Conditional Branching with Full Monte

John Owen
Barbecana, Inc.

Introduction

Critical Path Method (CPM) project models, as used in all the mainstream project scheduling solutions including Oracle Primavera, Microsoft Project, Deltek Open Plan and PowerProject, are based on a project plan that describes the work to be performed and the order in which the various components are required to be completed.

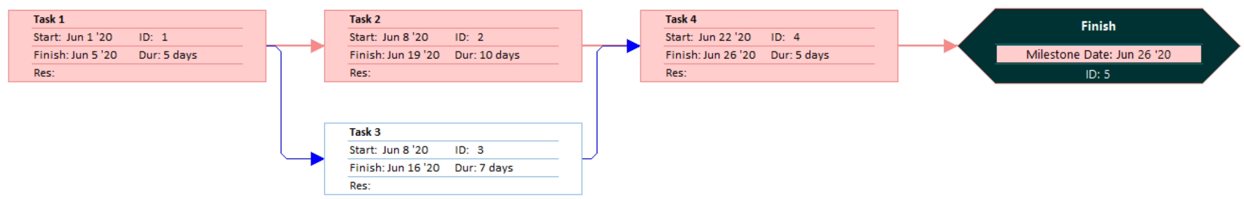

Figure 1

In the example shown in Figure 1, after Task 1 is completed both tasks 2 and 3 can be started but Task 4 cannot start until both tasks 2 and 3 are completed. This gives rise to the concept of a longest path through the project logic that drives the forecast completion date. This is more generally known as the Critical Path and is highlighted in red in Figure 1.

Uncertainty

All estimates for future work are subject to uncertainty. This can take two forms: Estimate Uncertainty and the impact of Random Events.

Estimate Uncertainty can be factored into the model by specifying a range of duration estimates for each task. Statistical modelling techniques like Monte Carlo simulation can then produce a range of forecast finish dates at various levels of confidence.

The impact of random events can be modelled by specifying the chance (probability) of additional time (schedule impact) being required to address the random events (aka Risks), should they occur.

Conditional Branching

But what if the revised model, including duration uncertainty estimates and the impact of random events (risks) suggest a forecast finish date that is unacceptable (too late) at the required level of confidence?

For each threat to the project timeline, be it a high degree of uncertainty in task duration estimates or the impact of a known risk that may potentially delay the project, a risk response should be developed.

Where that response is to mitigate the impact of the uncertainty by producing an alternate sequence of steps to execute that reduce the impact of the delay, then conditional branching can be used.

Consider the following example:

	Task Name	Duration	Predecessors	Risk Assessment	Percent Critical	Finish	Early Finish Percentile (80%)	Chance of Achieving MSP Early Finish
1	**Pre-Risk Model**	**30 days**			**100**	**Jul 10 '20**	**Jul 15 '20**	**9.11**
2	Assembly A	10 days		High Risk	100	Jun 12 '20	Jun 17 '20	9.11
3	Assembly B	9 days		Low Risk	0	Jun 11 '20	Jun 12 '20	50.04
4	Integration Testing	10 days	2,3		100	Jun 26 '20	Jul 1 '20	9.11
5	System Test	10 days	4		100	Jul 10 '20	Jul 15 '20	9.11
6	Launch	0 days	5		100	Jul 10 '20	Jul 15 '20	9.11

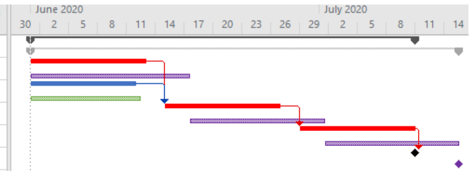

Figure 2

Assembly A and B must both be complete for Integration Testing to commence; however, Assembly A has been categorized as High Risk with a significant chance of delay. After performing a Monte Carlo simulation based on this data, the software is predicting only a 9% chance of completion by July 10, 2020 – the required completion date.

In Figure 2, the Red bars indicate the original plan and the Green/Purple bars show the forecast dates at an 80% level of confidence. A Green color indicates the tasks never impacts the critical path while Purple shows the task is on the critical/longest path in 100% of the simulations.

This is not acceptable, so a risk response analysis is performed for Assembly A. Unfortunately, no way to reduce the uncertainty associated with Assembly A is found so effort is focused on how to achieve the required date even if Assembly A arrives late. An alternate plan is developed as shown in Figure 3.

	Task Name	Duration	Predecessors	Risk Assessment	Percent Critical	Finish	Early Finish Percentile (80%)	Chance of Achieving MSP Early Finish	Percent Active
8	Assembly A	10 days		High Risk	13	Jun 12 '20	Jun 17 '20	9.09	100
9	Additional Unit Test	5 days	8		4	Jun 19 '20	Jun 24 '20	0	91
10	Assembly B	9 days		Low Risk	87	Jun 11 '20	Jun 12 '20	50.3	100
11	Integration Testing	10 days	8,10		96	Jun 26 '20	Jun 26 '20	100	100
12	System Test	10 days	11,9		100	Jul 10 '20	Jul 10 '20	100	100
13	Launch	0 days	12		100	Jul 10 '20	Jul 10 '20	100	100

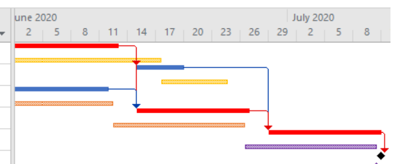

Figure 3

The alternate plan involves accepting that Assembly A may not be available in time for the planned Integration Testing and, if this is the case, Integration Testing will commence with just Assembly B available. Once Assembly A is delivered it will be subject to additional Unit Testing and then incorporated into the final System Test with Assembly B.

Conditional Branching is used to specify the logic to be used if Assembly A is delivered before June 15, 2020 (into Integration Testing) and the alternate, mitigation logic through Additional Unit Test, if delivery is any later.

In Barbecana's Full Monte schedule risk analysis software this is specified under 'Branching'.

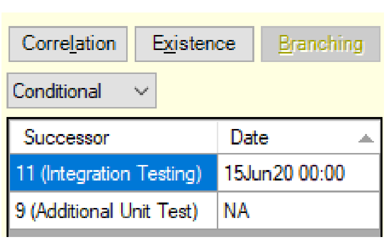

Figure 4

In Figure 3, we can see that the chance of achieving the required completion date is now 100%.

The ‘Percent Active’ column has been added to show how often the simulation invoked the risk mitigation steps. In this case 91% of the simulations required the risk mitigation process to be followed.

The critical path is now also more complex with the Yellow bars indicating tasks were on the Critical Path in between 1% and 49% of the simulations and the Brown Bars indicating tasks were on the critical path in between 50% and 99% of the simulations. This shows us, that following the adoption of the risk mitigation strategy for Assembly A, Assembly B is more often the driving factor for final delivery. It was on the Critical Path 87% of the time.

Summary

Conditional Branching, where the choice of project logic is based on the completion date of a predecessor, can be a useful tool to model risk mitigation strategies and ensure that project objectives are achieved.

It can be used in combination with Probabilistic Branching and Task Existence, which model the chance of additional tasks/work having to be performed, to create project plans with a high degree of realism and improve your chance of project success.