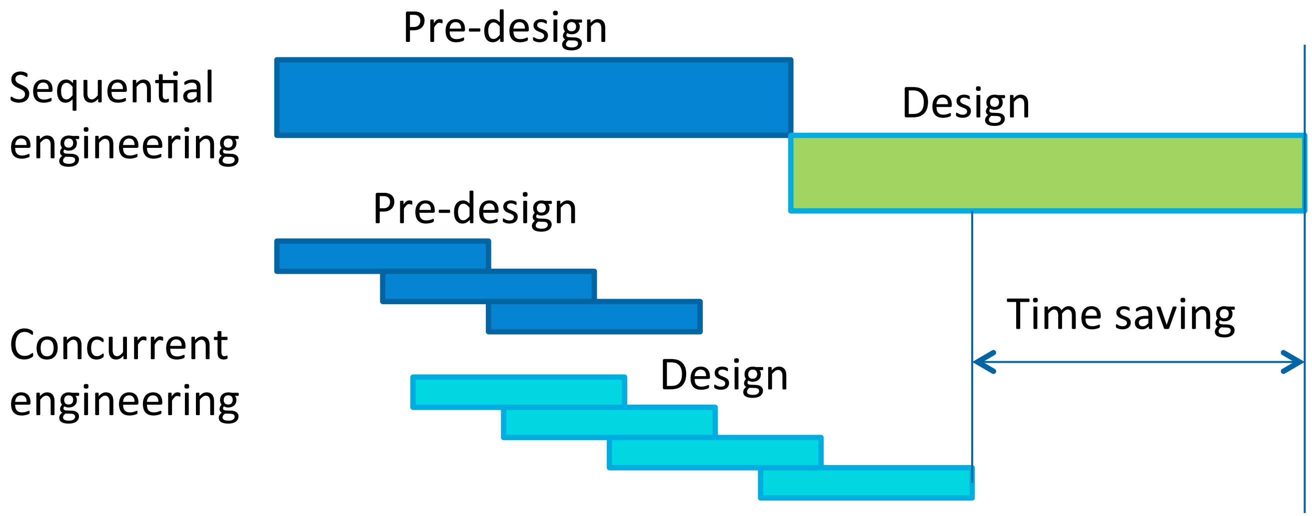
Using the P-factor to Monitor Concurrent Engineering Projects with a Variable Chance of Rework

Ruben De Schrijver
Mario Vanhoucke
Louis-Philippe Kerkhove

Table of contents:

Basics of concurrent engineering

- Concurrent Engineering?

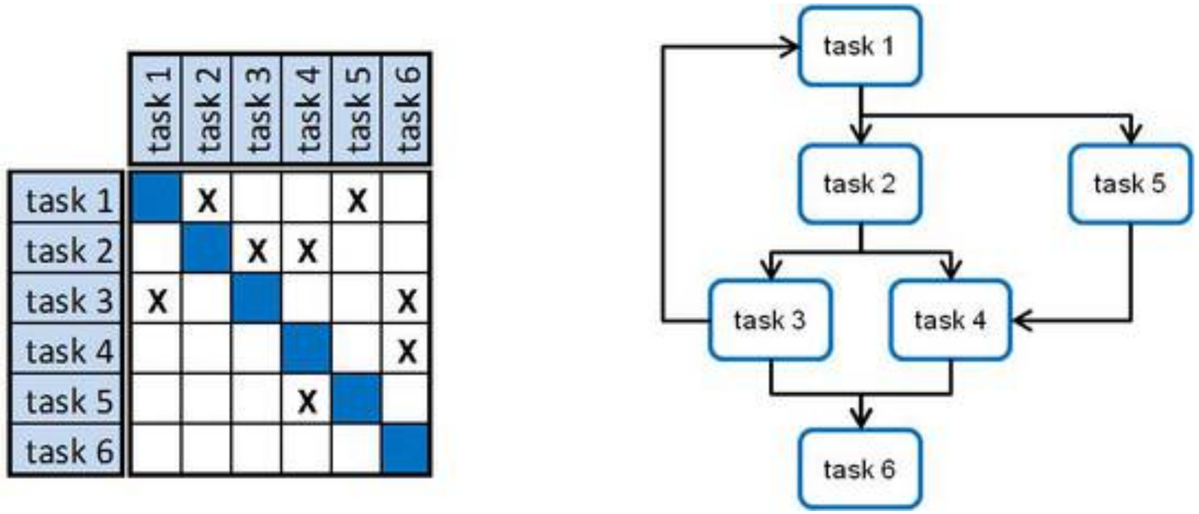


Advantage: time saving

Disadvantage: chance of rework

Basics of concurrent engineering

- Design Structure Matrix (DSM)?



http://www.dsmweb.org/typo3temp/pics/DSM_Tutorial_Basic3_9f5f52512b.jpg

- Binary DSM/Numerical DSM

Existing research

- Chua, D. K. H., & Hossain, M. A. (2011). A simulation model to study the impact of early information on design duration and redesign:

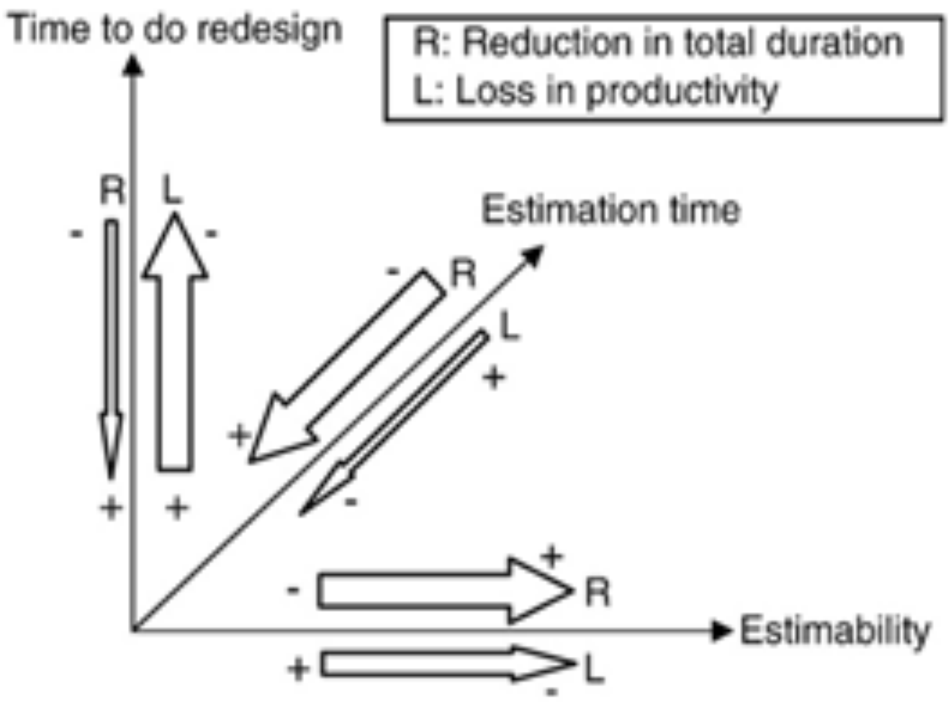


Fig. 7. Influence of three factors on reduction in total duration and loss in productivity.

