Appendix A Model Equations

Name	Definition
% Avail External Concurrence Build Stage	For (i=NofTasks| Graph('Fraction of Tasks Released from Build Stage'[i] ,0,0.1,'External Process Concurrence Relation Build Stage'[i]))
% Avail External Concurrence Operation Stage	For (i=NofTasks| Graph('Fraction of Tasks Released from Operation Stage'[i] ,0,0.1,'External Process Concurrence Relation Operation Stage'[i]))
% Avail Internal Concurrence Build Stage	For (i=NofTasks|Graph('Fraction Perceived Satisfactory Build Stage'[i] ,0,1,'Internal Process Concurrence Relation Graph Input Build Stage'[i]))
% Avail Internal Concurrence Design Stage	For (i=NofTasks|Graph('Fraction Perceived Satisfactory Design Stage'[i],0,1, 'Internal Process Concurrence Relation Graph Input Design Stage'[i]))
% Avail Internal Concurrence Operation Stage	For (i=NofTasks| Graph('Fraction Perceived Satisfactory Operation Stage'[i] ,0,1,'Internal Process Concurrence Relation Graph Input Commission Stage'[i]))
% Available External Concurrence Design Stage	For (i=NofTasks|Graph('Fraction of Tasks Released from Design Stage'[i] ,0,1,'External Process Concurrence Relation Design Stage'[i]))
A lower asymptote	0
Absolute Deviation per Defect Type	For (i=NofDefectsType|abs('Estimated Performance'[i]-'Building Performance'[i]))
Absolute Model Deviation	ArrAverage('Absolute Deviation per Defect Type')
Actor 1 Alignment	('Actor Alignment'[1])
Actor 2 Alignment	('Actor Alignment'[2])
Actor 3 Alignment	('Actor Alignment'[3])
Actor Alignment	For (i=NofActors, j=NofActors|0)
Actor Alignment Build Stage	'Actor 2 Alignment'[2]+'Initial Level of Intra Stage Alignment'
Actor Alignment Decrease	For (i=NofActors, j=NofActors|max(0,'Actor Alignment'[i,j]) /'Individual Actor Deadlines'[i]*'Switch for Alignment')
Actor Alignment Design Stage	'Actor 1 Alignment'[1]+'Initial Level of Intra Stage Alignment'
Actor Alignment Increase	For(i=NofActors,j=NofActors| 'Actor Project Net Increase in Overlap'[i,j] divz0'Actor Project Participation Duration'[i])*1<<mo>>*'Switch for Alignment'
Actor Alignment Operation Stage	'Actor 3 Alignment'[3]+'Initial Level of Intra Stage Alignment'
Actor Build Communication	'Actor Communication'[2]
Actor Build-Operation Alignment	+'Actor 2 Alignment'[3]*'Actor 3 Alignment'[2]+'Initial Level of Inter Stage Alignment'
Actor Build-Operation Communication Rate	'Actor Communication Build Stage'*'Actor Communication Operation Stage'* 'Actor Build-Operation Alignment'
Actor Communication	For (i=NofActors,j=NofActors| 'Actor Project Interaction Rate Increase'[i]*'Actor Project Interaction Rate Increase'[j])/1<<mo>> *'Actor Overlap Percentage of Interaction Duration'
Actor Communication Build Stage	'Actor Build Communication'[2]*'Actor Alignment Build Stage'
Actor Communication Design Stage	'Actor Design Communication'[1]*'Actor Alignment Design Stage'
Actor Communication Operation Stage	'Actor Operation Communication'[3]*'Actor Alignment Operation Stage'
Actor Design -Build Communication Rate	'Actor Communication Design Stage'*'Actor Communication Build Stage'*'Actor Design-Build Alignment'
Actor Design Communication	'Actor Communication'[1]
Actor Design-Build Alignment	'Actor 1 Alignment'[2]*'Actor 2 Alignment'[1]+'Initial Level of Inter Stage Alignment'
Actor Design-Operation Alignment	+'Actor 1 Alignment'[3]*'Actor 3 Alignment'[1]+'Initial Level of Inter Stage Alignment'
