Logistics

Lecture notes*

Maria Grazia Scutellà

Dipartimento di Informatica
Università di Pisa

September 2015

*These notes are related to the course of Logistics held by the author at the University of Pisa. They are based on some books from the literature, as acknowledged at the end of each chapter.

Chapter 9

Project management

The management of a project involves tactical and operational decisions. Some techniques were developed during the 1950s to help managers to plan, organize and control projects, such as the *Critical Path Method (CPM)* and the *Program Evaluation and Review Technique (PERT)*. The focus of these techniques is to determine when a project, i.e. a set of activities that require different amounts of time and must be accomplished by respecting certain precedence relationships, should be completed, and to schedule each activity in order to keep the overall project on schedule.

The main difference between the two techniques is that CPM assumes that the time required by each project activity is known, whereas PERT assumes that such a time is essentially a random variable, so estimating the probability of completing the project by a given deadline.

An example: Lightner Construction

Lightner Construction is a general contracting company specialized in the construction of family residences and small office buildings (so, this is an example of internal logistics). Let us consider the project described in Table 9.1, characterized by activities with the associated required relationships among them.

The project activities are usually described by means of an auxiliary directed graph, called *AON (Activity-On-Node) network*. The AON network of the Lightner Construction example is depicted in Figure 9.1, where A is the unique starting activity, and M is the unique final activity. If this was not the case, without loss of generality we could add two fictitious starting and final activities, as in the example network of Figure 9.2.

9.1 The Critical Path Method

The goal of the Critical Path Method (CPM) is to determine the minimum completion time of the overall project. The main observation underlying this technique is that such

activity (i)	time required in days (t_i)	immediate predecessors
A (excavate)	3	–
B (lay foundation)	4	A
C	3	B
D	10	B
E	8	D
F	4	D
G	6	D
H	8	C, E, F, G
I	5	H
J	5	H
K	4	I
L	2	J
M (install activity)	4	K, L

Table 9.1: The Lightner Construction's project

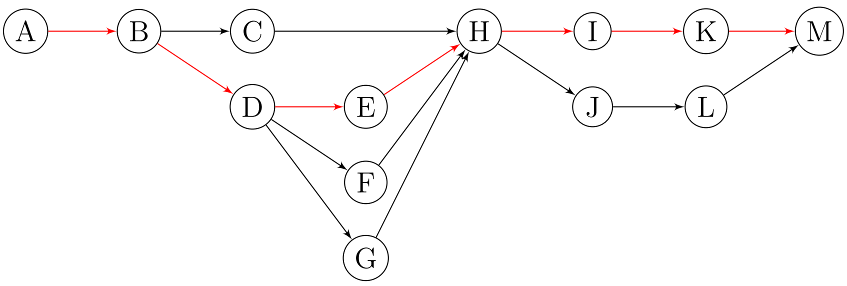

Figure 9.1: AON network of the Lightner Construction's project

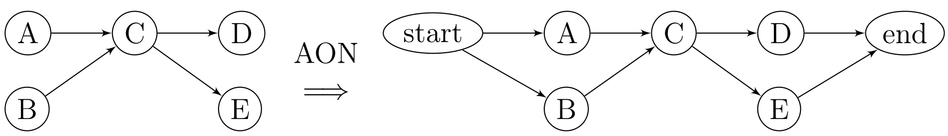

Figure 9.2: Adding fictitious starting and final activities

a minimum completion time is the time required to perform all the activities along the longest path in the AON network (from the unique starting node to the unique final node): this is called a *critical path*, and the corresponding activities are called *critical activities* (highlighted in red in Figure 9.1).

Therefore, CPM first determines the minimum completion time of the overall project, that is a critical path in AON, by assuming to start the project at time 0, and then it schedules each project activity (i.e. it determines its starting and ending time) so as to match the minimum completion time it previously found.

In detail, the CPM algorithm performs the following steps:

1. *Forward pass* through AON: determine the earliest possible start time and the earliest possible finish time for each activity (and so, for the overall project).
2. *Backward pass* through AON: determine the latest possible start time and the latest possible finish time that each activity may have without delaying the project.

   After, the critical path and the critical activities can be determined the critical path and the critical activities by using the computed information. This is important to discover for the project manager, since any delay on these activities will delay the project.

In detail, in the Forward pass CPM computes the following values, by using the time t_i required to perform activity i, for each i:

- EST_i: the earliest possible start time for activity i,
- EFT_i: the earliest possible finish time for activity i,

by exploiting the relation:

$$\mathrm{EFT}_i = \mathrm{EST}_i + t_i, \quad \forall i.$$

Then, in the Backward pass, the following values are computed:

- LST_i: the latest possible start time for activity i,
- LFT_i: the latest possible finish time for activity i,

by exploiting the relation:

$$\mathrm{LST}_i = \mathrm{LFT}_i - t_i, \quad \forall i.$$

Let us refer now to the Lightner example. The computed values are presented in Figure 9.4, using the graphical notation of Figure 9.3. Note that, for activity H, $\mathrm{EST}_H = 25 = \max\{25, 21, 23\}$, i.e. it is the maximum EFT_i among all the immediate predecessors of H in AON, and $\mathrm{LFT}_H = 33 = \min\{33, 35\}$, i.e. it is the minimum LST_i among all the immediate successors of H in AON.