Goal: Study the impact of these three factors on total duration and productivity

Probability of redesign changes for different values, but constant throughout simulation

Existing research

- Bogus, S., & Diekmann, J. (2011). Simulation of Overlapping Design Activities in Concurrent Engineering:

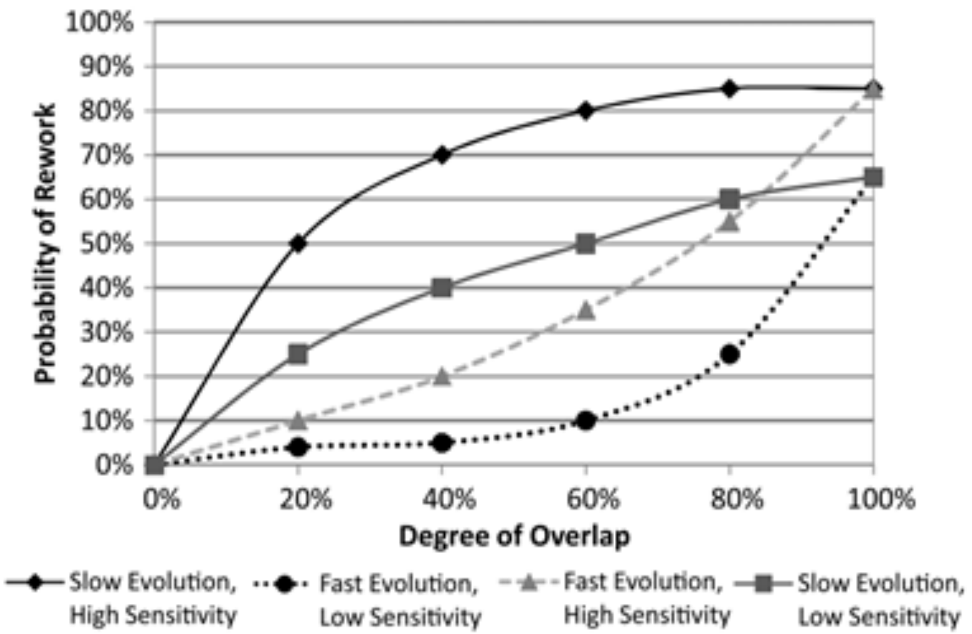


Fig. 3. Theoretical relationship between probability of rework and degree of overlap

Goal: determine optimal

ıt pairs of

evolution and

ʼrobability of rework changes for

sensitivity/evolution, but remains

constant during simulation

Existing research

- Smith, R., & Morrow, J. (2001). Product development process modeling:

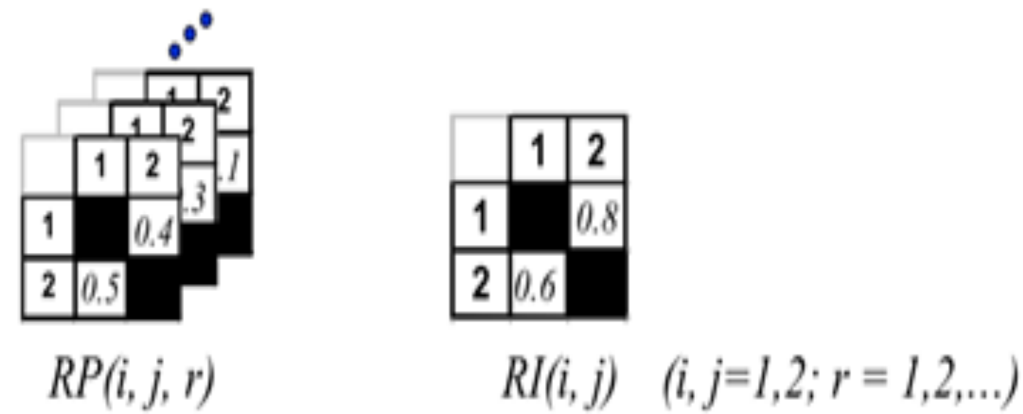


Figure 4. Rework Probability and Impact

Goal: “The model can be used for better project planning and control by identifying leverage points for process improvements and evaluating alternative planning and execution strategies”

Rework probability is fixed

Existing research

<u>Existing simulation models</u>

→ Constant chance of rework during simulation run

→ Not very realistic!

Amount of rework depends on schedule adherence!

Research questions

Is there a way to calculate schedule adherence?

P-factor =
Portion of earned value accrued in congruence with schedule

Formula:

$$p = \frac{\sum_{i \in N} min(\mathrm{PV}_{i,\mathrm{ES}}, \mathrm{EV}_{i,\mathrm{AT}})}{\sum_{i \in N} \mathrm{PV}_{i,\mathrm{ES}}}$$

N	Set of activities in a project
$\mathrm{PV}_{i,\mathrm{ES}}$	Planned Value of activity i at time instance ES
$\mathrm{EV}_{i,\mathrm{AT}}$	Earned Value of activity i at the current time AT

Lies between 0 and 1

Research questions

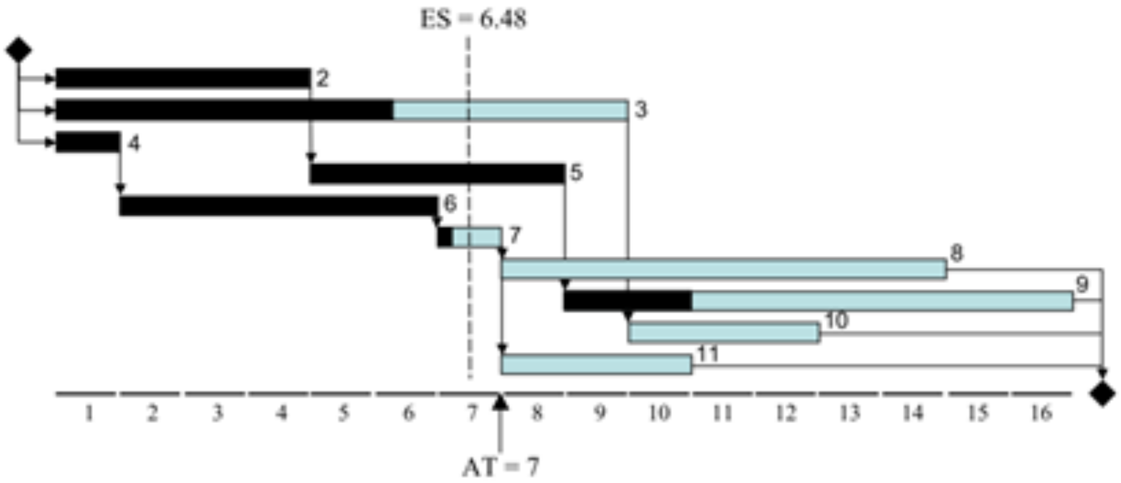


Fig. 2.3 Real life execution of the example project relative to the baseline schedule

Vanhoucke, M. (2009). Measuring time: improving project performance using earned value management (p. 183).