Actor Design-Operation Communication Rate	'Actor Communication Design Stage'*'Actor Communication Operation Stage'*'Actor Design-Operation Alignment'
Actor Engagement Overlap Matrix	For (i=NofActors, j=NofActors|If (i>j,'Actor Overlap Backward'[i,j],'Actor Overlap Forward'[i,j]))
Actor Entry Timing	{0,11.5,33}<<mo>>
Actor Inter Stage Cumulative Information	For(i=NofActors|0)
Actor Intra Stage Cumulative Information	For(i=NofActors|0)
Actor Operation Communication	'Actor Communication'[3]
Actor Overlap Backward	For (i=NofActors, j=NofActors| If (i=j,'Actor Project Interaction Duration'[i], If (i=First(NofActors), 0<<mo>>, max('Individual Absolute Actor Deadline'[Index(j)]+'Actor Entry Timing'[Index(j)]-'Actor Entry Timing'[i] ,'Actor Project Interaction Duration'[Index(j)]+'Actor Entry Timing'[Index(j)]-'Actor Entry Timing'[i],0<<mo>>))))
Actor Overlap Forward	For (j=NofActors, i=NofActors| If (i=j,'Actor Project Interaction Duration'[i], If (i=First(NofActors), 0<<mo>>, max('Individual Absolute Actor Deadline'[Index(j)]+'Actor Entry Timing'[Index(j)]-'Actor Entry Timing'[i] ,'Actor Project Interaction Duration'[Index(j)]+'Actor Entry Timing'[Index(j)]-'Actor Entry Timing'[i],0<<mo>>))))

Actor Overlap Percentage of Interaction Duration	For (I=NofActors,j=NofActors|'Actor Engagement Overlap Matrix'[i,j])
Actor Project Absolute Time Overlap	For (i=NofActors, j=NofActors|0)
Actor Project Interaction Duration	For(i=NofActors|0<<mo>>)
Actor Project Interaction Rate Increase	For(I=NofActors|(Step(1,StartTime+'Actor Project Interaction Timing'[i]) -Step(1,StartTime+'Individual Calendar Deadlines'[3]))
Actor Project Interaction Timing	{0,11.5,33}<<mo>>
Actor Project Net Increase in Overlap	For (i=NofActors,j=NofActors|'Actor Project Participation'[i]*'Actor Project Participation'[j])/1<<mo>>
Actor Project Participation	For(I=NofActors|(Step(1,StartTime+'Actor Entry Timing'[i])-Step(1,StartTime+'Individual Calendar Deadlines'[i]))) *(min(1,{ArrSum('Total Project Tasks Scope'-'Net Work Done Design Stage') ,ArrSum('Total Project Tasks Scope'-'Net Work Done Build Stage') ,ArrSum('Total Project Tasks Scope'-'Net Work Done Commission Stage')})) *'Switch Project Overtime' +For(I=NofActors|Step(1,StartTime+'Actor Entry Timing'[i])-Step(1,StartTime+'Individual Calendar Deadlines'[i])) *(1-'Switch Project Overtime')
Actor Project Participation Duration	For(i=NofActors|0<<mo>>)
Actor Schedule Pressure	'Initial Schedule Pressure'
Actor Schedule Pressure Release Delay	1<<mo>>
Actor Work Capacity	For (i=NofActors|00)/1<<mo>>
Additional Required Capacity	For (i=NofActors|ArrSum('Net Project Remaining Work Per Stage'[i])) *{'Avg Initial Completion Duration Design Stage','Avg Initial Completion Duration Build Stage','Avg Initial Completion Duration Commission Stage'} divz0('Individual Actor Available Time to Completion') /1<<mo>>
Approve Defects Operation Stage	max(0/1<<mo>>,min('Approve Task Rate Operation Stage', 'Undiscovered Defects Operation Stage'/1<<mo>>-'Discover Internal Defects Operation Stage'))
Approve Task Rate Build Stage	min('Tasks Completed not Checked Build Stage'/1<<mo>>, ('Quality Assurance Build Stage'-'Discover Intra Stage Defects Build Stage'-'Discover Inter Stage Defects Build Stage') *(1-'Probability of Intra Defect Discovery Build Stage')) +'Defect Correction Rate Build Stage'
