura 1.10b

Particolare attenzione deve essere posta per valutare la funzione  $d$  di una macroattività; nel caso particolare che le  $a_j$  formino una catena, una scelta possibile (ma non necessaria) è

$$d(a) = \sum_j d(a_j)$$

### Esercizi

E8. Determinare tutte le informazioni necessarie per il calcolo della durata di una macroattività.

E9. Date tutte le informazioni necessarie per calcolare la durata di una macro attività, scrivere un programma per eseguire il calcolo.

L'uso delle macroattività (insieme alle fasi) offre un metodo fondamentale per derivare *top down* il WBS completo di un progetto complesso. Questa possibilità è conseguenza del fatto che possono essere costruite gerarchie di macro attività in analogia alla scomposizione di problemi complessi in sottoproblemi via via più semplici.

### Percorso critico

Riprendendo quanto finora detto, per ottenere il Gantt di un progetto è bene partire costruendo l'insieme delle fasi del progetto che si intende attuare.

Ciascuna fase del progetto può essere “esplosa” in attività descritte da un WBS; ogni attività può (anche, a sua volta) essere trattata come una macroattività che deve essere opportunamente scomposta per la fattispecie in esame.

Il processo di scomposizione successiva va avanti (almeno) fino a quando è possibile, per ogni attività, determinare (con buona approssimazione) l'impegno corrispondente, il tipo di risorse necessarie e gli eventuali vincoli.

Parallelamente occorre descrivere le risorse a disposizione e la loro effettiva disponibilità. Da ultimo occorre calcolare la durata e il costo di tutte le attività e la loro allocazione nel tempo (la funzione  $t$ ).

In questo modo rimangono determinati il costo e la durata totale del progetto che vengono indicati con  $C_b$  e  $T_b$  per sottolineare che si è in fase previsionale (l'indice  $b$  sta per *budget*).

Se in un Gantt si sposta la fine di una attività  $a$  dal tempo  $t_f(a) = t(a) + d(a)$  al tempo  $t_f(a) + \Delta a$ , perché per esempio è cambiato  $d(a)$  diventato  $d(a) + \Delta a$ , può succedere che occorre o no cambiare (globalmente) la funzione  $t$  (che deve tener conto dell'ordine tra le attività). Si danno essenzialmente tre casi:

- non occorre cambiare  $t$ ; si chiama  $\Delta_f a$ , “flessibilità libera” associata in quel Gantt all'attività  $a$ , il massimo cambiamento che permette di non cambiare  $t$ ;
- se  $\Delta a > \Delta_f a$  occorre ricalcolare la  $t$ , cioè determinare una funzione  $t'$  che coincide con  $t$  sulle attività che precedono l'attività  $a$ ;
- si chiama  $\Delta_m a$ , “flessibilità massima” associata in quel Gantt all'attività  $a$ , il massimo  $\Delta a$  per cui si ha  $t'(\omega) = t(\omega)$ .

Una attività  $a$  in un Gantt si dice *attività critica* quando  $\Delta_m a = 0$ ; quindi, per una attività critica un  $\Delta a > 0$  “sposterebbe” una qualche altra attività che a sua volta ne “sposta” un'altra e così via fino a “spostare”  $\omega$ , cioè a cambiare  $t(\omega)$ . Esiste quindi un *percorso critico* cioè una catena

$$\alpha = a_0, a_1, \dots, a_n = \omega$$

con  $t(a_j) = t(a_{j-1}) + d(a_{j-1})$ , con  $j=1, \dots, n$ , per cui cioè  $\Delta_m a_j = 0$ .

### Esercizi

E9. Dato uno schema di progetto, determinare la struttura dati che rende facile il disegno effettivo del Gantt.

E10. Dato uno schema di progetto e la struttura dati determinata con l'esercizio precedente, trovare tutti i percorsi critici.

## 1.4 Verifica e controllo

L'attività finora descritta è di “pianificazione del progetto” che porta al *documento di previsione* e rende conto dei tempi e dei costi previsti.

Tale documento contiene (almeno) i seguenti dati:

- a) l'insieme delle attività e degli eventi con la loro relazione d'ordine (WBS),
- b) la valutazione dell'impegno, dei vincoli e del tipo di risorse per ogni attività,
- c) le risorse utilizzabili per il progetto, le tariffe e la disponibilità nel tempo,
- d) l'assegnazione delle risorse alle attività e la determinazione della durata e del costo,

- e) allocazione delle attività nel tempo nel rispetto della disponibilità delle risorse e dei vincoli (piano di progetto o Gantt),
- f) la durata e il costo totali e i cammini critici.

Preparato il piano, inizia il vero e proprio *project management* che consiste essenzialmente nelle seguenti azioni (spesso da condurre in parallelo):

- verifica dell'avanzamento,
- controllo,
- valutazione dei rischi ed *expediting*.

#### 1.4.1 Verifica dell'avanzamento

Si cominci con l'osservare che in fase di pianificazione, in generale, non si può porre  $t(\alpha) = 0$  e spostare l'origine a un  $t_0$  data effettiva di inizio del progetto, a causa della presenza (possibile) di vincoli forti, che di solito sono date assolute e non relative all'inizio del progetto.

La verifica del progetto (da confrontare con il *tracking* visto precedentemente) consiste nel

- determinare le funzioni  $d$  e  $t$  che associano i valori "sperimentali" alle attività già iniziate alla data della verifica ( $t$  e  $d$  sono funzioni parziali),
- riallocare le risorse se necessario,
- ricalcolare  $t$  e  $d$  da quella data in poi.

Si pongono almeno due problemi: il primo è legato alla determinazione della funzione  $d$  sperimentale relativa alle attività iniziate ma non ancora terminate nel momento della verifica; il secondo è legato alla frequenza con cui si devono effettuare le verifiche.

Per il primo problema occorre sostanzialmente prevedere la durata delle attività iniziate ma non ancora terminate; questo in generale è un processo soggetto ad errore e comporta il suggerimento pressante di utilizzare un WBS con attività "brevi" (spezzando quelle che hanno una durata "eccessiva"<sup>3</sup>). La risposta al secondo problema si riduce alla regola empirica di fare le verifiche a intervalli di durata pari al 5% della durata totale del progetto; per i progetti più piccoli o più grandi di quelli oggetto di questa trattazione, questa indicazione può non essere applicabile.

Detta  $V_j$  la  $j$ -esima verifica, indichiamo con  $d_j$  e  $t_j$  le funzioni durata e i tempo di inizio conseguenti alla verifica. Per le attività già terminate,  $d_j$  e  $t_j$  sono i valori constatati sperimentalmente; per le attività in corso, devono essere stimate le  $d_j$ , mentre le  $t_j$  sono quelle effettive; per le attività rimanenti, si devono ricalcolare le  $t_j$ , mentre le  $d_j$  vanno ricalcolate solo se è variata l'allocazione delle risorse. Si ricorda che alla verifica  $V_j$  per l'attività  $a$ , se  $t_j(a) \neq t_{j-1}(a)$ , allora viene calcolato lo *spostamento*

$$\Delta a = t_j(a) - t_{j-1}(a) + d_j(a) - d_{j-1}(a)$$

e, indicati con  $\Delta_f a$  e  $\Delta_m a$  la flessibilità libera e massima della attività  $a$  nel Gantt successivo alla verifica  $V_{j-1}$ , si hanno i seguenti casi:

- $\Delta a < \Delta_f a$  : lo spostamento non influisce sulle attività successive;
- $\Delta_f a < \Delta a < \Delta_m a$  : lo spostamento influisce su alcune attività successive, ma viene riassorbito, cioè non influisce su  $\omega$ ;
- $\Delta_m a < \Delta a$  : lo spostamento influisce su  $\omega$ , cioè  $t_j(\omega) > t_{j-1}(\omega)$ .

Ad ogni verifica  $V_j$  si ricalcola anche la funzione costo determinando  $C_j$ ; il costo di una attività può variare per due motivi:

- perché l'attività ha avuto una durata diversa da quella preventivata precedentemente, a parità di risorse assegnate;

<sup>3</sup> Naturalmente la compilazione del documento di previsione di un progetto è un processo iterativo: compiuto un passo, ad esempio il calcolo di  $d$ , spesso occorre tornare indietro per modificare decisioni o parametri già fissati.

- perché alla verifica è variata l'assegnazione delle risorse.

Durante la realizzazione di un progetto, normalmente in occasione delle verifiche, può anche accadere che occorra rivedere i WBS, cioè inserire, cancellare e scomporre attività e questo comporta una variazione (anche sostanziale) del piano del progetto. Inoltre può succedere che i costi e i tempi totali variano (in genere in aumento) perché le attività previste prendono più tempo e/o costano di più e qualche volta devono essere inserite nuove attività anche di elevata consistenza.

Riassumendo: la verifica registra ciò che è accaduto e, in generale, ricalcola un nuovo piano, ma non genera una previsione "ragionata"; quindi non si usa ciò che realmente è accaduto per migliorare la conoscenza del futuro. Dopo ogni verifica è consuetudine creare i seguenti documenti:

- il Gantt riaggiornato (che può essere ridisegnato sovrapposto al precedente);
- l'elenco delle attività iniziate o terminate in ritardo;
- l'elenco delle attività che hanno assorbito un quantità di risorse superiore al previsto (o perché era stata sbagliata la stima dell'impegno o perché si sono verificati eventi imprevedibili);
- la lista delle attività pianificate che non sono state svolte;
- l'elenco dei costi reali delle attività completate.