Graphical interpretation: Black bars left of ES-line / total EV

Research questions

Does schedule adherence influence rework?

Decrease of schedule adherence

→ Increased risk of rework

Idea: Use of p-factor to calculate rework

$$R = f(r) \bullet EV(r) = f(r) \bullet (1 - P) \times EV$$

With: $f(r) = 1 - C\text{^}n \bullet e\text{^}(-m \bullet (1 - C))$

C = fraction complete of project (EV/BAC)
e = natural number (base "e")
^ = signifies an exponent follows

Research questions

Problems: No trend information and no forecast total rework possible

Solution: Normalize R to work remaining

Schedule Adherence Index = SAI = R / (BAC – EV)

Increasing SAI = decreasing schedule adherence

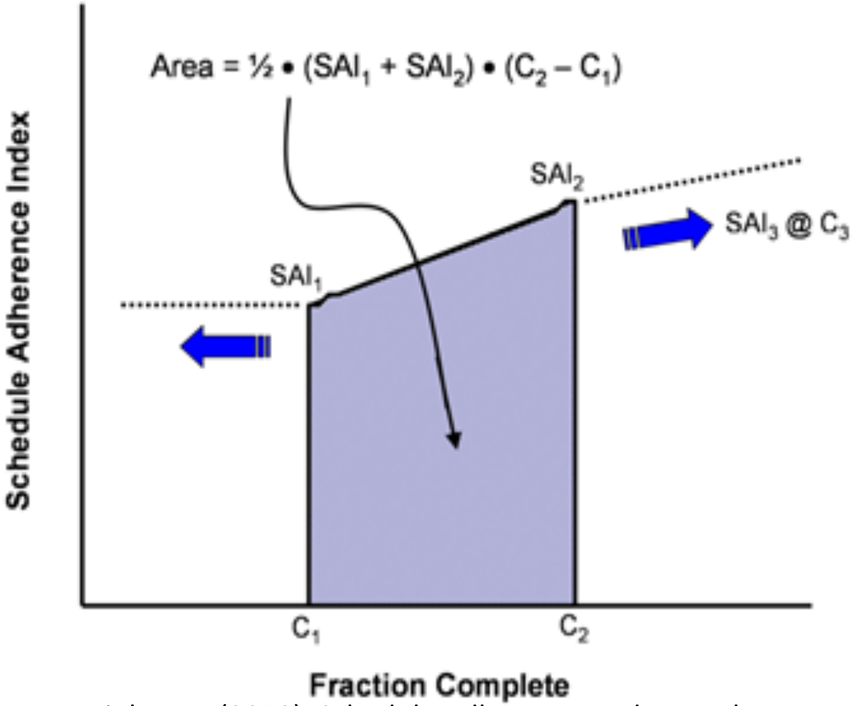
Characteristics:

1. Enables forecasting of total rework
2. Takes cumulative efffects of imperfect schedule adherence into account

Research questions

Forecast of total rework:

Compute rework through trapezoidal approximation technique



Lipke, W. (2010). Schedule Adherence and Rework.

$$R_p(n) = BAC \bullet [\tfrac{1}{2} \bullet (SAI_n + SAI_{n-1}) \bullet (C_n - C_{n-1})]$$

n = the performance period of interest

Area multiplied with BAC to obtain rework cost for any period

$$R_{cum} = \Sigma R_p(n)$$

$$R_{tot} = R_{cum} + SAI \bullet (BAC - EV)$$

Research questions

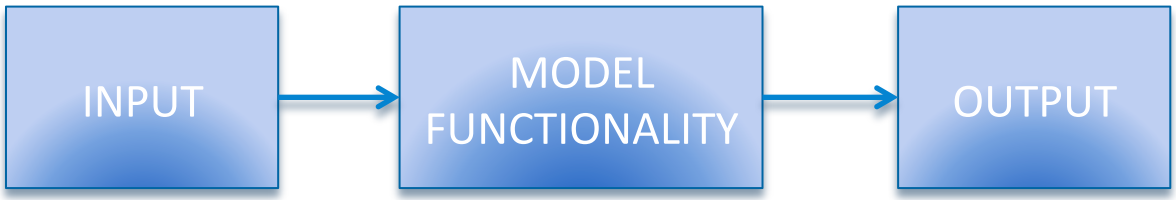
How to determine chance of rework based upon this forecast?

The forecasted rework will be used to compute a chance of rework at each status point of the project

Rtot = **Chance of rework** x rework impact x duration

→ **Chance of rework** = Rtot /(rework impact x duration)

The simulation model



Input levers

- Activity duration
- DSM
- Rework Impact
- Evolution
- Sensitivity
- Risk-factor

Input levers

- Activity duration
- DSM
- Rework Impact
- Evolution
- Sensitivity
- Risk-factor

3 estimates for each acitivity:

1. Worst-case estimate(=P)
2. Most likely estimate(=M)
3. Best-case estimate(=O)

This is also known as the PERT method:

Expected value of activity duration= (O+ 4M + P) / 6

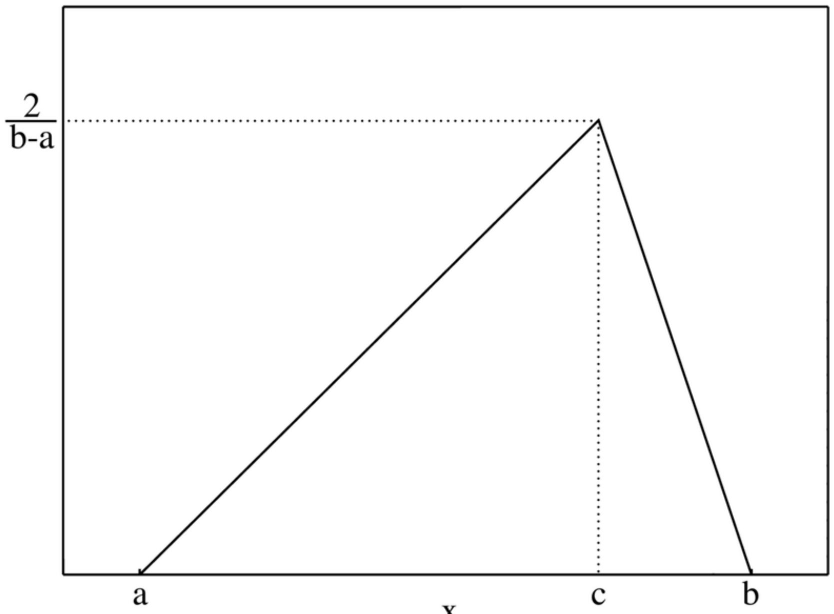
Random values will be selected between worst-case and best-case estimates