Approve Task Rate Design Stage	min('Tasks Completed Not Checked Design Stage'/1<<mo>>, ('Quality Assurance Design Stage'-'Discover Intra Stage Defects Design Stage') *(1-'Probability of Intra Defect Discovery Design Stage')) +'Defect Correction Rate Defects Design Stage'
Approve Task Rate Operation Stage	min('Tasks Completed not Checked Operation Stage'/1<<mo>>, ('Quality Assurance Operation Stage'-'Discover Intra Stage Defects Operation Stage' -'Discover Inter Stage Defects Commission Stage') *(1-'Probability of Intra Defect Discovery Operation Stage')) +'Defect Correction Rate Operation Stage'
Avg Coord Duration Build Stage	0.5<<mo>>
Avg Coord Duration Commission Stage	1<<mo>>
Avg Initial Completion Duration Build Stage	1<<mo>>
Avg Initial Completion Duration Commission Stage	1<<mo>>
Avg Initial Completion Duration Design Stage	1<<mo>>
Avg Iteration Duration Build Stage	1<<mo>>
Avg Iteration Duration Commission Stage	1<<mo>>
Avg Iteration Duration Design Stage	1<<mo>>
B growth rate	{0.5,1,2}
Building Actor Pay Rate	Step(1,StartTime+'Actor Entry Timing'[2]) *('Net Work Done Build Stage'-'Building Actor Payment')/0.5<<mo>>
Building Actor Payment	0
Building Performance	'Remaining Defects'/'Total Project Tasks Scope'
Calendar Time Rate	For (I=NofActors|Step(1,StartTime+'Actor Entry Timing'[i]))
Capacity Decrease	max(0/1<<mo>>, 'Actor Work Capacity'-ArrSum('Net Project Remaining Work Per Stage')/1<<mo>>) /'Time to Reduce Work Capacity'
Capacity Increase	'Additional Required Capacity' *For(i=NofActors|Step(1,StartTime+'Actor Entry Timing'[i])) *min(1,'Relative Schedule Pressure') /'Work Capacity Adjustment Time'
Change Task Rate Build Stage	min('Actor Work Capacity'[2]-'Initial Completion Rate Build Stage' , 'Tasks to be Iterated Build Stage'/'Avg Iteration Duration Build Stage')
Change Task Rate Commission Stage	min('Actor Work Capacity'[3]-'Initial Completion Rate Operation Stage' , 'Tasks to be Iterated Operation Stage'/'Avg Iteration Duration Commission Stage')
Change Task Rate Design Stage	min('Actor Work Capacity'[1]-'Initial Completion Rate Design Stage' , 'Tasks to be Iterated Design Stage'/'Avg Iteration Duration Design Stage')
Commission Actor Pay Rate	Step(1,StartTime+'Actor Entry Timing'[3]) *('Net Work Done Commission Stage'-'Commission Actor Payment')/0.5<<mo>>
Commission Actor Payment	0
Corrected Defects	sum('Corrected Defects Design Stage','Corrected Defects Build Stage','Corrected Defects Operation Stage')

Corrected Defects Build Stage	For (I=NofDefectsType|0)
Corrected Defects Design Stage	For (I=NofDefectsType|0)
Corrected Defects Operation Stage	For (i=NofDefectsType|0)
Deadlines in Months	Number('Individual Actor Deadlines'/12)
Decrease in Schedule Pressure	max(0/1<<mo>>,'Actor Schedule Pressure'/('Actor Schedule Pressure Release Delay') *For (I=1..3|(1-'Total Remaining Tasks'[i]/ArrSum('Total Project Tasks Scope'))))
Decrease Interstage Defect Probability to Build Stage	min('Number of Defects Design to Build Stage'/1<<mo>>, 'Defects Released Build Stage'+'Discover Internal Defects Build Stage')
Decrese Interstage Defect Probability to Operation Stage	min('Number of Defects Build to Operation Stage'/1<<mo>>, 'Approve Defects Operation Stage'+'Discover Internal Defects Operation Stage')