### 1.4.2 Controllo

Il controllo viene effettuato periodicamente e di norma dopo una verifica; consiste nel prevedere come sarà l'andamento finale del progetto se non vengono presi provvedimenti e suggerisce anche cosa deve essere fatto per modificare questa previsione. La verifica consente di prendere atto degli scostamenti constatati e di calcolarne la "ovvia" conseguenza sulla parte di progetto ancora da realizzare, mentre il controllo, tenuto conto di quanto avvenuto fino a quel momento, consente di effettuare previsioni sui reali scostamenti futuri.

Per quanti sforzi si facciano durante la pianificazione del progetto, difficilmente sarà possibile rispettare completamente tutti gli impegni e le scadenze previsti. Una tecnica ampiamente usata è quella di pianificare tempi e costi con un "margine di sicurezza" in modo da costruirsi una riserva da usare opportunamente per fare fronte agli imprevedibili e non rendere necessaria (se non in casi estremi) la ripianificazione del progetto.

#### Metodo del controllo con la riserva

Il committente (chi paga ed è interessato al progetto) fissa (in generale mediante un contratto) due parametri:

- $T_d$  che è il termine entro cui richiede che il progetto sia terminato;
- $C_d$  che è il costo totale che non è disposto a superare.

Il metodo del controllo con la riserva consiste nel pianificare il progetto in modo che i seguenti parametri:

- $T_b$  che è l'intervallo di tempo entro cui si prevede di terminare il progetto  $T_b = t(\alpha) - t(\omega)$ ,
- $C_b$  che è il costo totale previsto per il progetto,

siano minori dei corrispondenti fissati dal committente.

Si chiamano *indici di riserva* i rapporti (maggiori di 1)

$$C_r = C_d/C_b \quad \text{e} \quad S_r = T_d/T_b$$

La riserva è illustrata nella figura 1.11. Con riferimento alla figura 1.12, supponiamo di fare il controllo di un progetto in un certo momento, che in figura è indicato con  $T$ ; le previsioni riguardo al costo possono non essersi avverate per due motivi: o si è svolta una quantità di lavoro diversa (probabilmente minore) di quella preventivata oppure il costo del lavoro fatto è risultato diverso (probabilmente superiore) a quello preventivato.

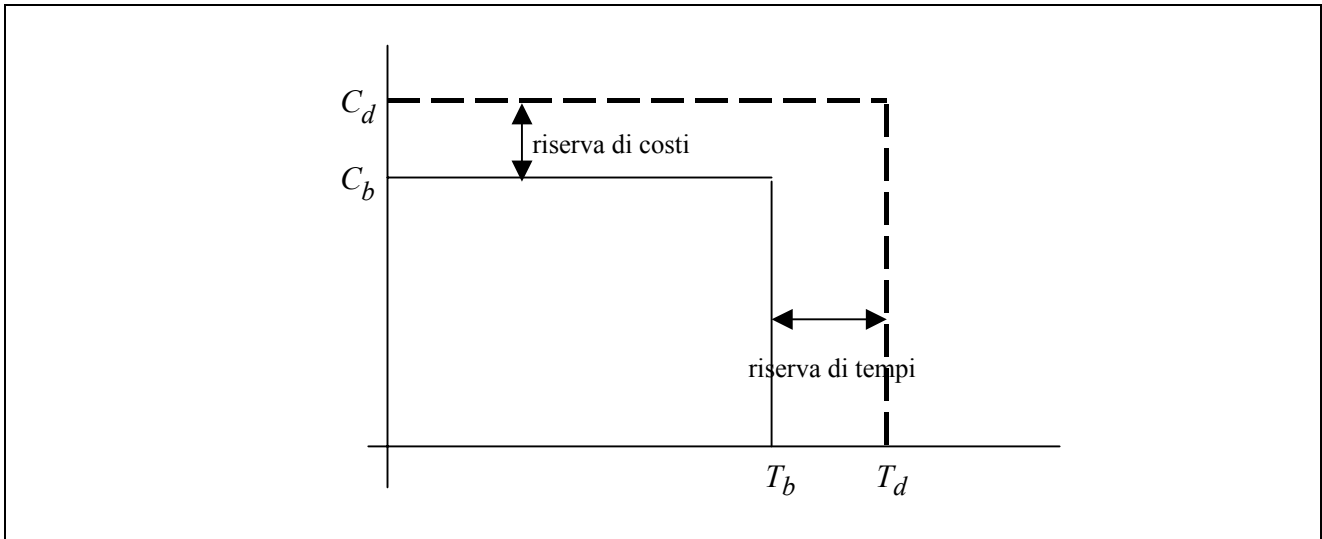


Figura 1.11

Si calcolano i seguenti parametri

- $C_{bb}$  che è il costo a piano per il lavoro a piano alla data in cui si effettua il controllo;
- $C_{bf}$  che è il costo a piano per il lavoro effettivamente fatto al momento del controllo (prevedibilmente si è fatto un lavoro minore del previsto e quindi la spesa pianificata per tale lavoro è “bassa”);
- $C_{ff}$  che è il costo reale del lavoro fatto fino al momento del controllo (prevedibilmente si è fatto meno lavoro del previsto che, comunque, è costato più di quanto messo a piano).

Con questi parametri (vedi figura 1.12) si costruiscono i due indici che stimano approssimativamente l'andamento dei costi e dei tempi:

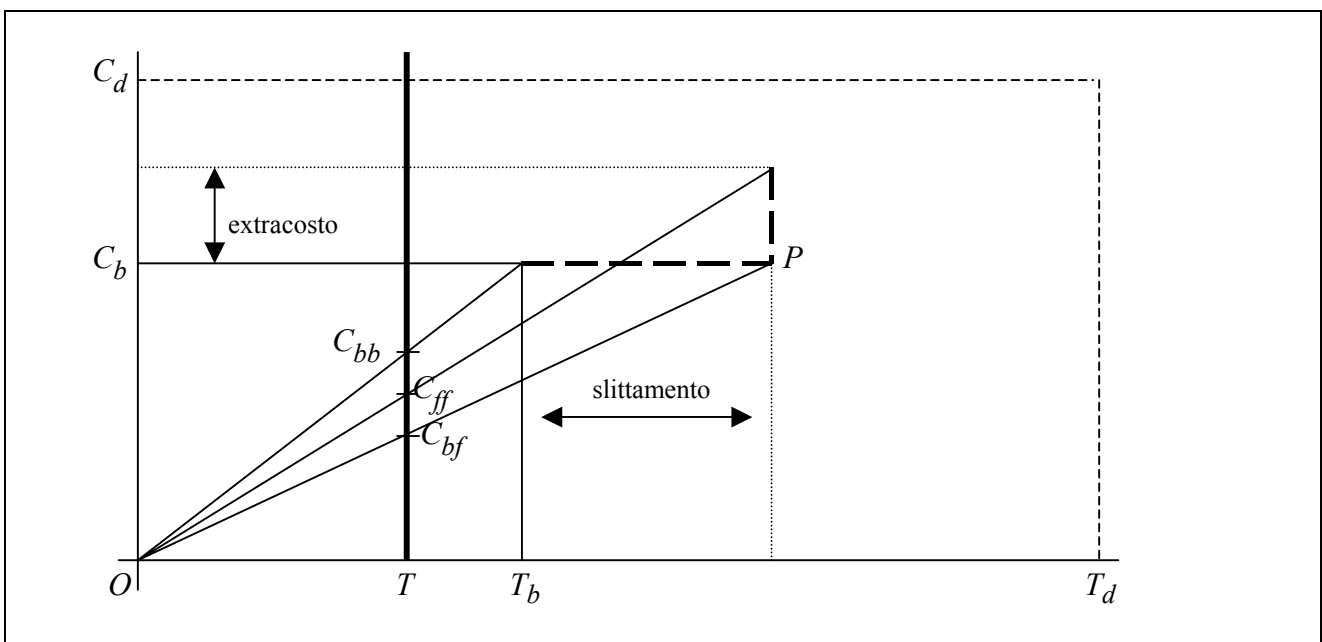


Figura 1.12

- $CI = C_{bf}/C_{ff}$  misura approssimativamente l'andamento dei costi,

-  $SI = C_{bf}/C_{bb}$  misura approssimativamente l'andamento dei tempi.

Quando lo stato d'avanzamento del progetto è in accordo con il piano gli indici sono maggiori o uguali a 1; un'osservazione empirica, ma fortemente cogente, è che  $CI$  e  $SI$  sono poco significativi nei primi (due o tre) controlli e variano fortemente da controllo a controllo. Successivamente, quando è passato un intervallo di tempo compreso fra il 15 e il 20% del tempo preventivato per l'intero progetto, oppure quando si è speso una quantità di denaro compresa fra il 15 e il 20% del costo complessivo, i due rapporti tendono a stabilizzarsi o variano lentamente, a meno che non vengano prese decisioni per apportare qualche modifica sostanziale alla conduzione del progetto.

Quando i valori sono stabilizzati danno una indicazione attendibile su come sta andando il progetto: un valore maggiore di 1 indica che il progetto sarà completato senza intaccare la riserva, mentre un valore minore di 1 indica che la riserva può non bastare per portare a termine il progetto.