Input levers

- Activity duration
- DSM
- Rework Impact
- Evolution
- Sensitivity
- Risk-factor

Assumption: Each activity duration follows a triangular distribution

Probability density function of triangular distribution:



http://upload.wikimedia.org/wikipedia/commons/thumb/4/45/Triangular_distribution_PMF.png/325px-

Input levers

- Activity duration
- **DSM**
- Rework Impact
- Evolution
- Sensitivity
- Risk-factor

Binary DSM to represent information dependency

Used to:

1. Make schedule:
   100% overlapping between independent activities

2. Add rework:
   Rework added only to dependent activities

Input levers

- Activity duration
- DSM
- Rework Impact
- Evolution
- Sensitivity
- Risk-factor

Characteristics:

1. For each activity rework impact will be needed

2. Determines duration of rework when rework occurs

3. Varies from 0% to 100%

Input levers

Activity duration

DSM

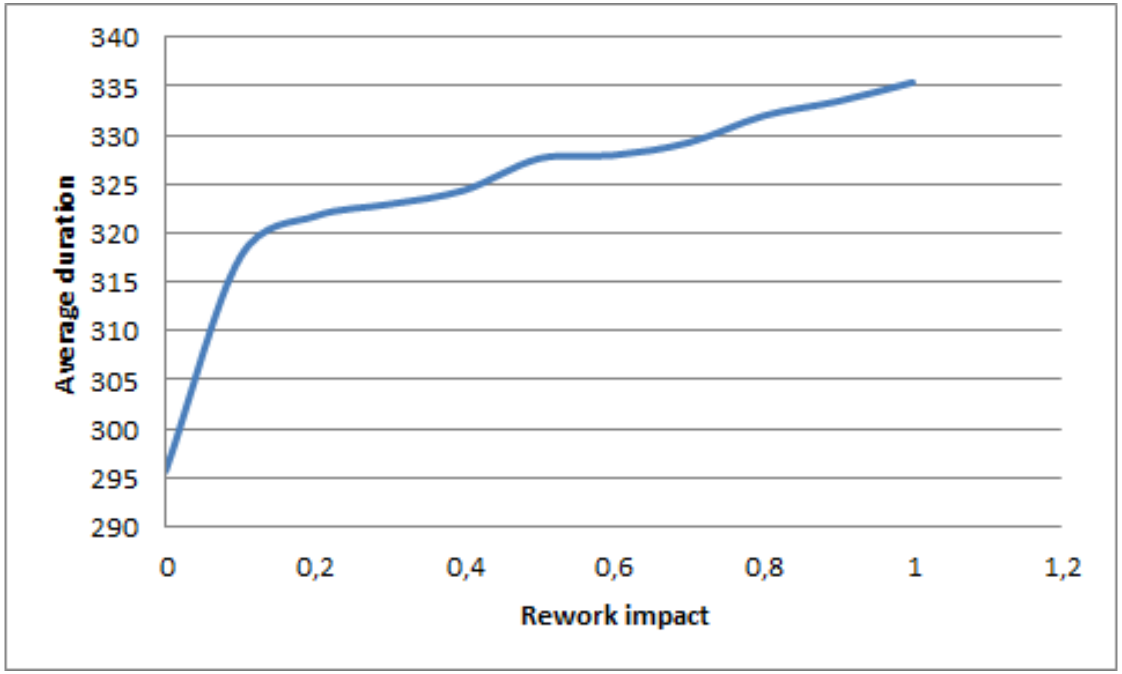
Rework Impact

Evolution

Sensitivity

Risk-factor

Impact on project duration:



Input levers

- Activity duration
- DSM
- Rework Impact
- Evolution
- Sensitivity
- Risk-factor

Development speed of information and knowledge

4 factors of evolution:

1. Design optimisation
2. Constraint satisfaction
3. External information exchange
4. Standardisation

General rule: More iterations in design is slower evolution

Input levers

- Activity duration
- DSM
- Rework Impact
- Evolution
- Sensitivity
- Risk-factor

Range of values:

0 = Slow evolution

1 = High evolution

Influence on model:

Changing evolution will lead to a change in overlapping degree

0	10% - 20% overlap
1	40% - 60% overlap

Input levers

- Activity duration
- DSM
- Rework Impact
- Evolution
- **Sensitivity**
- Risk-factor

Sensitivity of downstream activity to change of information coming from an upstream activity

Factors of sensitivity:

1. Constraint sensitive
2. Input sensitive
3. Integration sensitive

Value range:

0 = Low sensitivity

1 = High sensitivity

Input levers

- Activity duration
- DSM
- Rework Impact
- Evolution
- **Sensitivity**
- Risk-factor

Influence on model:

Changing sensitivity will also lead to a change in overlapping degree

0 → 20% - 60% overlap

1 → 10% - 40% overlap

Input levers

- Activity duration
- DSM
- Rework Impact
- Evolution
- Sensitivity
- **Risk-factor**

Risk-factor: value of 0, 1 or 2

Determines degree of overlap in accordance with figure 3 of Bogus, S., & Diekmann, J. (2011). Simulation of Overlapping Design Activities in Concurrent Engineering.