Note also that $\mathrm{LFT}_M = \mathrm{EFT}_M = 46$: this is always true for the final project activity, and 46 is the minimum completion time of the overall project.

In order to discover the critical activities, observe that an activity i is critical if and only if $\text{EST}_i = \text{LST}_i$ (or, equivalently, $\text{EFT}_i = \text{LFT}_i$). Therefore, (A, B, D, E, H, I, K, M) is a critical path for our example (it is the longest path in the AON network).

Those activities that are not critical are characterized by a *slack*, i.e. a possible (maximum) delay of the earliest starting time without causing the delay of the overall project. For instance, in the Lightner Construction example, the project must be completed at time 46, so the slack for activity C is $\text{slack}_C = \text{LST}_C - \text{EST}_C = \text{LFT}_C - \text{EFT}_C = 15$. Note that, e.g. for activity C, we can delay up to slack_C only if each immediate predecessor i of C starts at its EST_i time.

9.1.1 Project crashing

In the Critical Path Method, t_i is the normal time required by activity i, $\forall i$. Often, t_i can be shortened, or *crashed* (e.g., if we can use additional resources).

Let C_i be the amount of time by which activity i is crashed, $\forall i$. We can then state the following *Project crashing problem*: determine the earliest possible completion time of the project, if each activity i can be crashed by $C_i \leq U_i$ (a given upper bound on C_i), $\forall i$.

Example

Let us resume the Lightner example and build a project crashing problem upon it, by introducing the new input data $\{U_i\}$, presented in Table 9.2. The decision variables associated with the new problem are:

- C_i: the time reduction for activity i, $\forall i$,
- T_i: the starting time of activity i, $\forall i$.

The Linear Programming model that can be used to formulate the related Project crashing problem is presented in Model 9.1. The optimal Linear Programming solution indicates that the project can be completed in 28 days rather than 46.

In practice, we could be guided by a budget constraint rather than constraints of type $0 \leq C_i \leq U_i$, $\forall i$.

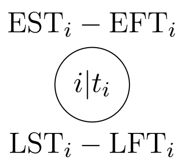

Figure 9.3

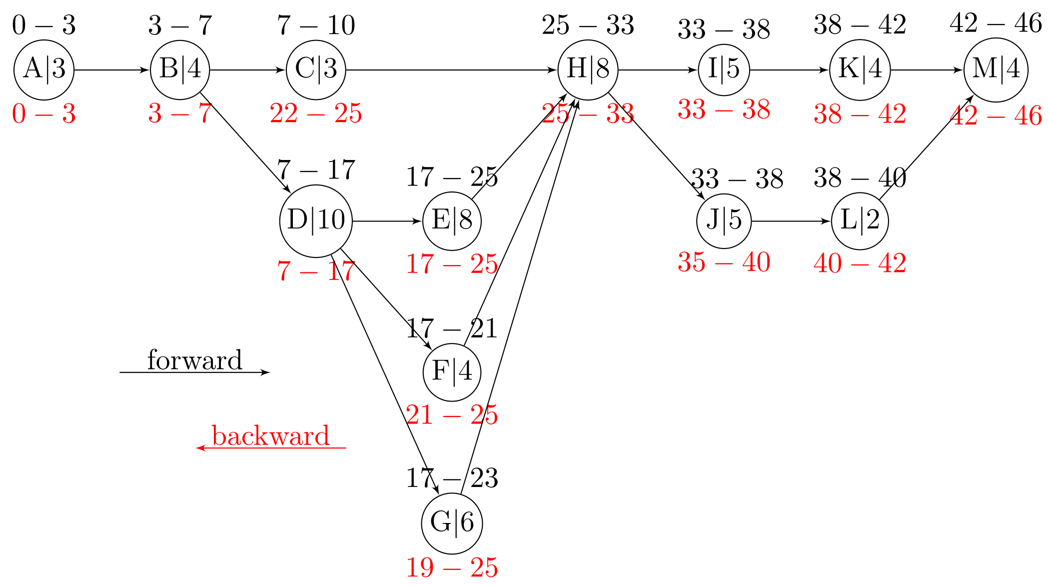

Figure 9.4

$$
\begin{aligned}
&\min\ T_M + (t_M - C_M) \\
&T_B \geq T_A + (t_A - C_A) \\
&T_C \geq T_B + (t_B - C_B) \\
&T_D \geq T_B + (t_B - C_B) \\
&\qquad\vdots \\
&\text{(one constraint for each arc in AON)} \\
&\qquad\vdots \\
&T_M \geq T_K + (t_K - C_K) \\
&T_M \geq T_L + (t_L - C_L) \\
&T_i \geq 0, \quad \forall i \\
&0 \leq C_A \leq 1 \\
&0 \leq C_B \leq 1 \\
&\qquad\vdots \\
&\text{(one constraint for each activity)}
\end{aligned}
$$

Model 9.1: The project crashing problem of Lightner Construction

activity (i)	U_i **(in days)**
A	1
B	1
C	1
D	4
E	3
F	1
G	2
H	3
I	2
J	3
K	2
L	1
M	2

Table 9.2

9.2 The PERT method

The PERT method works by assuming that the time required by each project activity is uncertain (i.e. a random variable). In this regard, three estimates are required for each activity i:

1. a_i: the estimate of the duration of i in the most favorable case;
2. b_i: the estimate of the duration of i in the least favorable case;
3. m_i: the estimate of the most likely duration of i.