La figura 1.12 è stata ottenuta dalla precedente 1.11 nella seguente maniera; si è disegnata la linea verticale al tempo  $T$ ; su tale linea si sono riportati i tre valori  $C_{bb}$ ,  $C_{ff}$ ,  $C_{bf}$  che, presumibilmente sono nelle reciproche posizioni mostrate in figura. Le tre rette che passano per l'origine  $O$  e per ciascuno dei tre punti tracciati hanno come pendenza le "velocità di spesa" rispettivamente *prevista*, *effettiva* e *virtuale*. Si è poi costruito il punto  $P$  intersezione tra la retta  $(O, C_{bf})$  e la orizzontale per  $C_b$ ; la ascissa di questa intersezione dà una previsione ragionevole del tempo di fine del progetto: si noti come si prevede tale tempo considerando la velocità di spesa virtuale al tempo  $T$  e utilizzando tale velocità per esaurire il *budget*, cioè per raggiungere il costo previsto. La intersezione tra la retta  $(O, C_{ff})$  e la verticale per il punto  $P$  ha come ordinata una previsione ragionevole per la spesa totale del progetto: questo equivale a supporre la velocità di spesa futura eguale a quella effettiva per tutta la durata prevista del progetto.

Le linee guida per suggerire le decisioni più appropriate da prendere durante le fasi di controllo (affinché il progetto rispetti i vincoli imposti dal committente) fanno uso del così detto metodo dei semafori. Questo consiste nel considerare due segnali (semafori), uno per i costi e uno per i tempi, che possono assumere i valori verde, giallo e rosso. Il valore verde del semaforo indica che la pianificazione viene rispettata, il valore giallo indica che si sta intaccando la riserva e il valore rosso segnala che, se non si interviene, non verranno rispettati i vincoli imposti dal committente.

Semaforo dei costi	condizioni
verde	$CI^{-1} \leq 1 \leq C_r$
giallo	$1 \leq CI^{-1} \leq C_r$
rosso	$1 \leq C_r \leq CI^{-1}$

Semaforo dei tempi	condizioni
verde	$SI^{-1} \leq 1 \leq S_r$
giallo	$1 \leq SI^{-1} \leq S_r$
rosso	$1 \leq S_r \leq SI^{-1}$

Costo	Tempo	Azioni raccomandate
verde	verde	tempi e costi sono in linea, o migliori, delle previsioni; probabilmente il progetto terminerà con una riserva non utilizzata di tempi e costi.
verde	giallo	si sta intaccando la riserva dei tempi; è bene introdurre o aumentare il lavoro stra-

		ordinario.
verde	rosso	la riserva di tempo non è sufficiente per terminare il Progetto; occorre aumentare ancora lo straordinario o il personale.
giallo	verde	si sta intaccando la riserva dei costi; si deve diminuire lo straordinario.
giallo	giallo	si sta intaccando sia la riserva dei costi sia quella dei tempi; si deve riaggiustare l'assegnazione del personale sulle varie attività.
giallo	rosso	migliorare l'assegnazione del personale e prendere in considerazione l'idea di rinegoziare $T_d$ .
rosso	verde	diminuire il lavoro straordinario e/o le persone assegnate al progetto.
rosso	giallo	migliorare l'assegnazione delle persone e prendere in considerazione l'idea di rinegoziare $C_d$ .
rosso	rosso	rinegoziare $T_d$ e $C_d$ ; riconsiderare gli obiettivi del progetto; cambiare il capo progetto.

### 1.4.3 Valutazione dei rischi ed *Expediting*

La valutazione dei rischi e lo *expediting* sono azioni volte a minimizzare lo scostamento fra ciò che accade e ciò che era stato pianificato. La prima cerca di risolvere i problemi, qualora questi si siano presentati; la seconda serve e viene attivata per evitare che i problemi si presentino. Queste azioni hanno a che fare con il problema più generale di trattare i rischi insiti in progetti impegnativi.

In generale si possono classificare tre tipi di rischi.

- a. difficoltà di prevedere e stimare,
- b. ipotesi e assunzioni, fatte durante la pianificazione, che possono non verificarsi,
- c. eventi imprevisti.

Le aree di rischio (cioè le aree dove è prevedibile che nascano problemi che impediscono il rispetto della pianificazione) per i progetti informatici sono essenzialmente le seguenti.

1. **Applicazione**; è relativa ai problemi che nascono dalle difficoltà insite nel tipo di applicazione informatica che si deve realizzare: costruire un *software* per il controllo di processo di un impianto completamente nuovo non ancora in esercizio pone dei rischi decisamente maggiori che realizzare un sistema di contabilità con regole definite da una normativa ben nota.
2. **Personale**; è relativa ai problemi delle persone coinvolte nel progetto connessi con l'esperienza, le competenze, gli aspetti caratteriali, il *turnover*, la disomogeneità culturale, ecc.
3. **Progetto**; è relativa ai problemi (molto più frequenti di quanto ci si può aspettare) dovuti alla non chiarezza riguardo agli obiettivi e a vincoli non enunciati sui tempi di realizzazione e sui costi (in relazione all'intero progetto o a fasi significative).
4. **Metodologie**; comprende i rischi connessi alla non completa comprensione e condivisione (da parte delle persone coinvolte) delle varie metodologie impiegate nel progetto (di sviluppo, di test, di *management*, ecc); è particolarmente elevato in caso di nuove tecnologie.
5. **Hw e Sw di base**; si riferisce ai rischi connessi con la presenza di errori nel *software* usato, alla disomogeneità degli ambienti (tra sviluppo, *test* e gestione), alla mancanza di specifiche formali, ecc.
6. **Changeover**; riguarda le difficoltà (che sono estremamente rilevanti per la classe di progetti trattati) connesse con la messa in opera del sistema informatico realizzato (tipicamente un sistema informativo aziendale) che induce cambiamenti organizzativi, procedurali o di comportamento.
7. **Forniture**; si riferisce ai ritardi o alle non conformità dei prodotti che devono essere acquisiti durante il progetto.

8. **Fattori esterni**; quest'area comprende le cose più varie: la disponibilità dei locali nei quali viene effettuato lo sviluppo, il cambiamento di legislazione o di normative che influenzano le materie coinvolte nel progetto, l'andamento dei mercati che influiscono sui costi, ecc.
9. **Sicurezza e salute**; tiene conto di malattie e incidenti: per esempio se il progetto occupa i mesi da novembre ad aprile di cinque persone è "quasi certa" la perdita di uno/due mesi uomo per malattie da raffreddamento.

Per ogni piano, occorre produrre una lista di rischi dei tipi a) e b) (cioè dovuti alla difficoltà di prevedere e a ipotesi non sicure), per ciascuna delle nove aree, con le relative valutazioni.

La valutazione del rischio si fa usualmente con una formula euristica del tipo

$$\text{valutazione} = \text{"probabilità"} \times \text{impatto}.$$

In questo contesto per le "probabilità" e per l'impatto è consuetudine usare una valutazione soggettiva su una scala da 1 a 10. La tabella che segue è un esempio di lista dei rischi.

	<b>rischio</b>	<b>"prob."</b>	<b>imp.</b>	<b>val.</b>
1	cambio delle specifiche durante la codifica	1	9	9
2	l'analisi richiede più tempo del previsto	3	6	18
3	malattie del personale lungo il percorso critico	4	10	40
4	malattie del personale su attività non critiche	10	3	30
5	la codifica richiede più tempo del previsto	4	4	16
6	la fase di test evidenzia errori di analisi	3	9	27
7	la fase di test evidenzia errori di architettura	2	10	20

Nei casi reali la lista dei rischi comprende spesso centinaia di righe. Nell'esempio visto, i rischi 3, 4 e 6 sono particolarmente gravi; occorre mettere in campo azioni specifiche per affrontarli.

In generale ci sono tre maniere per affrontare i rischi:

- ridurre la "probabilità" o l'impatto;
- approntare un *contingency plan*;
- trasferire il rischio.

Per esempio, per il rischio 3 si può aumentare il personale assegnato alle attività del percorso critico (compatibilmente con i vincoli economici) per diminuirne la probabilità o si può pianificare la possibilità di trasferimento momentaneo di personale da altra attività, dello stesso progetto o di altri progetti (*contingency plan*), o fare un contratto-assicurazione con una ditta di servizi che offra personale con le caratteristiche richieste con piccolo preavviso (trasferimento del rischio). Per il rischio 6 si può rafforzare l'attività di analisi o prevedere in vari momenti prima del test forme di controllo che evidenzino il più tempestivamente possibile difetti o non conformità.

Lo *expediting* è quell'insieme di attività di carattere logistico e organizzativo che servono ad accorgersi che le attività del piano possono svolgersi nella maniera prevista e a prendere tutti i provvedimenti (di carattere economico ed organizzativo) che possono contribuire al rispetto delle previsioni.