Defect Correction Rate Build Stage	'Known Defects Build Stage' divz0 'Tasks to be Iterated Build Stage' *(1-('Probability of Generating Additional Internal Defect when Receiving Downstream Tasks Build Stage')) *'Change Task Rate Build Stage' *(1-'Project Changes Difficulty'[2])
Defect Correction Rate Defects Design Stage	'Known Defects Design Stage' divz0 'Tasks to be Iterated Design Stage' *(1-('Probability of Generating Additional Internal Defect when Receiving Downstream Tasks Design Stage')) *'Change Task Rate Design Stage' *(1-'Project Changes Difficulty'[1])
Defect Correction Rate Operation Stage	'Known Defects Operation Stage' divz0 'Tasks to be Iterated Operation Stage' *(1-('Probability of Generating Additional Internal Defect when Receiving Downstream Tasks Operation Stage')) *'Change Task Rate Commission Stage' *(1-'Project Changes Difficulty'[3])
Defect Knock on Effect Build Stage	(1+('Undiscovered Defects Build Stage'+'Known Defects Build Stage')/'Scope Build Stage') ^('Strength of Defect Effect Build Stage'-1)
Defect Knock on Effect Design Stage	(1+('Undiscovered Defects Design Stage'+'Known Defects Design Stage') /'Total Project Tasks Scope')^('Strength of Defect Effect Design Stage'-1)
Defects Released Build Stage	max(0/1<<mo>>, min('Approve Task Rate Build Stage', 'Undiscovered Defects Build Stage'/1<<mo>>-'Discover Internal Defects Build Stage'))
Defects Released Design Stage	max(0/1<<mo>>,min('Approve Task Rate Design Stage', 'Undiscovered Defects Design Stage'/1<<mo>> -'Discover Internal Defects Design Stage'))
Defects Released with Workarounds to Build Stage	('Known Defects Design Stage'/1<<mo>>-'Defect Correction Rate Design Stage') *'Project Changes Difficulty'[1]
Defects Released with Workarounds to Operation Stage	('Known Defects Build Stage'/1<<mo>>-'Defect Correction Rate Build Stage') *'Project Changes Difficulty'[2]
Defects to Build Stage	'Defect Knock on Effect Design Stage' *('Defects Released Design Stage'+'Defects Released with Workarounds to Build Stage')
Defects to Operation Stage	'Defect Knock on Effect Build Stage'*('Defects Released Build Stage' +'Defects Released with Workarounds to Operation Stage')
Design Actor Pay Rate	Step(1,StartTime+'Actor Entry Timing'[1]) *('Net Work Done Design Stage'-'Design Actor Payment')/0.5<<mo>>
Design Actor Payment	0
Discover Inter Stage Defects Build Stage	min('Tasks Completed not Checked Build Stage'/1<<mo>>, ('Quality Assurance Build Stage'-'Discover Intra Stage Defects Build Stage') *max(0,('Probability of Defects Design to Build Stage' -'Probability Intra Stage Defect Build Stage')*'Probability of Intra Defect Discovery Build Stage')) *(1-'Proportion of Design Defects Possible to Rework in Build Stage')
Discover Inter Stage Defects Commission Stage	min('Tasks Completed not Checked Operation Stage'/1<<mo>>, ('Quality Assurance Operation Stage'-'Discover Intra Stage Defects Operation Stage') *max(0,('Probability of Defects Build to Operation Stage' -'Probability Intra Stage Defect Commission Stage')*'Probability of Intra Defect Discovery Operation Stage')) *(1-'Proportion of Build Defects Possible to Rework in Operation Stage')
Discover Inter Stage Defects in Build Stage	min('Undiscovered Defects Build Stage'/1<<mo>>,'Discover Inter Stage Defects Build Stage')
Discover Inter Stage Defects in Operation Stage	min('Undiscovered Defects Operation Stage'/1<<mo>>,'Discover Inter Stage Defects Commission Stage')
