Análise de Risco

Sessão 5
Risco de Prazo

Prof. E.A. Schmitz

PPGI/UFRJ

July 9, 2018

Planejamento da execução de um projeto

Estimativa do prazo de execução de um projeto

- Parte essencial do planejamento
- Elementos: atividades, recursos, atividades contingenciadas
- Incertezas
  - duração das atividades
  - quantidade e produtividade dos recursos
  - eventos incertos
- Duração é uma variável aleatória
- Risco de prazo: distribuição de probabilidade da duração

O problema do agendamento

Agendamento *(Scheduling)*: gerar um cronograma especificando quando cada tarefa deverá ser executada

Formalmente: relação binária $C = \{(a_i, t_i)\}$ onde a_i denota a atividade e t_i seu tempo de início

Exemplos de problemas de agendamento ótimo:

- minimizar prazo (min-makespan)
- maximizar retorno (max-NPV)

Problemas deterministas: tarefas tem duração conhecida

Variantes do *min-makespan*

- CPM: minimizar prazo sem restrição de recursos (polinomial)
- RCPS: minimizar prazo com restrição de recursos (NP-hard)

Elementos de um problema de agendamento

Elementos:

- Função objetivo a ser otimizada: p.ex. *max-npv*
- $A = \{a_1, a_2, \ldots a_n\}$: conjunto de atividades de um projeto
- $P = \{(a_i, a_j\}$ onde a_i precede a_j representa a relação de precedência entre as atividades
- P também pode ser representado por um grafo direcionado $G = (E, N)$, onde $E =$elos e $N =$ nós
- δ_n: prazo máximo para realização do projeto

Características das atividades:

- d_i: duração de a_i
- s_i: tempo de início de a_i
- r_i: recursos necessários para sua execução
- c_i: valor do fluxo de caixa descontado ao final da atividade a_i

Exemplo de projeto

Dados do projeto

- n=14
- $\delta_n = 44$
- vetor de durações d=[0,6,5,3,1,6,2,1,4,3,2,3,5,0]
- vetor de fluxo de caixa c=[0,-140,318,312,-329,153,193,361,24,33,387,-386,171,0]
- relação de precedencia P={(1,2),(1,3),(1,4),(2,9),(3,5),(3,6),(3,7),(4,8),(5,10), (6,12),(7,8),(7,11),(8,13),(9,14),(10,12),(11,12),(12,13),(13,14)}

Representação gráfica de um projeto

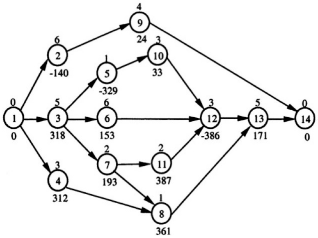

Figure: Exemplo de grafo de projeto

min-makespan 1/2

P: minimizar o prazo (*min-makespan*)

- *minimizar* s_n
- sujeito às restrições:
  - $\forall (a_i, a_j) \in E: \; s_j \geq s_i + d_i$
  - $s_1 = 0$

Algoritmo - passo de ida

Data: d: duração das atividades, succ: lista de sucessores

Result: est: tempo mais cedo de inicio das atividades

$cpmf \leftarrow function(s)$;

$eft[s] \leftarrow est[s] + d[s]$;

for $i \in succ[s]$ **do**

  if $est[i] < eft[s]$ **then**

    $eft[s] \leftarrow eft[s]$;

    $cpmf(i)$;

  end

end

Algorithm 1: CPMf

min-makespan 2/2

Podemos também determinar o tempo mais tarde de inicio (lst) das atividades

Folga: diferença entre o tempo mais tarde (lft) e o tempo mais cedo (est)

Algoritmo passo de volta

Data: d: duração das atividades, pred: lista de predecessores

Result: lft: tempo mais tarde de inicio das atividades

$cpmb \leftarrow function(s)$;

$lst[s] \leftarrow lft[s] - d[s]$;

for $i \in pred[s]$ **do**

    if $lft[i] > lst[s]$ **then**

        $lft[i] \leftarrow lst[s]$;

        $cpmb(i)$;

    end

end

Algorithm 2: CPMb

Exemplo do algoritmo CPM

- Nós visitados: 1 2 5 8 9 7 9 3 8 9 4 6 8 9
- Tempo mais cedo de inicio (est):[0000242911]
- Tempo mais tarde de inicio (lft): [0000242911]
- Folgas: [043040500]
- Atividades críticas: $\{1, 4, 6, 8, 9\}$ aquelas que tem folga zero
- Caminho crítico: $1 \rightarrow 4 \rightarrow 6 \rightarrow 8 \rightarrow 9$ caminho que determina a duração mínima do projeto
- min-makespan=11