### 1.5 CPM e Pert

Dato uno schema di progetto, per esempio  $P = \{\alpha, a, b, c, d, e, f, g, h, \omega\}$ , con la relazione d'ordine che è la chiusura riflessiva e transitiva della seguente

$\{(\alpha, a), (\alpha, b), (\alpha, f), (a, c), (b, d), (b, e), (e, g), (f, g), (c, h), (d, h), (h, \omega), (g, \omega)\}$ ;

le durate delle attività sono mostrate dalla seguente tabella

Attività	Durata
----------	--------

<i>a</i>	6
<i>b</i>	4
<i>c</i>	3
<i>d</i>	4
<i>e</i>	3
<i>f</i>	10
<i>g</i>	3
<i>h</i>	2

Il grafo delle precedenze è mostrato nella figura 1.13;

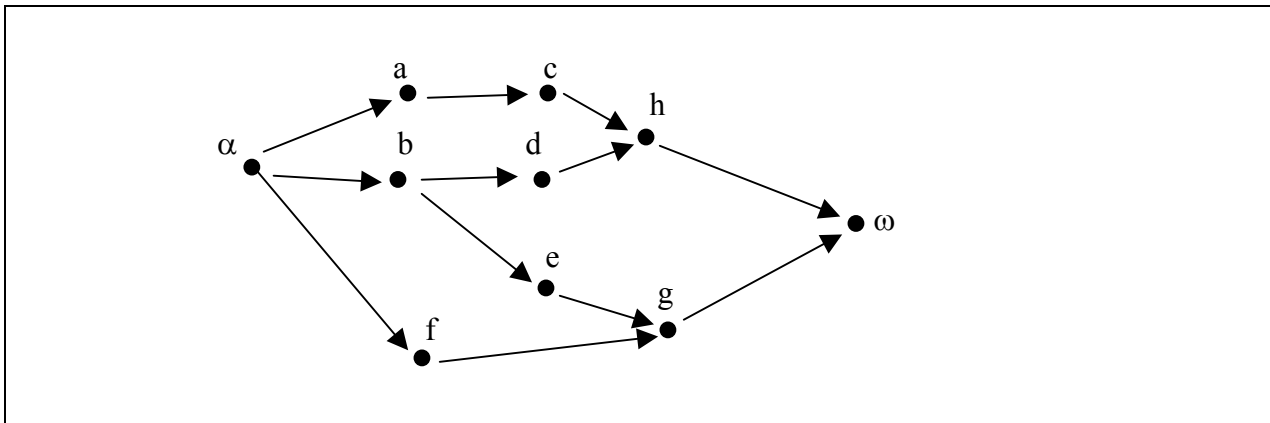


Figura 1.13

Il piano di progetto (quindi la funzione *t*) *standard* (cioè senza vincoli sulle attività e che permette di svolgere il progetto nel tempo minore) è mostrato dal Gantt di figura 1.14.

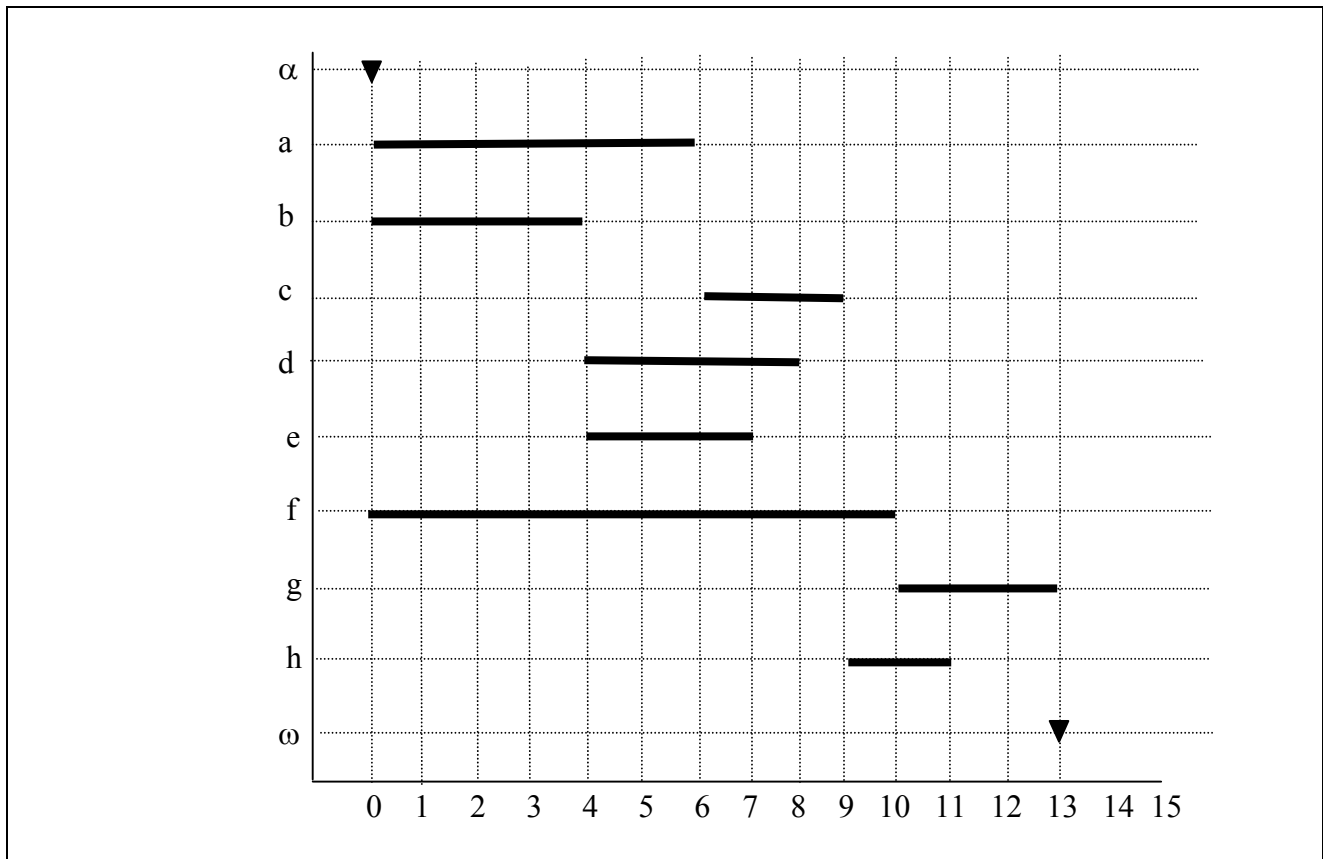


Figura 1.14

Come si è (ampiamente) visto, il grafo orientato delle precedenze mostra, mediante frecce, la relazione d'ordine tra le attività, rappresentate da punti; mentre il Gantt standard mostra la medesima relazione collocando opportunamente nel tempo le attività stesse, rappresentate da segmenti la cui lunghezza è proporzionale alla rispettiva durata. (Si noti che un Gantt qualunque, in generale, non permette di recuperare la relazione di precedenza.)

### 1.5.1 CPM (*Critical Path Method*)

Il grafo (orientato) CPM è una tecnica per rappresentare le stesse informazioni volta ad individuare rapidamente il o i percorsi critici. In tale grafo, i nodi rappresentano eventi; sono sempre presenti  $\alpha$ ,  $\omega$ , cioè l'inizio e la fine del progetto; gli archi rappresentano le attività. Questo grafo può essere ottenuto dal grafo delle precedenze utilizzando il seguente algoritmo, di cui vengono elencati i passi.

1. Si disegna il grafo delle precedenze.
2. Si "indeboliscono" le frecce (facendole diventare tratteggiate).
3. Si sdoppiano tutti i nodi (le attività) mediante due nodi eventi uniti tra loro da una freccia "forte" che ha come etichetta quella del nodo originale (con due attributi: il nome e la durata).
4. Si cancellano (ove possibile) le frecce deboli, fondendo i loro nodi estremi usando le seguenti due regole:
  - un nodo che ha un solo predecessore "debole", cioè unito da una freccia debole, si fonde con esso,
  - un nodo che ha due o più predecessori deboli, ciascuno dei quali non abbia altri successori, si fonde con tutti questi.
5. Le frecce deboli rimaste vengono trasformate in frecce forti dette attività *dummy* (ciascuna con un proprio nome e durata uguale a zero).
6. I nodi sono etichettati con (una struttura dati che ha) 4 attributi.

Gli stadi intermedi, dopo il passo 2 e dopo il passo 4, e quello finale della trasformazione del grafo delle precedenze di figura 1.13 sono riportati in figura 1.15 (a e b) e in figura 1.16, rispettivamente.