Values for overlapping degree (for different combinations of evolution/sensitivity):

0 → 0% overlap
1 → 10% (0,1); 20% (0,0); 40% (1,1); 60% (1,0)
2 → 20% (0,1); 40% (0,0); 60% (1,1); 80% (1,0)

Input levers

Activity duration

DSM

Rework Impact

Evolution

Sensitivity

Risk-factor

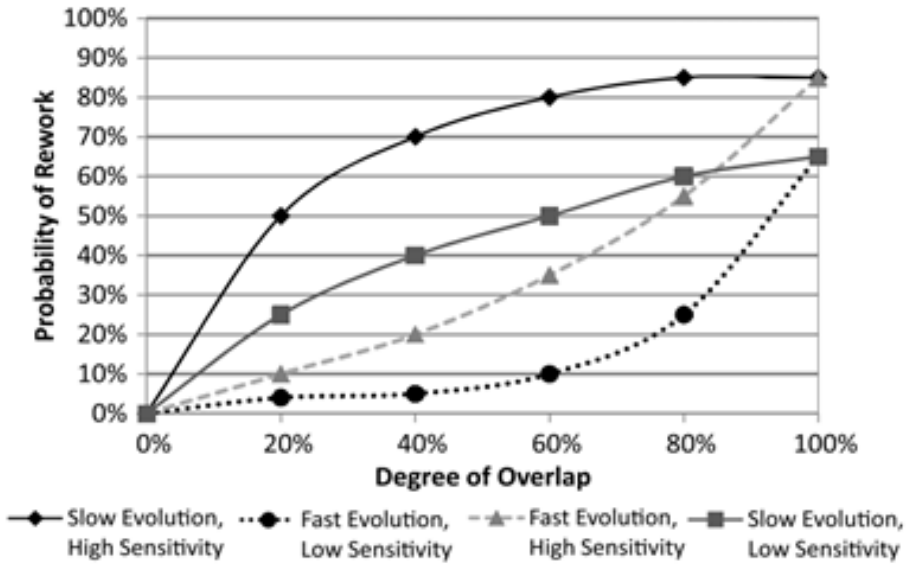


Fig. 3. Theoretical relationship between probability of rework and degree of overlap

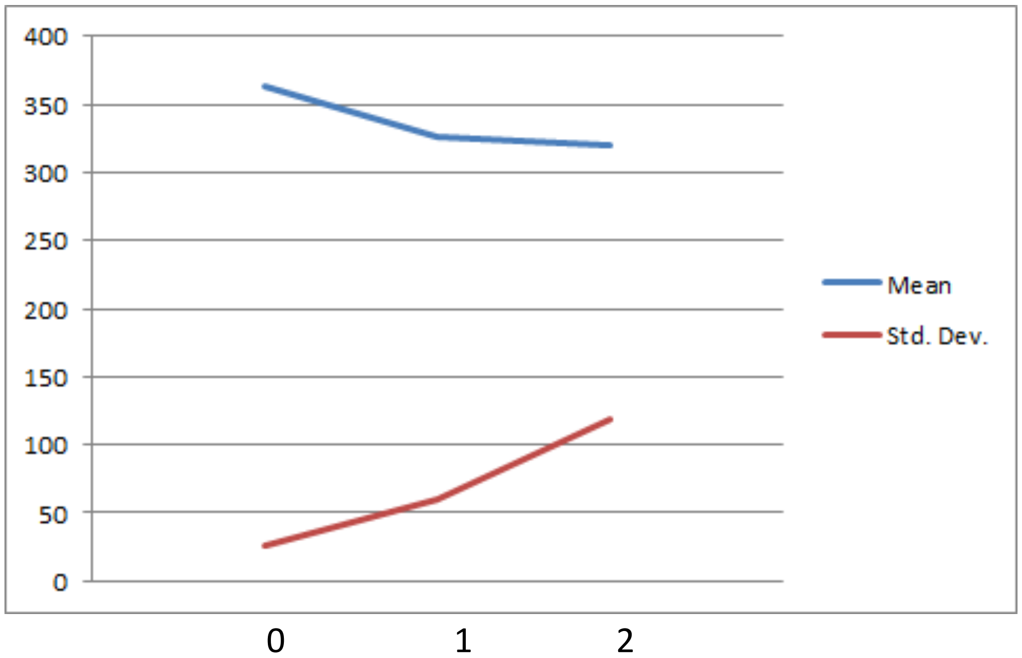
Bogus, S., & Diekmann, J. (2011). Simulation of Overlapping Design Activities in Concurrent Engineering.

Input levers

- Activity duration
- DSM
- Rework Impact
- Evolution
- Sensitivity
- Risk-factor

Impact of increasing risk:

1. Mean project duration decreases
2. Variance/ standard deviation increases
3. Chance of rework increases



Model Functionality

Simulation language: C++

Steps of simulation:

1. Read input files
2. Determine degree of overlap
3. Create baseline schedule
4. Randomize durations
5. Determine real schedule
6. Calculate SAI and forecast total rework
7. Calculate chance of rework for each acitivity
8. Add rework
9. Advance to next status point

Outputs

Two outputs will be generated:

1. Average SAI
2. The actual project duration

This should allow to find link between SAI and project duration for different values of input variables

Examplary problem

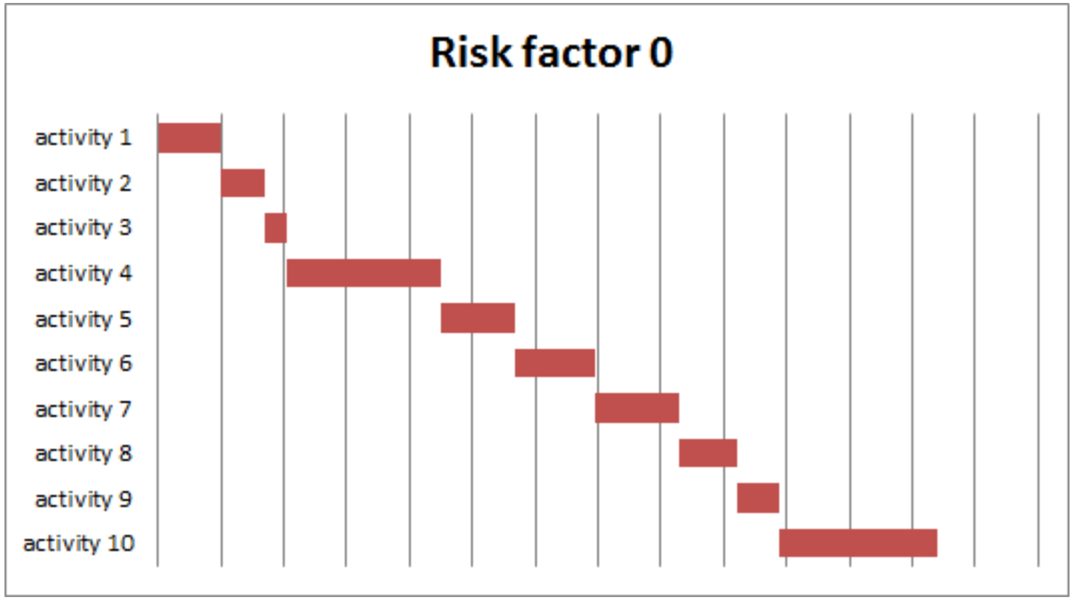
Project characteristics:

- 10 activities
- Start 01/01/2013
- Rework impact 0,5
- Activities use information from all previous activities