PERT uses these estimates as follows:

- $t_i = \dfrac{a_i + 4m_i + b_i}{6}$ is considered by PERT as the (mean) expected duration of i;
- $v_i = \dfrac{(b_i - a_i)^2}{36}$ is considered by PERT as the estimated variance of the duration of i.

The formulas above are based on the assumption that the activity durations are random variables following the *beta probability distribution* (see next).

Given an AON network, PERT determines the critical path in AON, i.e. the path with the longest expected completion time, using the values $\{t_i\}$. Precisely, for each path P in AON, the expected completion time of P is the sum of the t_i of the activities i along P. By assuming that the single activity durations are independent random variables, the variance of the completion time of P is the sum of the v_i of the activities i along P.

Once the critical path P^* has been found, PERT uses the expected completion time

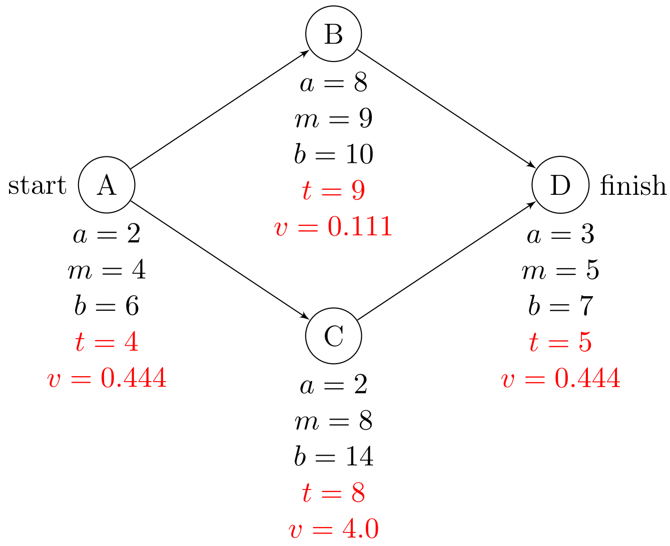

Figure 9.5: Example AON network

and the variance for P^* to estimate the probability of completing the project by various dates, for example to set deadlines for the project completion.

Example

Consider the AON network in Figure 9.5, where there are only two paths, with expected completion time:

(ABD) $\quad 4 + 9 + 5 = 18;$

(ACD) $\quad 4 + 8 + 5 = 17.$

Therefore, the critical path for PERT is (ABD), and so the expected completion time of the overall project is 18 days. However, if we also estimate the variance:

(ABD) $\quad 0.444 + 0.111 + 0.444 = 1.0;$

(ACD) $\quad 0.444 + 4.0 + 0.444 = 4.889;$

we can observe that the selected critical path (ABD) has a small variance, hence we could more accurately say that with high probability the project could be completed within $18 + 1 = 19$ days. But the non critical path (ACD) has a much larger variance, so there is a good probability that the project will not be completed within 19 days.

The example shows the first drawback of PERT: it does not take into account paths which are not critical, and this can be a serious concern. An additional drawback of PERT is that it is not realistic to assume that the activity durations are independent random variables, therefore it is not correct to estimate the variance of a path as the sum of the $\{v_i\}$ of all the activities i along the path.

References C. Ragsdale (2004): Chapter 14

Textbooks

Bigi, G., A. Frangioni, G. Gallo, and M.G. Scutellà (2014). *Appunti di Ricerca Operativa.* Italian. URL: http://didawiki.di.unipi.it/doku.php/ingegneria/ricercaoperativa1/start#testi_di_riferimento.

Drezner, Z. and H.W. Hamacher (2004). *Facility Location: Applications and Theory.* Springer series in operations research. Springer. ISBN: 9783540213451.

Ghiani, G., G. Laporte, and R. Musmanno (2004). *Introduction to Logistics Systems Planning and Control.* Wiley Interscience Series in Systems and Optimization. Wiley. ISBN: 9780470091654.

Ragsdale, C.T. (2004). *Spreadsheet Modeling and Decision Analysis: A Practical Introduction to Management Science.* Cengage South-Western. ISBN: 9780324321777.

Toth, P. and D. Vigo (2002). *The Vehicle Routing Problem.* Monographs on Discrete Mathematics and Applications. Society for Industrial and Applied Mathematics. ISBN: 9780898715798.