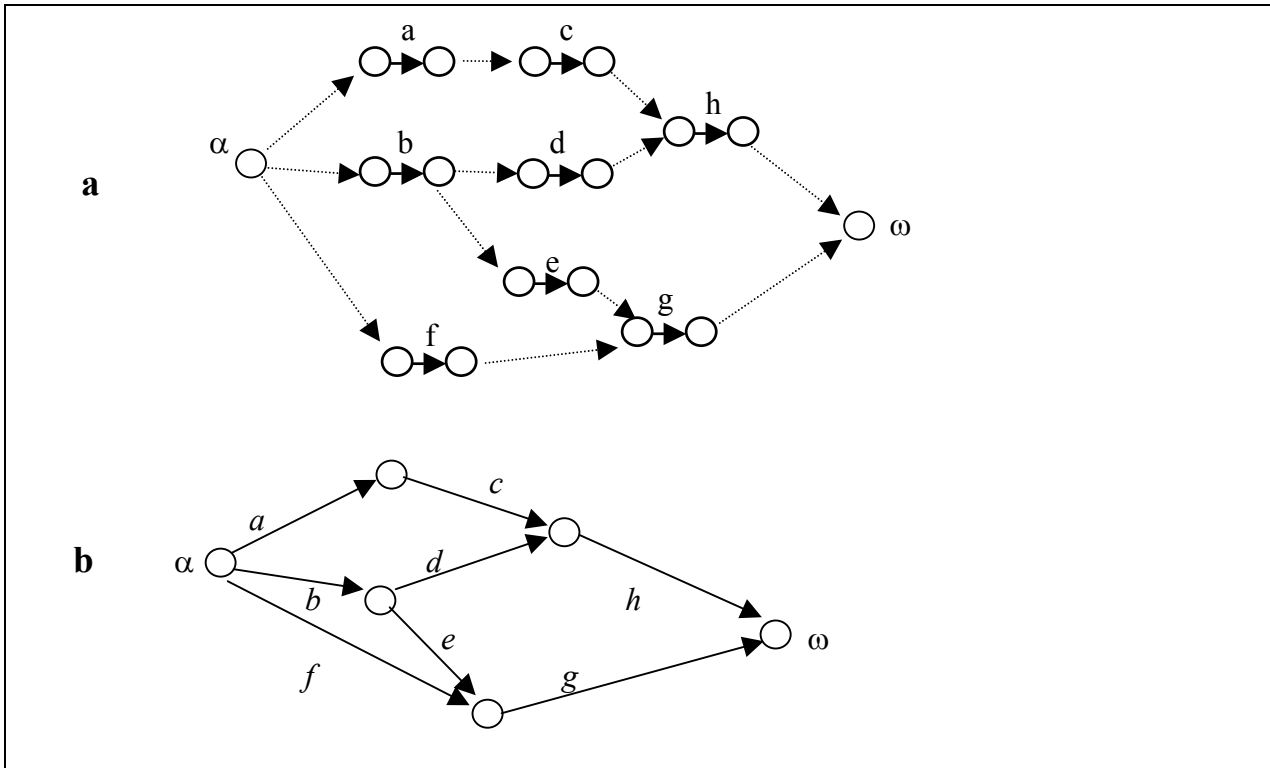


Figura 1.15

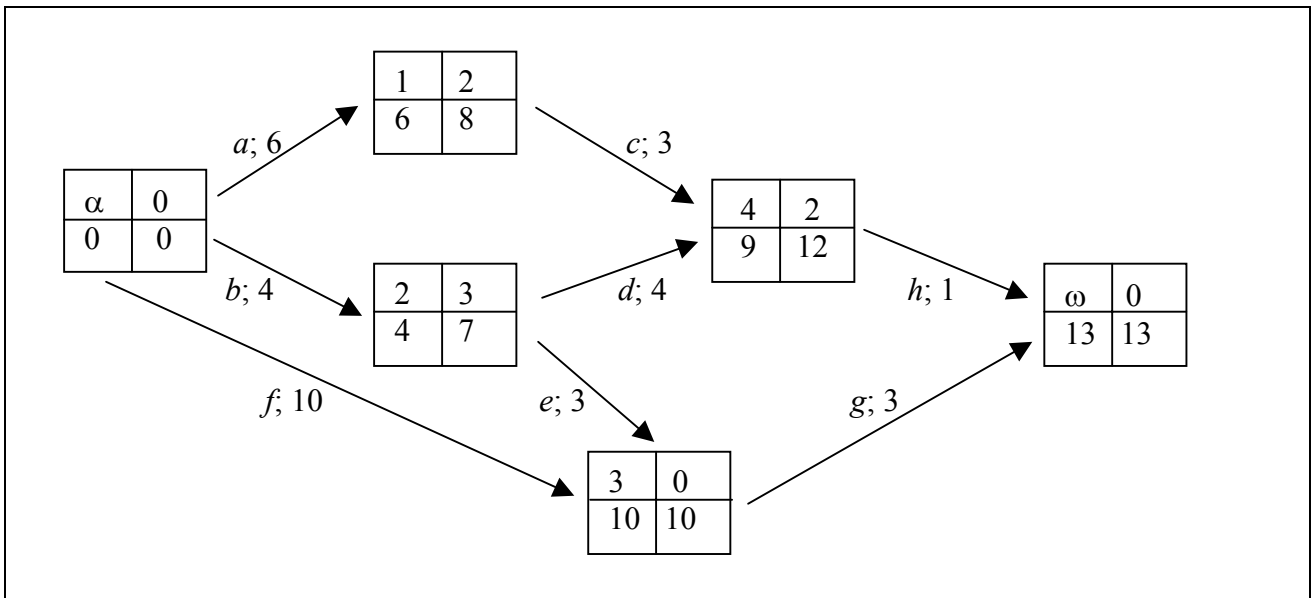


Figura 1.16

### Esercizi

E11. Definire la struttura dati appropriata per rappresentare il grafo delle precedenze e scrivere un programma che lo disegna.

E12. Definire la struttura dati appropriata per rappresentare il grafo CPM e scrivere un programma che lo disegna.

E13. Implementare l'algoritmo che trasforma il grafo delle precedenze nel relativo grafo CPM.

Ricapitolando, in un grafo CPM i nodi rappresentano gli eventi e le frecce rappresentano le attività. La struttura dati per descrivere i nodi ha quattro etichette, come mostrato in figura 1.17: il nome,  $T_m$  (tempo minimo entro cui può accadere l'evento senza dover cambiare  $t(\omega)$ ),  $T_M$  (tempo massimo entro cui può accadere l'evento senza dover cambiare  $t(\omega)$ ) e  $S$  (intervallo di tolleranza, o *slack*); gli archi hanno come etichetta il nome e la durata dell'attività (che è data). Per riempire i tre campi (diversi dal nome) dell'etichetta dei nodi si usano i seguenti tre algoritmi.

nome	$S$
$T_m$	$T_M$

Figura 1.17

**Passo forward, per calcolare il tempo minimo  $T_m$ .**

Si esplora il grafo in avanti, partendo dal nodo iniziale  $\alpha$ ; in  $T_m(\alpha)$  si mette il valore  $t_0$ , cioè il tempo di inizio del progetto; per ogni altro nodo  $x$  che ha tutti i suoi  $k$  predecessori immediati  $y_j$  ( $j = 1, 2, \dots k$ ) con  $T_m(y_j)$  valutata, si pone  $T_m(x) = \max \{T_m(y_j) + durata(y_j, x), j = 1, 2, \dots k \}$ , dove  $j$  numerava le frecce in entrata per il nodo  $x$ .

**Passo backward per calcolare il tempo massimo  $T_M$ .**

Si esplora il grafo all'indietro, partendo dal nodo finale  $\omega$ ; in  $T_M(\omega)$  si copia il valore  $T_m(\omega)$ , cioè il tempo di fine del progetto; per ogni altro nodo  $x$  che ha tutti i suoi  $k$  successori immediati  $y_j$  ( $j = 1, 2, \dots k$ ) con  $T_M(y_j)$  valutata, si pone  $T_M(x) = \min \{T_M(y_j) - durata(x, y_j), j = 1, 2, \dots k \}$ , dove  $j$  numerava le frecce in uscita per il nodo  $x$ .

**Passo slack per calcolare l'intervallo di tolleranza.**

Per ogni nodo  $x$  si pone  $S(x) = T_M(x) - T_m(x)$ . Le catene di nodi per cui  $S(x)$  risulta nullo sono i percorsi critici, cioè quelli che contengono gli eventi che non possono subire ritardi, pena l'aumento della durata del progetto.

**Esercizi**

E14. Implementare l'algoritmo *forward*.

E15. Implementare l'algoritmo *backward*.

E16. Dimostrare che  $S(x)$  è sempre non negativa.

E 17. Usare gli *slack* per ridurre i costi .....

E 18. Sia dato un grafo CPM con i seguenti 8 nodi:  $\alpha$ , B, C, D, E, F, G,  $\omega$ . Le frecce (cioè le attività) e la loro durata (per esempio in settimane) sono date dalla seguente tabella

Nodo di inizio	Nodo di fine	Durata dell'attività
$\alpha$	B	4
$\alpha$	D	3
B	D	5
B	E	8
B	C	5
C	E	6
C	G	4
D	E	7
D	F	9
E	F	10
E	G	7
E	$\omega$	3
F	$\omega$	3
G	$\omega$	5

Disegnare il grafo e trovare il cammino critico e lo *slack* (totale) di ogni attività.

### 1.5.2 Pert

Il metodo CPM, come si è visto serve a determinare facilmente il percorso critico di un progetto e a individuare l'intervallo di tolleranza (o *slack*) in cui i vari eventi possono accadere senza ritardare la fine del progetto. L'assunzione di fondo è che la durata delle attività sia determinabile con precisione; questa assunzione è abbastanza ben verificata quando le attività sono "semplici" e ben individuabili e quindi l'incertezza sulla loro durata è trascurabile: il metodo CPM utilizza le durate delle singole attività (e la relazione di precedenza tra di esse) per calcolare in maniera deterministica la durata del progetto.

Quando le attività sono qualche migliaio oppure non se ne può determinare con sufficiente certezza la durata, questo schema per calcolare la durata del progetto non è più applicabile e si devono usare metodi probabilistici.

Il metodo Pert (*Program evaluation and review technique*) fu introdotto nel 1957 per la gestione del progetto Polaris (la costruzione di un missile che potesse essere lanciato da un sommergibile in immersione) che era composto da migliaia di attività, molte delle quali erano programmi di ricerca e quindi di durata non esattamente prevedibile.

Questo metodo si usa principalmente in caso di grandi progetti con attività non omogenee (per esempio non solamente di sviluppo *software*), alcune delle quali ad alto rischio (come per esempio lo sviluppo di *hardware custom*, cioè progettato e realizzato *ad hoc*, oppure la individuazione di un nuovo combustibile solido, come nel caso del Polaris), quindi è applicabile alla fascia "alta" dei progetti qui considerati. Il Pert fa riferimento a un grafo in cui, come in quelli del CPM, i nodi rappresentano degli eventi (o stati parziali) e le frecce le attività.

Riguardo alle attività, si fanno le seguenti (molto ragionevoli) ipotesi:

- le durate delle attività sono statisticamente indipendenti;
- la durata di una attività segue una distribuzione beta;
- la durata complessiva del progetto segue la distribuzione normale.