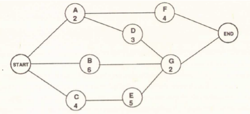

Figure: Exemplo

min-makespan não deterministico

- Assumindo que: d_i: é uma variável aleatóra que representa a duração de a_i
- Problema: avaliar a distribuição de probabilidade do prazo mínimo de execução de um projeto
- Caminho crítico é uma variável aleatória

Soluções aproximativas

- Solução por Monte Carlo
  - repita N vezes
    - gerar uma amostra das durações das atividades
    - avaliar o prazo mínimo
- Solução PERT
  - avaliar o caminho crítica determinista assumindo a média como duração
  - aplicar o TCL para as atividades do caminho crítico

Exercício

Assuma que as durações do exemplo 1 seguem uma distribuição de probabilidade triangular, calcule uma aproximação PERT para a duração do projeto. Estime a probabilidade da duração do projeto ficar entre 9 e 11.5 unidades de tempo.

Ativ	Min	Mprov	Max
A	1	2	4
B	5	6	7
C	2	4	5
D	1	3	4
E	4	5	7
F	3	4	5
G	1	2	3

min-makespan RCPS

P: minimizar o prazo *min-makespan* com restrição de recursos
RCPS: *Resource constrained project scheduling*

- a_k total de recursos do tipo k
- r_{ik} total de recursos do tipo k consumidos pela atividade i
- S_t: activities in execution at time t
- $minimiza\, s_n$
- sujeito às restrições:
  - $\forall (a_i, a_j) \in E : \; s_j \geq s_i + d_i$
  - $s_1 = 0$
  - $\sum_{i \in S_t} r_{ik} \leq a_k \; k \in 1 \cdots m, t \in 1 \cdots f_n$

Algoritmo min-makespan RCPS

Algoritmo construtivo

1. Inicialização:
   $E \leftarrow \emptyset, F \leftarrow \emptyset, T \leftarrow 0, R_k \leftarrow Tot_k$
2. Avaliar atividades disponíveis:
   $D \leftarrow \{j \in A | P(j) = \emptyset\}$
   if $(D = \emptyset) \Rightarrow$ goto 8
3. Avaliar atividades bloqueadas
   $B \leftarrow (A–D)$
4. Colocar atividades em processamento
   $E \leftarrow E \cup \{j \in D | R_k - r_{i,k} \geq 0, \forall k\}; R_k \leftarrow (Rk–r_{i,k}), \forall k$
5. Processa a atividade que está na frente da lista das atividades em processamento
   Pegar próximo $j \in E$
   $T \leftarrow t_j$
   $R_k \leftarrow (R_k + r_{i,k}), \forall k$
   $F \leftarrow F \cup \{j\}$
6. Libera para processamento atividades cujos predecssoras já foram processadas
   $D \leftarrow D \cup \{j \in B | \forall i \in P(j) \Rightarrow i \in F\}$
7. **if** $(E = \emptyset \vee D = \emptyset) \Rightarrow$ goto 8 else 4
8. Todas atividades foram escalonadas

Maximizar o valor de um projeto:max-npv

P: maximizar o *npv* de um projeto:

- $$\max \sum_{i=1}^{n} c_i \times e^{-r\cdot(s_i+d_i)} \tag{1}$$
- sujeito às restrições:
  - $\forall (a_i, a_j) \in E: \; s_j \geq s_i + d_i$
  - $s_1 = 0$
  - $S_n \leq \delta_n$

P : problema de otimização não-linear

- P é um caso particular do problema *start-time dependent cost*
- P pode ser transformado no problema de *min-cut* (polinomial)
- ver (Demeulemeester, 2002) p.141
- no caso de todos c_i positivos: *max-npv* = *min-makespan*

O problema max-npv equivalente P'

Problema linear equivalente P':

- $$\max \sum_{i=1}^{n} c_i \times y_i \tag{2}$$
- sujeito às restrições:
  - $y_i \cdot K_{ij} \leq y_i \quad \forall (a_i, a_j) \in A$:
  - $y_i = 1$
  - $y_i \geq 0 \quad \forall i \in \{2..n\}$

Onde:

- $y_i = e^{-r \cdot (s_i + d_i)}$
- $K_{ij} = e^{r \cdot (l_{ij} - d_i + d_j)}$

Mas:

- P' tem a mesma estrutura do *weighted distribution problem*
- apresentado em Dantzig (1963)