10.000 simulation runs for each risk factor

Examplary problem

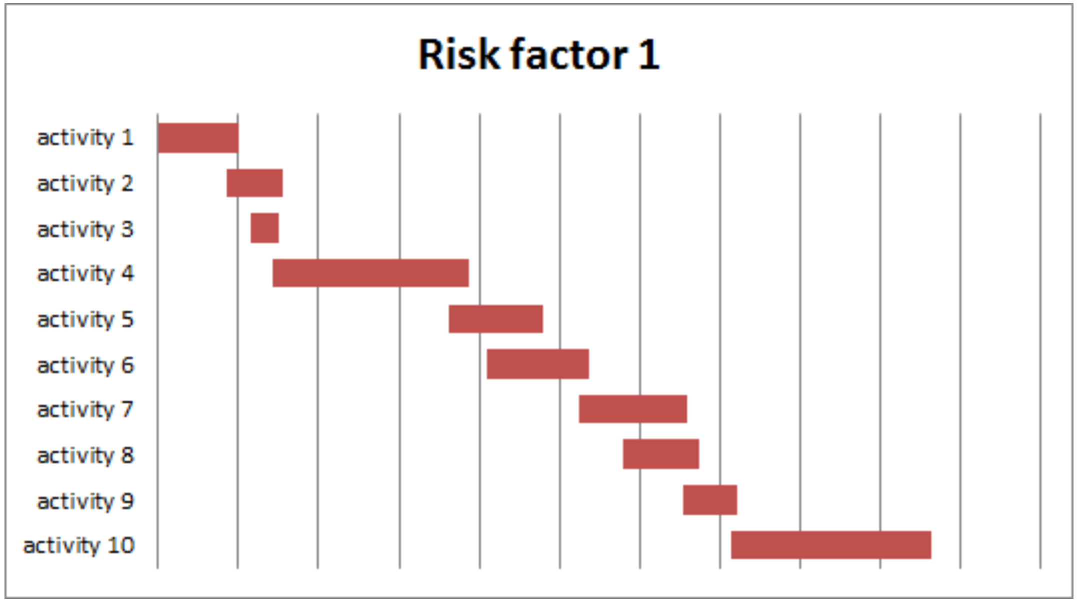
Baseline schedule(risk factor 0):



Expected duration: 371,6667 days

Examplary problem

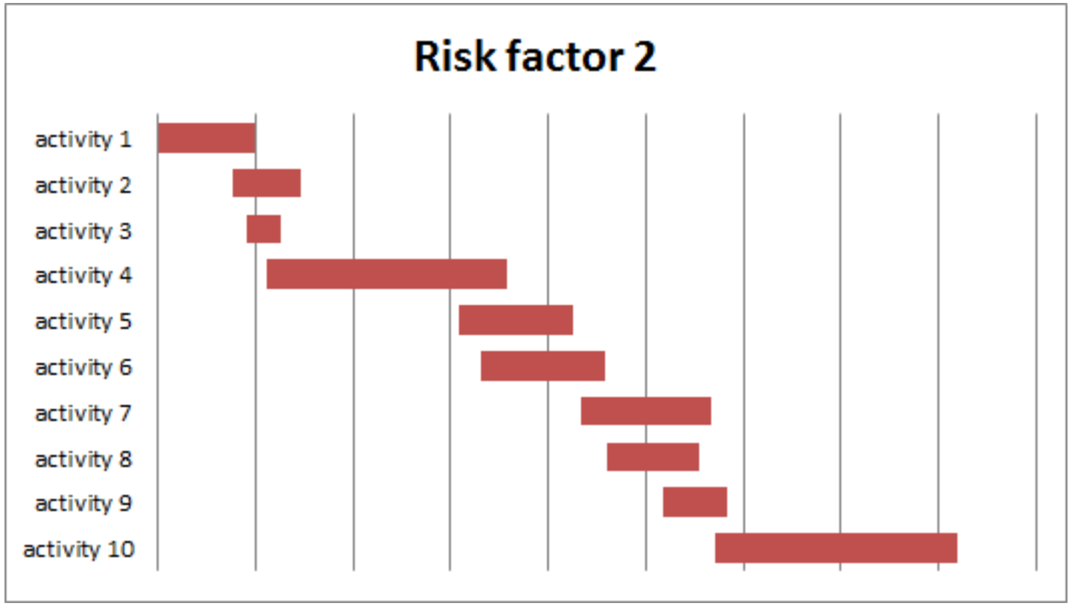
Baseline schedule(risk factor 1):



Expected duration: 289,16 days

Examplary problem

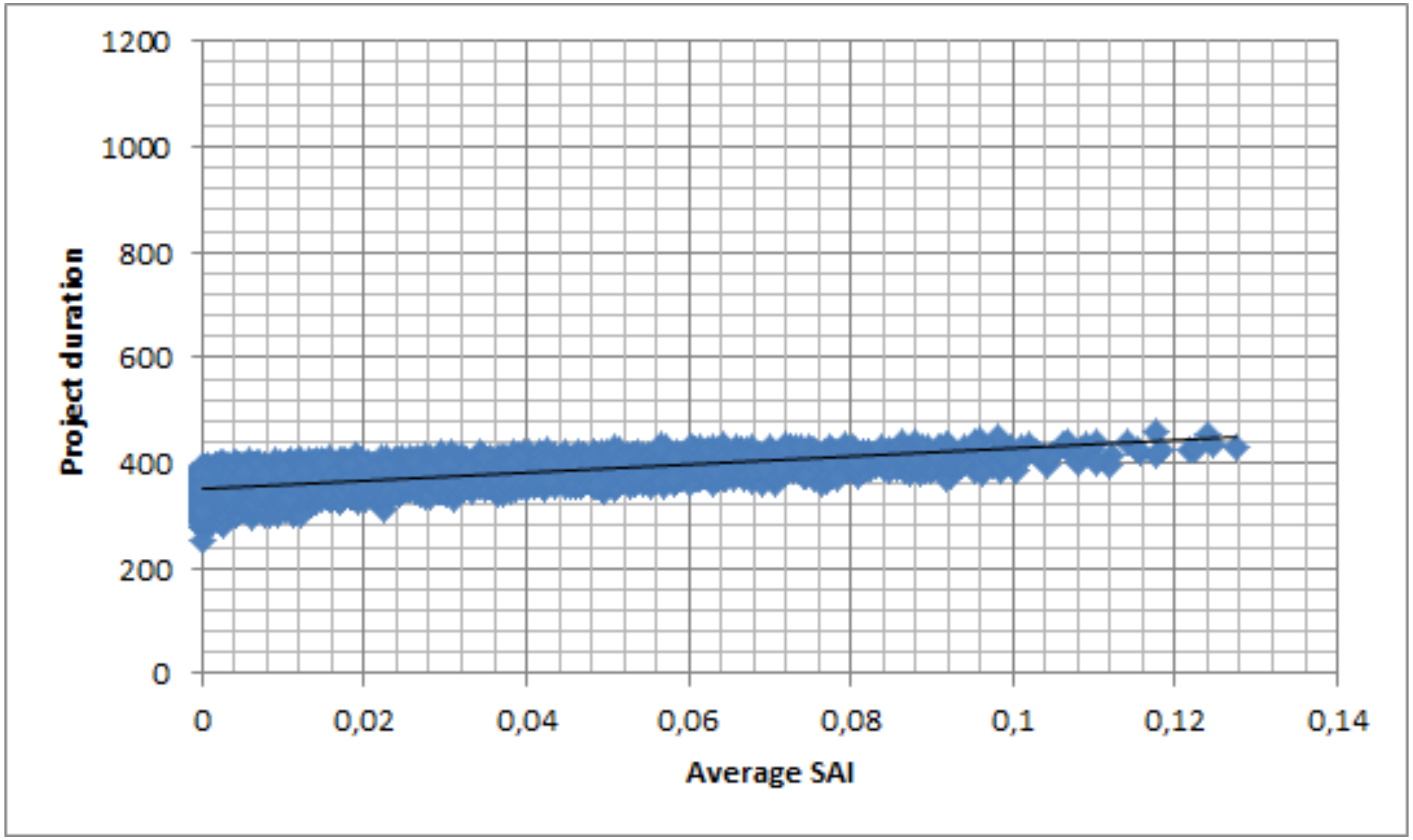
Baseline schedule(risk factor 2):