Il cammino critico in questo caso ha il significato del cammino per il quale il valore atteso della durata (totale del progetto) è massima.

Per ogni attività vengono stimate tre durate (che in casi particolari possono coincidere):

$d_p$ , che è la durata pessimistica,  
 $d_m$ , durata stimata più probabile,  
 $d_o$ , la durata più ottimistica.

Si suppone inoltre che la differenza fra  $d_p$  e  $d_o$  sia equivalente a 6 deviazioni standard.

Quanto detto, viene illustrato con un esempio che riprende (e generalizza) l'esempio precedente ed è composto dalle medesime 8 attività con la stessa relazione di precedenza; esso è descritto dalla seguente tabella (nella quale la colonna  $d_m$  riporta i medesimi valori della colonna durata del caso CPM).

Attività	$d_o$	$d_m$	$d_p$	$d_e$	$\sigma$
<i>a</i>	5	6	8	6.17	.50
<i>b</i>	3	4	5	4	.33
<i>c</i>	2	3	3	2.83	.17
<i>d</i>	3.5	4	5	4.08	.25
<i>e</i>	1	3	4	2.83	.50
<i>f</i>	8	10	15	10.5	1.17
<i>g</i>	2	3	4	3	.33
<i>h</i>	2	2	2.5	2.08	.08

Assegnati i valori delle prime tre colonne, si calcolano per ciascuna attività il valore atteso  $d_e$  e la deviazione standard  $\sigma$  con le seguenti formule

$$d_e = (d_p + 4d_m + d_o)/6$$

$$\sigma = (d_p - d_o)/6$$

Nei grafi tipo Pert, le frecce hanno una etichetta con tre attributi che sono: il nome, la durata attesa  $d_e$  e la deviazione standard  $\sigma$  (che sono dati). I nodi hanno una etichetta con cinque attributi: il nome, il tempo atteso per il verificarsi dell'evento  $T_e$ , la deviazione standard  $\sigma$ , il *target*  $T$  e la probabilità di rispettarlo. Il *target* rappresenta il tempo massimo entro cui l'evento deve accadere per rispettare i vincoli posti (dal committente, dalla disponibilità delle risorse, ecc.); naturalmente questo attributo può essere indicato solo per alcuni nodi e non per altri.

Per semplicità spesso i nodi sono disegnati solo con i primi quattro attributi, come mostrato in figura 1.18

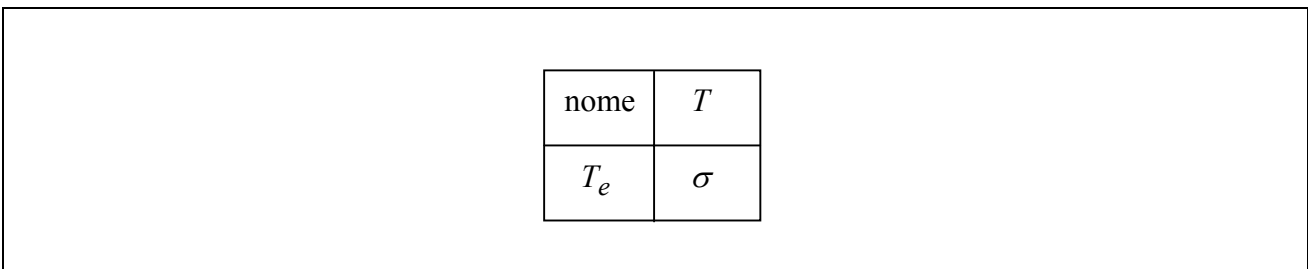


Figura 1.18

Dai dati riportati nella precedente tabella (e dalla relazione di ordine fra le attività) si deriva il grafo mostrato in Figura 1.19.

Nel disegnare il grafo Pert, vengono immediatamente riempite le etichette di ogni freccia (nome, durata attesa e deviazione standard) perché sono dati iniziali assegnati. Per i nodi è dato il nome e, per alcuni di essi, il target; gli altri attributi (tempo atteso, deviazione standard e la probabilità) devono essere calcolati. Esiste tuttavia una differenza fondamentale fra gli algoritmi per calcolare questi attributi rispetto a quelli usati nel caso CPM; in quest'ultimo caso si calcolano nodo per nodo gli intervalli di tempo entro i quali l'evento deve accadere (descritti dagli estremi e dall'ampiezza dell'intervallo). Nel caso Pert invece si "propaga" con metodi probabilistici il valore del tempo atteso per l'evento insieme con la sua deviazione standard.

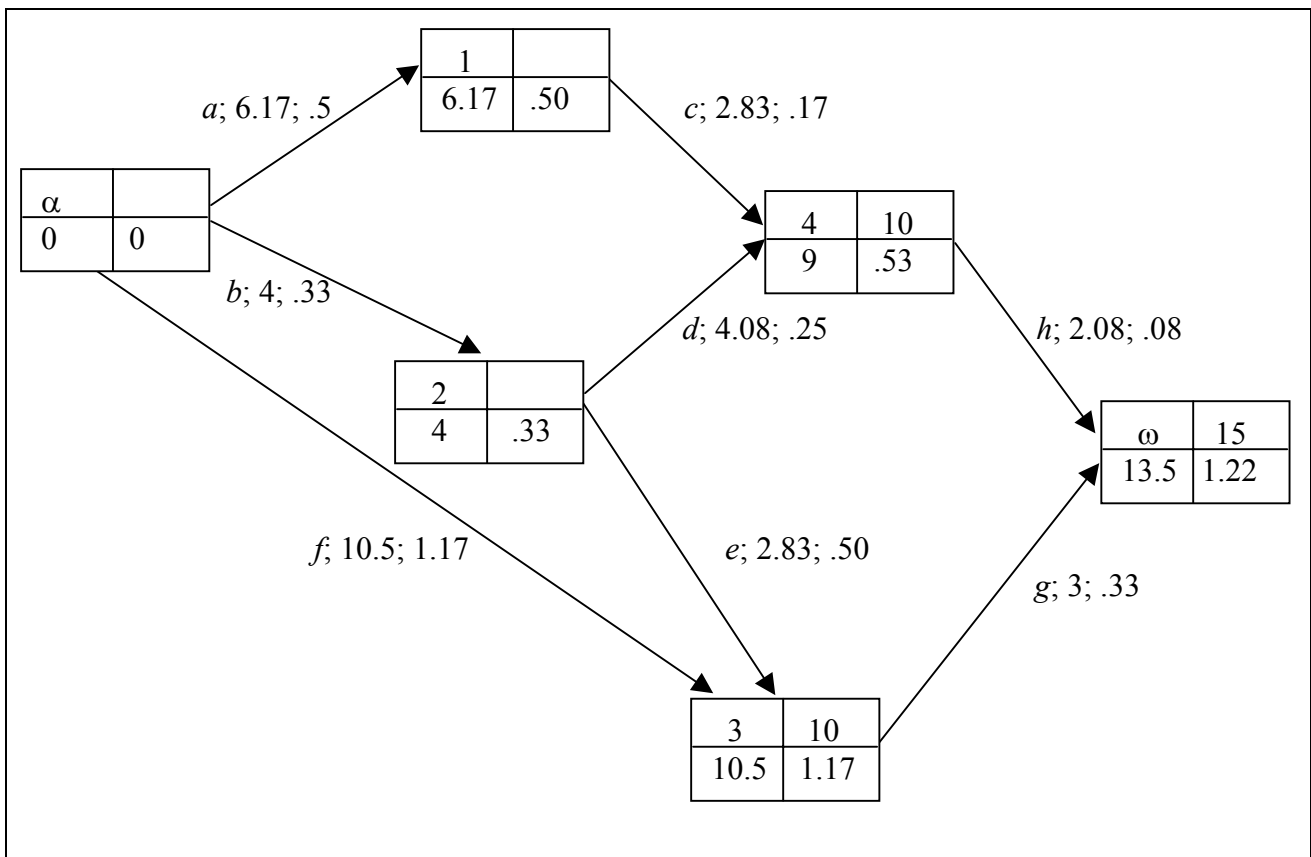


Figura 1.19

#### Algoritmo (*forward*), per calcolare il tempo atteso.

Occorre esplorare il grafo (in avanti) partendo dal nodo iniziale  $\alpha$ ; in  $T_e(\alpha)$  si mette il valore  $t_0$ , cioè il tempo di partenza del progetto; per ogni altro nodo  $x$  che ha tutti i suoi  $k$  predecessori immediati  $y_j$  ( $j = 1, 2, \dots, k$ ) con  $T_e(y_j)$  valutata, si pone  $T_e(x) = \max_j(T_e(y_j) + \text{durata attesa}(y_j, x))$ , dove  $(y_j, x)$  è la freccia tra il nodo  $y_j$  e il nodo  $x$ .