Discover Internal Defects Build Stage	min('Undiscovered Defects Build Stage'/1<<mo>>, 'Discover Intra Stage Defects Build Stage')
Discover Internal Defects Design Stage	min('Undiscovered Defects Design Stage'/1<<mo>>, 'Discover Intra Stage Defects Design Stage')
Discover Internal Defects Operation Stage	min('Undiscovered Defects Operation Stage'/1<<mo>>, 'Discover Intra Stage Defects Operation Stage')
Discover Intra Stage Defects Build Stage	min('Quality Assurance Build Stage','Tasks Completed not Checked Build Stage'/1<<mo>> *'Probability of Intra Defect Discovery Build Stage'*'Probability Intra Stage Defect Build Stage') +('Tasks Completed not Checked Build Stage'* 'Proportion of Design Defects Possible to Rework in Build Stage' *'Probability of Defects Design to Build Stage' *'Probability of Intra Defect Discovery Build Stage'/1<<mo>>)
Discover Intra Stage Defects Design Stage	min('Quality Assurance Design Stage','Tasks Completed Not Checked Design Stage'/1<<mo>> *'Probability of Intra Defect Discovery Design Stage' *'Probability of Intra Stage Defect Design Stage')
Discover Intra Stage Defects Operation Stage	min('Quality Assurance Operation Stage','Tasks Completed not Checked Operation Stage'/1<<mo>> *'Probability of Intra Defect Discovery Operation Stage'*'Probability Intra Stage Defect Commission Stage') +('Tasks Completed not Checked Operation Stage'* 'Proportion of Build Defects Possible to Rework in Operation Stage' *'Probability of Defects Build to Operation Stage' *'Probability of Intra Defect Discovery Operation Stage'/1<<mo>>)
Estimated Performance	{0.15, 0.1, 0.3, 0.5, 0.5, 0.5, 1, 0.5}
External Process Concurrence Relation Build Stage	For (i=NofTasks|{0,0,0,0,0,0,0,0,0.1,1})

External Process Concurrence Relation Design Stage	For (i=NofTasks|{0,1})
External Process Concurrence Relation Operation Stage	For (i=NofTasks|{0,0,0,0,0,0,0,0,0.1,1})
Fraction of Tasks Released from Build Stage	'Net Work Done Design Stage'/'Total Project Tasks Scope'
Fraction of Tasks Released from Design Stage	'Tasks Not Completed Design Stage'/'Total Project Tasks Scope'
Fraction of Tasks Released from Operation Stage	'Net Work Done Build Stage'/'Scope Build Stage'
Fraction Perceived Satisfactory Build Stage	'Net Work Done Build Stage'/'Scope Build Stage'
Fraction Perceived Satisfactory Design Stage	'Net Work Done Design Stage'/'Total Project Tasks Scope'
Fraction Perceived Satisfactory Operation Stage	'Net Work Done Commission Stage'/'Stage Scope Operation Stage'
Generate Defects in Iteration Build Stage	'Known Defects Build Stage' divz0 'Tasks to be Iterated Build Stage' *'Probability of Generating an Internal Defect in Iteration Build Stage' *'Change Task Rate Build Stage'
Generate Defects in Iteration Design Stage	'Known Defects Design Stage' divz0 'Tasks to be Iterated Design Stage' *'Probability of Generating an Internal Defect in Iteration Design Stage' *'Change Task Rate Design Stage'
Generate Defects in Iteration Operation Stage	'Known Defects Operation Stage' divz0 'Tasks to be Iterated Operation Stage' *('Probability of Generating an Internal Defect in Iteration Operation Stage') *'Change Task Rate Commission Stage'
Generate Internal Defects in Completion Design Stage	'Initial Completion Rate Design Stage'*'Probability of Intra Stage Defect Design Stage' *(1-'Switch Effect of Communication on Defect Generation'*'Actor Inter Stage Cumulative Communication'[1] /'Total Project Tasks Scope')