Expected duration: 246 days

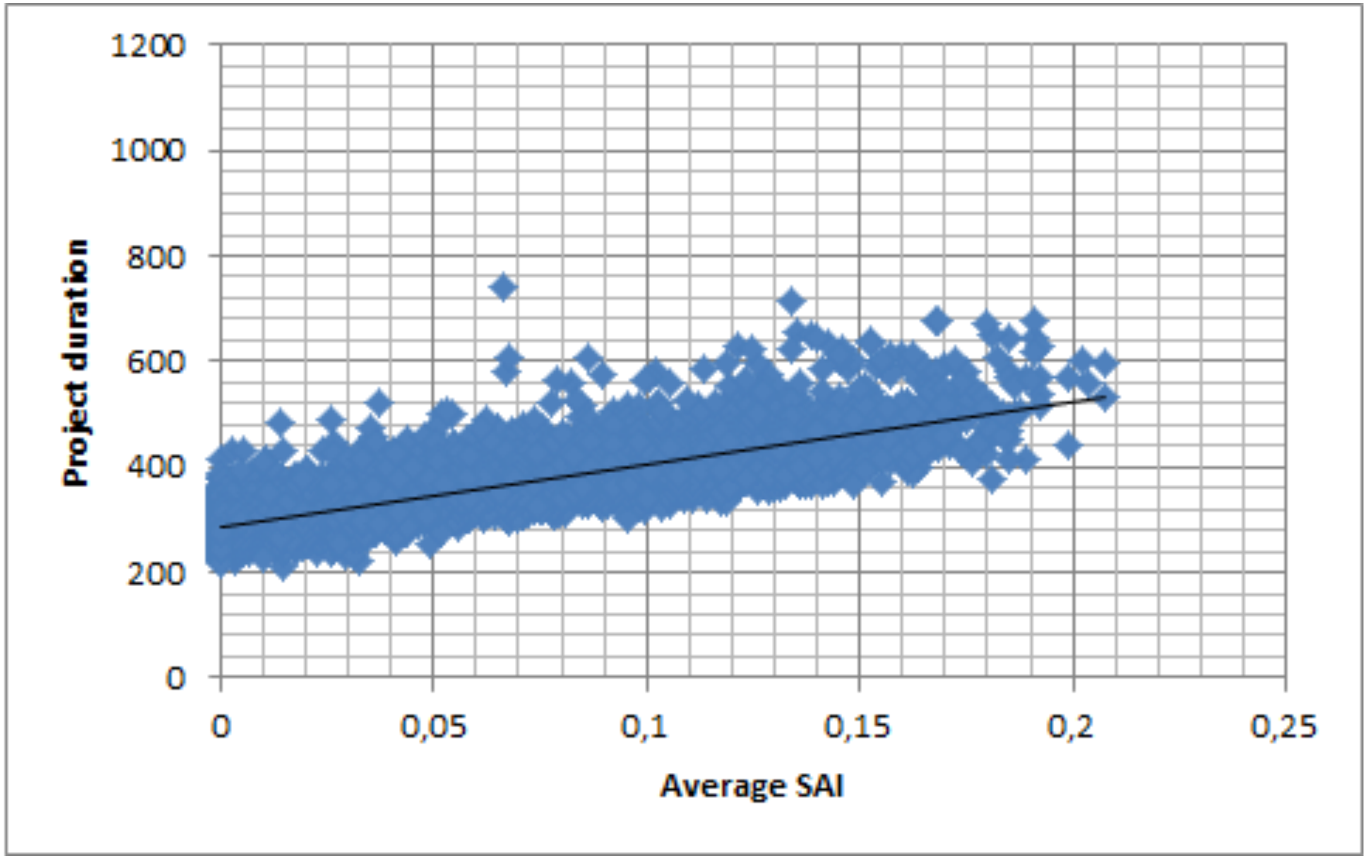
Examplary problem

Risk-factor 0:



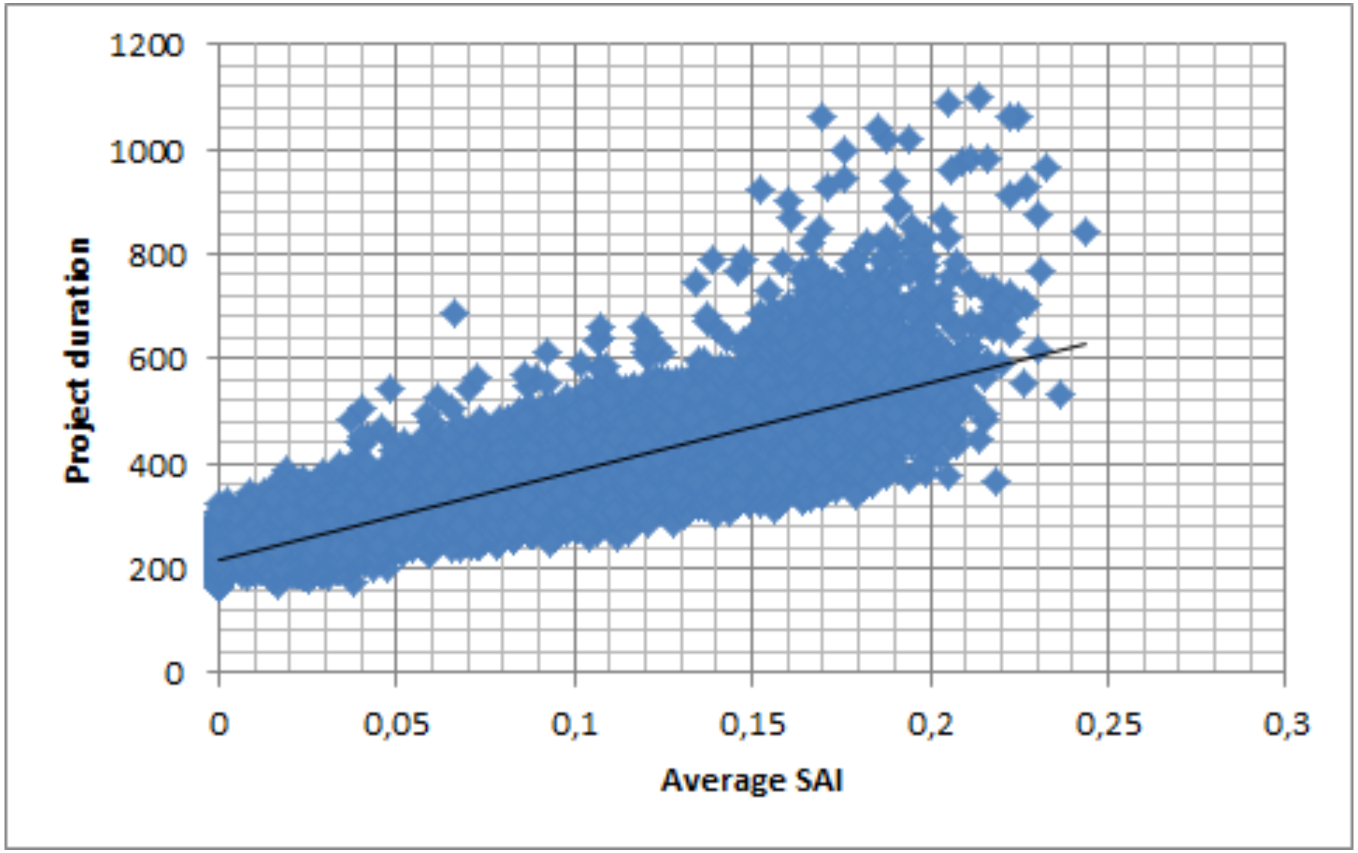
Examplary problem

Risk-factor 1:



Examplary problem

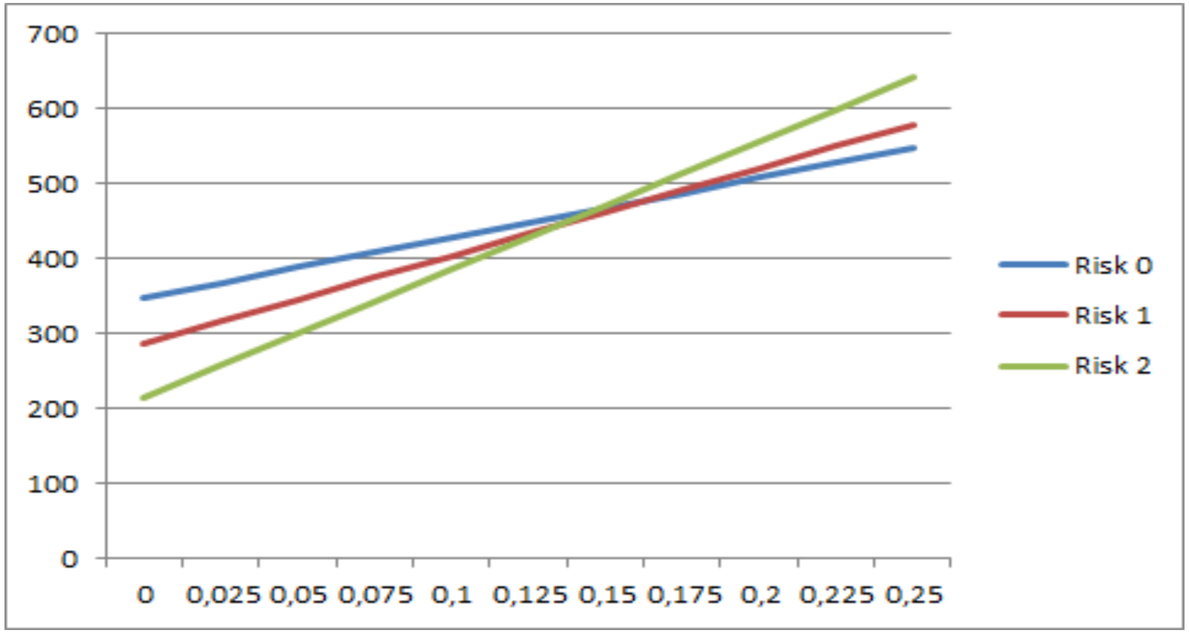
Risk-factor 2:



Examplary problem

Regression results:

1. DUR_0 = 348.722156499855 + 795.1174*SAI
2. DUR_1 = 286.9742+1169.713*SAI
3. DUR_2 = 215.6778+1703.153*SAI



Examplary problem

For what values of SAI is risk not appropiate?

Look at intersections:

1. Risk 0 and Risk 1 at 0,165153 SAI
2. Risk 0 and Risk 2 at 0,146519 SAI

On average from SAI of 0,165 scheduling sequential is better than scheduling concurrent for the example project

Conclusion

The example shows us that concurrent scheduling isn't always beneficial

Simulation model allows managers to identify the degree of schedule adherence when scheduling sequential becomes more beneficial for a certain project

This can be used to, for example, modify the baseline schedule when schedule adherence deteriorates to much during project execution

Questions?

Thank you very much for your attention!

References

- Chua, D. K. H., & Hossain, M. A. (2011). A simulation model to study the impact of early information on design duration and redesign.
- Bogus, S., & Diekmann, J. (2011). Simulation of Overlapping Design Activities in Concurrent Engineering.
- Lipke, W. (2010). Schedule Adherence and Rework.
- Vanhoucke, M. (2009). Measuring time: improving project performance using earned value management (p. 183).
- Bogus, S. M., Molenaar, K. R., & Diekmann, J. E. (2006). Strategies for overlapping dependent design activities.
- Smith, R., & Morrow, J. (2001). Product development process modeling.
- Bogus, S. (2005). Concurrent engineering approach to reducing design delivery time.