#### Algoritmo (*forward*), per calcolare la deviazione standard.

Per calcolare, come illustrato in Appendice B, la deviazione standard della somma  $c$  di due variabili aleatorie  $a$  e  $b$  si utilizza la regola pitagorica

$$\sigma(c) = \text{sqrt}(\sigma(a)^2 + \sigma(b)^2).$$

Quindi, per calcolare la deviazione standard di un evento (nodo del grafo) che ha un solo predecessore occorre comporre con la regola pitagorica sopra vista la deviazione standard del nodo precedente con quella della freccia che li unisce; nel caso di più predecessori si sceglie il massimo fra questi valori. Quindi, per calcolare la deviazione standard di ogni nodo, occorre esplorare il grafo (in avanti) partendo dal nodo iniziale  $\alpha$ ; in  $\sigma(\alpha)$  si mette il valore 0, cioè la data di partenza del progetto si considera certa; per ogni altro nodo  $x$  che ha tutti i suoi  $k$  predecessori immediati  $y_j$  ( $j = 1, 2, \dots, k$ ) con  $\sigma(y_j)$  valutata, si pone  $\sigma(x) = \max_j(\sqrt{\sigma(y_j)^2 + \sigma(y_j, x)^2})$ , dove  $j$  numerava le frecce in entrata per il nodo  $x$ .

### Algoritmo per calcolare la probabilità dei nodi con target.

Nei grafi tipo CPM il percorso critico serve per individuare le attività da tenere sotto controllo per la loro importanza al fine di assicurare la terminazione del progetto nei tempi prefissati. Nei grafi di tipo Pert non esiste più il concetto di percorso critico, nel senso che tutte le valutazioni sono probabilistiche e quindi lo è anche il valore atteso per il completamento dell'intero progetto (evento  $\omega$ ); l'esigenza di individuare le attività "critiche", cioè da tenere sotto controllo perché determinanti per il rispetto del tempo finale, viene risolta, per ogni evento per cui è dato il *target*, calcolando la probabilità di non rispettarlo.

Per ogni nodo  $x$  cui è stato assegnato un *target*, si calcola il valore  $z(x) = (T(x) - T_e(x)) / \sigma(x)$ , cioè il numero di deviazioni standard che separano il *target* dal valore atteso. Come illustrato nell'Appendice A, ad ogni valore di  $z$  (da  $-\infty$  a  $+\infty$ ) corrisponde un valore di probabilità che in pratica si ottiene per interpolazione (o approssimazione numerica).

Nell'esempio che si sta considerando, sono dati tre *target*:

$target(\omega) = 15$ , vincolo posto dal committente per l'intero progetto;

$target(4) = 10$ , vincolo imposto dalla disponibilità delle risorse per l'attività  $c$ ;

$target(3) = 10$ , vincolo posto dal committente come *milestone* intermedio.

Per i tre nodi si ottiene:

$z(\omega) = 1.89$ , cui corrisponde una probabilità di non rispettare il *target* di circa il 3%;

$z(4) = 1.23$ , cui corrisponde una probabilità di non rispettare il *target* di circa il 11%;

$z(3) = -0.43$ , cui corrisponde una probabilità di non rispettare il *target* di circa il 65%.

### Esercizi

E18. Sia dato un grafo Pert come mostrato in figura 1.20.

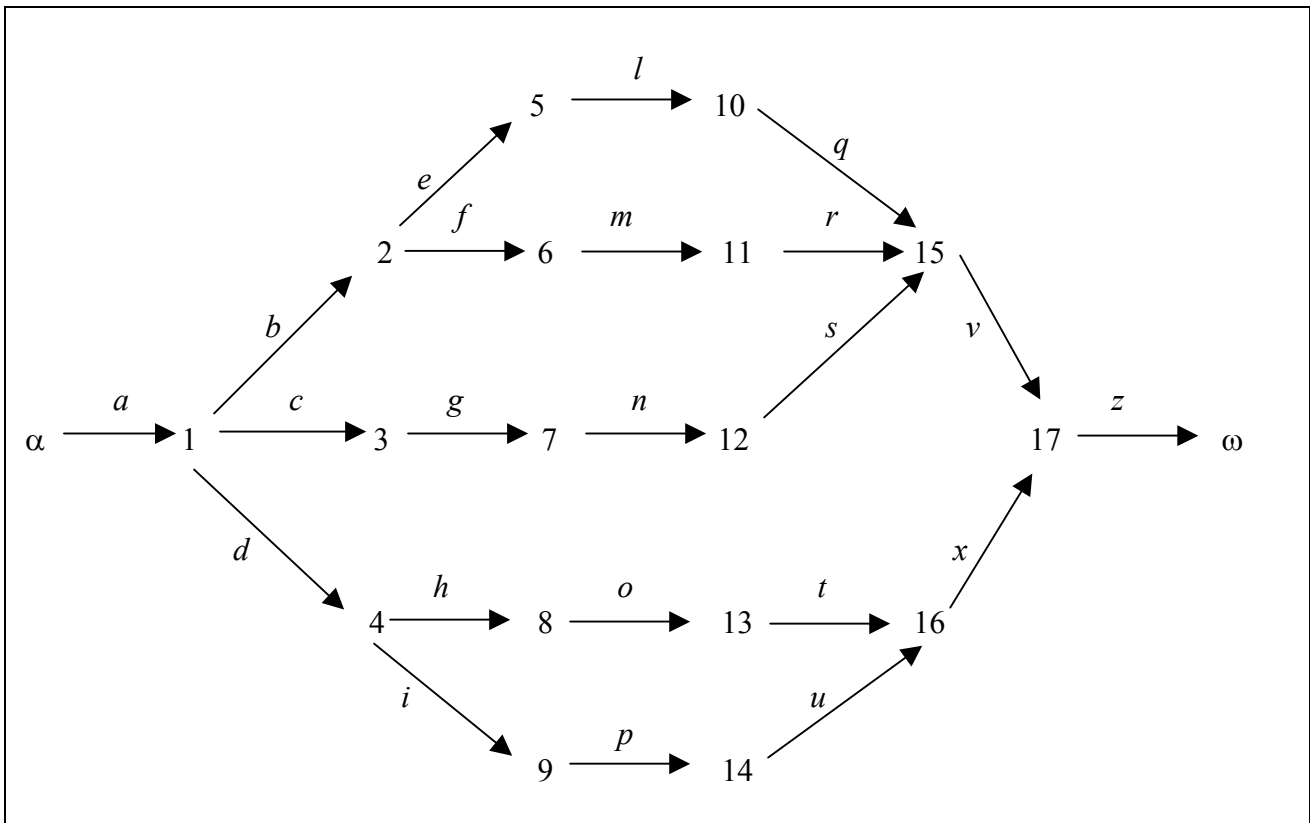


Figura 1.20

Le durate delle attività, assegnate in giorni, sono riportate dalla seguente tabella (quando è indicata una sola durata vuol dire che l'attività ha una durata fissa ed esattamente prevedibile).

attività	$d_o$	$d_m$	$d_p$	attività	$d_o$	$d_m$	$d_p$
a		10		n		10	
b	7	9	13	o	20	22	30
c	6	8	15	p	18	25	30
d	5	6	7	q	3	3	4
e		10		r	3	3	4
f	8	9	10	s		2	
g	15	17	25	t	3	4	5
h	3	3	5	u	2	3	6
i	4	7	8	v		15	
l		5		x		5	
m	6	7	9	z		10	

Si supponga che l'obiettivo sia di finire il progetto in 70 giorni ( $target(\omega) = 70$ ): calcolare la probabilità che ciò avvenga.

## 1.6 La pianificazione come compromesso fra tempo e risorse

In questo paragrafo vengono brevemente discussi due problemi.

1. Dato uno schema di progetto, rappresentato da un Gantt o da un grafo CPM, cioè supponendo assegnate le attività, la struttura di precedenza, la loro durata e le risorse necessarie per portarle a termine (nella durata stabilita), utilizzando gli eventuali *slack*, risistemare la attività nel tempo in modo da diminuire i picchi di impiego delle risorse, supponendo, inoltre, che tutte le risorse siano equivalenti.
2. Dato uno schema di progetto, rappresentato da un Gantt o da un grafo CPM, cioè supponendo assegnate le attività, la struttura di precedenza, la loro durata e le risorse necessarie per portarle a termine (nella durata stabilita), riassegnare le risorse alle attività in modo che il numero di risorse impegnate in ogni momento del progetto non superi un valore assegnato: naturalmente può variare la durata totale del progetto.

**Problema 1.** Nella soluzione del problema 1 è di fondamentale importanza riferirsi al cammino critico, quindi conviene partire dal CPM e dalla determinazione del cammino critico. La prima cosa da costruire è un Gantt la cui prima attività è il cammino critico, che è una catena connessa di attività che nel complesso iniziano al tempo  $t(\alpha)$  e terminano al tempo  $t(\omega)$ ; le altre righe del Gantt portano le altre attività singolarmente, riportando per ciascuna l'intervallo completo di variabilità (cioè la sua durata preceduta dallo *slack*). Corrispondentemente, nell'intervallo di tempo  $[t(\alpha), t(\omega)]$  si costruisce l'istogramma del numero di risorse utilizzate. La soluzione del problema si ottiene euristicamente per tentativi, spostando la durata delle attività che non sono sul cammino critico nel loro intervallo di variabilità in modo che il corrispondente istogramma abbia le proprietà richieste (sia tutto al di sotto del livello corrispondente al numero massimo di risorse prefissato, senza naturalmente cambiare le risorse assegnate alle singole attività).

Si consideri per esempio lo schema di progetto descritto dal grafo CPM descritto dalla seguente tabella.

Nodo di inizio	Nodo di fine	Durata dell'attività	Risorse assegnate
$\alpha$	B	4	3
$\alpha$	D	3	4
B	C	5	2
B	D	5	2
B	$\omega$	8	3
D	$\omega$	7	3
C	$\omega$	6	2

Si verifica facilmente che il cammino critico è dato dalla seguente catena:  $(\alpha, B, D, \omega)$ . Si verifica inoltre che gli *slack* sono dati dalla seguente tabella

Nodo di inizio	Nodo di fine	Slack totale	Slack libero
$\alpha$	D	6	6
B	C	1	0
B	$\omega$	4	4
C	$\omega$	1	1

Il Gantt corrispondente (costruito come descritto sopra), in cui tutte le attività sono allocate il più tardi possibile, è mostrato in figura 1.21.