Generate Internal Defects in Initial Completion Build Stage	'Initial Completion Rate Build Stage'*'Probability Intra Stage Defect Build Stage'//*'Defect Knock on Effect Design Stage'// *(1-'Switch Effect of Communication on Defect Generation'*'Actor Inter Stage Cumulative Communication'[2] /'Total Project Tasks Scope')
Generate Internal Defects in Initial Completion Commission Stage	'Initial Completion Rate Operation Stage'*'Probability Intra Stage Defect Commission Stage'//*'Defect Knock on Effect Build Stage'// *(1-'Switch Effect of Communication on Defect Generation'*'Actor Inter Stage Cumulative Communication'[2] /'Total Project Tasks Scope')
Increase in Schedule Pressure	(max(0,('Required Time to Completion'divz0'Individual Actor Available Time to Completion') *('Sensitivity to Time Pressure'))) /'Schedule Pressure Adjustment Delay' *For(i=NofActors|Step(1,StartTime+'Actor Entry Timing'[i]))
Increase Interstage Probability to Build Stage	'Defects Released Design Stage'
Increase Interstage Probability to Operation Stage	'Defects Released Build Stage'
Individual Absolute Actor Deadline	'Individual Actor Deadlines'
Individual Actor Available Time to Completion	max(0<<mo>>,'Individual Absolute Actor Deadline'-'Individual Actor Calendar Time')
Individual Actor Calendar Time	For (I=NofActors|0<<mo>>)
Individual Actor Deadline Decision	If ('Net Change Required in Deadline'>'Relative Time Crunch'*'Individual Actor Available Time to Completion', 'Relative Time Crunch'*'Actor Schedule Pressure'/'Initial Schedule Pressure' *max(0<<mo>>,'Required Time to Completion'-'Individual Actor Available Time to Completion'),0<<mo>>)/1<<mo>> *'Switch Deadline Extension'
Individual Actor Deadlines	{12.5,22,36}*1<<mo>>
Individual Calendar Deadlines	'Actor Entry Timing'+'Individual Actor Deadlines'
Information Decrease	For (i=NofActors| 'Actor Inter Stage Cumulative Communication'[i]/'Individual Actor Deadlines'[i]) /max(1,{'Actor Design-Build Alignment','Actor Build-Operation Alignment','Actor Design-Operation Alignment'})
Information Increase Delay	1<<mo>>
Initial Completion Rate Build Stage	min('Tasks Not Completed Build Stage'/'Avg Initial Completion Duration Build Stage','Actor Work Capacity'[2] , 'Tasks Available for Completion Build Stage'/'Avg Initial Completion Duration Build Stage')
Initial Completion Rate Design Stage	min('Tasks Not Completed Design Stage'/'Avg Initial Completion Duration Design Stage','Actor Work Capacity'[1] , 'Tasks Available for Completion Design Stage'/'Avg Initial Completion Duration Design Stage')
Initial Completion Rate Operation Stage	min('Tasks not Completed Operation Stage'/'Avg Initial Completion Duration Commission Stage','Actor Work Capacity'[3] , 'Tasks Available for Completion Operation Stage'/'Avg Initial Completion Duration Commission Stage')
Initial Level of Inter Stage Alignment	0.5
Initial Level of Intra Stage Alignment	0.5
Initial Probability of Generating Additional Internal Defect when Receiving Downstream Tasks Build Stage	0
Initial Probability of Generating an Additional Internal Defect when Receiving Downstream Tasks Commission Stage	0
Initial Probability of Generating an Additional Internal Defect when Receiving Downstream Tasks Design Stage	0