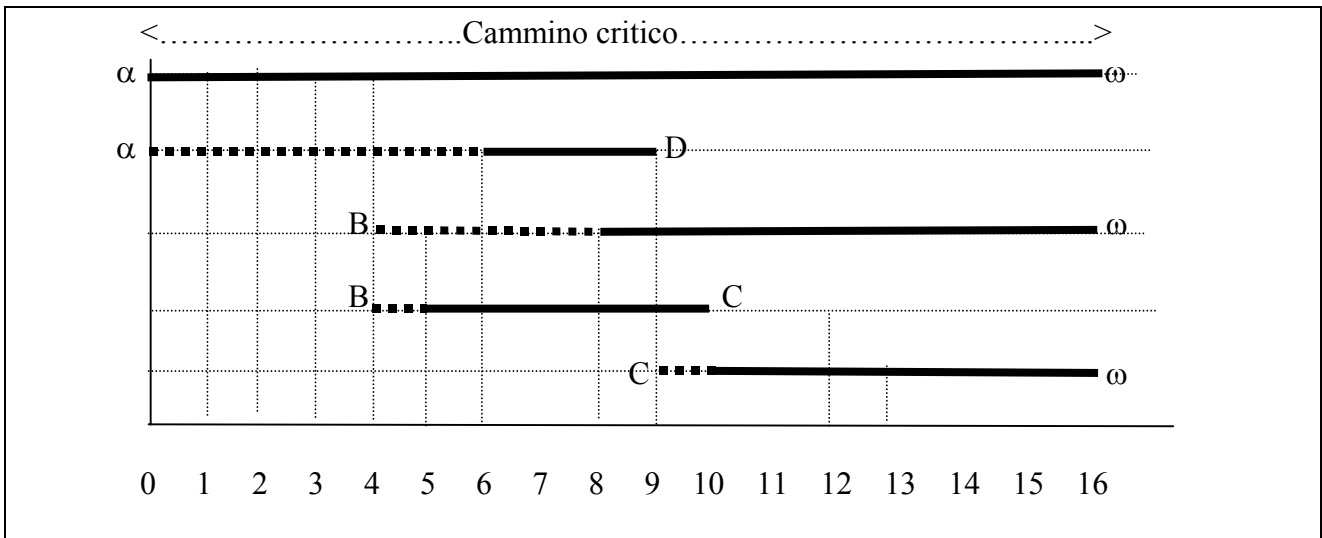


Figura 1.21

Costruendo l'istogramma sopra definito si nota un picco di 11 risorse tra l'ottava e la nona settimana nella quale si sovrappongono 4 attività (B-D con 2,  $\alpha$ -D con 4, B- $\omega$  con 3 e B-C con 2).

Una diversa distribuzione temporale delle attività, in cui ognuna di esse viene allocata prima possibile, come mostrato in figura 1.22, porta ad un picco di 8 risorse nell'intervallo fra la nona e la dodicesima settimana (D- $\omega$  con 3, B- $\omega$  con 3 e C- $\omega$  con 2).

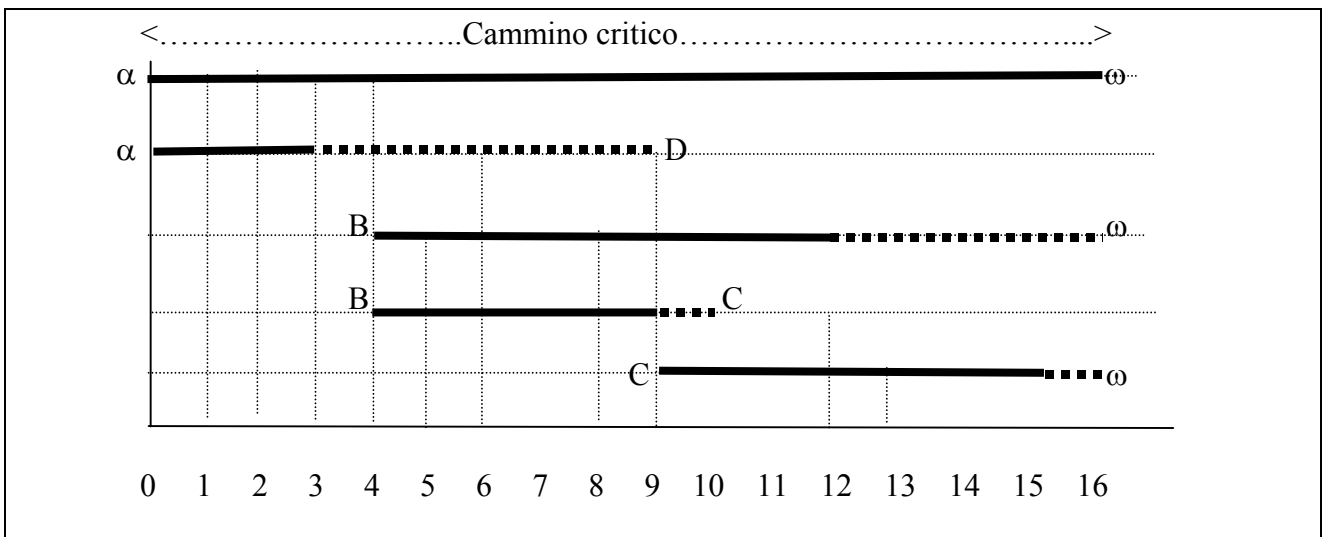


Figura 1.22

Nei casi semplici, la soluzione del problema di ridurre i picchi di impegno delle risorse è facilmente ottenibile per tentativi; nei casi più complessi, si usano le tecniche di programmazione lineari.

**Problema 2.** Nel problema precedente si supponevano assegnate (e non variabili) le risorse per ciascuna attività, si supponeva che il progetto dovesse terminare prima possibile e si poteva solo spostare l'inizio delle singole attività (all'interno dello *slack*); l'obiettivo era quello di rendere uniforme l'impiego delle risorse (abbassando i picchi) in modo da ottenere la migliore efficienza (tutte le risorse disponibili sono il più possibile al lavoro e quindi si migliora la loro produttività). Per ottenere una produttività ottimale, occorre rimuovere il vincolo della assegnazione a priori di risorse alle attività.

Il problema in esame aggiunge questa ulteriore variabilità: è fissato l'impegno delle singole attività, ma non è imposto il numero delle risorse ad esse assegnato (e quindi può diventare variabile la loro durata). La soluzione del problema nei casi semplici può ancora essere risolta per tentativi. Si consideri ancora lo schema di progetto visto sopra in cui per ogni attività è assegnato l'impegno come mostrato nella seguente tabella.

Nodo di inizio	Nodo di fine	Impegno dell'attività in settimane uomo
$\alpha$	B	12
$\alpha$	D	12
B	C	10
B	D	10
B	$\omega$	24
D	$\omega$	21
C	$\omega$	12

Il problema si può impostare nel modo seguente. Avendo  $K$  risorse interscambiabili (per esempio 3), trovare la pianificazione a cui corrisponde il tempo totale minimo.

Per esempio, con  $K = 1$ , la durata minima del progetto è determinata, ma rimangono da stimare quanti e quali Gantt la permettono. La soluzione euristica generale dell'esercizio è lasciata al lettore.

## Appendice: formule utili da ricordare

1. Integrale di Gauss:

$$\int_{-\infty}^{+\infty} e^{-x^2} dx = \sqrt{\pi} ; \quad \int_{-\infty}^{+\infty} e^{-ax^2} dx = \sqrt{a\pi}$$

2. Distribuzione normale:

$$P(x) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{(x-\bar{x})^2}{2\sigma^2}}$$

$$\int_{-\infty}^{+\infty} P(x) dx \equiv 1, \quad \int_{-\infty}^{+\infty} P(x)x dx \equiv \bar{x}, \quad \int_{-\infty}^{+\infty} P(x)(x-\bar{x})^2 dx = \sigma^2$$

3. Funzione integrale standard

$$F(z) = \int_{-\infty}^z P(x) dx \quad \text{con } P(x) = \frac{1}{\sqrt{2\pi}} e^{-\frac{x^2}{2}}$$

$$\lim_{z \rightarrow \infty} F(z) = 1; \quad \lim_{z \rightarrow -\infty} F(z) = 0;$$

## **Conclusioni**

Al termine di questo capitolo il lettore dovrebbe essere in grado di discutere i seguenti argomenti:

- attività, eventi, WBS e schemi di progetto;
- fasi e scomposizioni successive per ottenere il WBS;
- macroattività;
- risorse e attributi;
- vincoli sulle attività
- assegnazione delle risorse alle attività;
- allocazione delle attività nel tempo; algoritmi elementari (a risorse infinite, ad una attività alla volta);
- diagrammi di Gantt e uso di un prodotto commerciale per disegnarli;
- durata e costo totali dei progetti;
- flessibilità di una attività e percorso critico;
- verifica e controllo;
- metodo della riserva;
- analisi dei rischi: aree, valutazione, azioni correttive;
- grafi CPM; algoritmo di conversione dai diagrammi di Gantt e algoritmo di calcolo dei nodi;
- grafi Pert e algoritmo di calcolo dei nodi.