Initial Probability of Generating an Internal Defect in Iteration Build Stage	'Probability of Generating Internal Defect in Iteration Construction Stage'
Initial Probability of Generating an Internal Defect in Iteration Commission Stage	'Probability of Generating Internal Defect in Iteration Operation Stage'
Initial Probability of Intra Defect Discovery Build Stage	{28%, 15%, 0%, 0%, 0%, 0%, 40%, 43%}
Initial Probability of Intra Defect Discovery Commission Stage	{53%, 53%, 38%, 25%, 45%, 25% ,25%, 38%}
Initial Probability of Intra Defect Discovery Design Stage	{48%, 45%, 50% ,43% ,45% ,43%, 45%, 53%}
Initial Probability of Generating an Internal Defect in Iteration Design Stage	'Probability of Generating Internal Defect in Iteration Design Stage'
Initial Schedule Pressure	For (i=NofActors|1)
Input Probability Intra Stage Defect Build Stage	{35%, 20%, 13%, 13%, 0%, 13%, 40%, 30%}
Input Probability Intra Stage Defect Operation Stage	{30%, 35%, 50%, 50%,50% ,50%, 20%, 55%}
Inter Stage Information Increase	min({'Actor Design -Build Communication Rate' ,'Actor Build-Operation Communication Rate','Actor Design-Operation Communication Rate'}, ('Max Info per Kind of Task'-'Actor Inter Stage Cumulative Communication'))/'Information Increase Delay'*'Switch for Com'
Internal Process Concurrence Relation Graph Input Build Stage	For (i=NofTasks|{1,1})
Internal Process Concurrence Relation Graph Input Commission Stage	For (i=NofTasks|{1,1})
Internal Process Concurrence Relation Graph Input Design Stage	For (i=NofTasks|{1,1})
Intra Stage Information Decrease	'Actor Intra Stage Cumulative Communication' /max(1,{'Actor Alignment Design Stage','Actor Alignment Build Stage','Actor Alignment Operation Stage'}) /'Individual Actor Deadlines'
Intra Stage Information Increase	min({'Actor Communication Design Stage','Actor Communication Build Stage','Actor Communication Operation Stage'} ,('Max Info per Kind of Task'-'Actor Intra Stage Cumulative Communication')) /'Information Increase Delay'*'Switch for Com'
Inverse Derivative	Derivn('Project Changes Difficulty')
K upper asymptote	{1,1,1}
Known Defects Build Stage	For (I=NofDefectsType|0)
Known Defects Design Stage	For (I=NofDefectsType|0)
Known Defects Operation Stage	For (I=NofDefectsType|0)
M max growth t	{0.2,0.5,1}
Max Info per Kind of Task	ArrMax('Total Project Tasks Scope')*1
Net Change Required in Deadline	max(0<<mo>>,('Actor Schedule Pressure'*1<<mo>>-'Individual Absolute Actor Deadline'))/'Resistance to Move Deadline' *'Required Time to Completion'*('Actor Schedule Pressure'/'Initial Schedule Pressure')
Net Defects	'Corrected Defects'-'Remaining Defects'
Net Project Remaining Work Per Stage	{'Total Project Tasks Scope'-'Net Work Done Design Stage', 'Total Project Tasks Scope'-'Net Work Done Build Stage', 'Total Project Tasks Scope'-'Net Work Done Commission Stage'}
Net Work Done Build Stage	For (i=NofTasks|0)
Net Work Done Commission Stage	For (i=NofTasks|0)
Net Work Done Design Stage	For (i=NofTasks|0)
Net Work Increase Design Stage	'Tasks to Build Stage'-'Tasks Coordinate Design Stage Build Stage'
Net Work Rate Build Stage	'Tasks to Operation Stage'-'Tasks Coordinate Build Stage Commission Stage'
Net Work Rate Commission Stage	'Tasks to Project Completion'
Number of Defects Build to Operation Stage	0
Number of Defects Design to Build Stage	0
Percentage Work Done	{'Net Work Done Design Stage' / ArrSum('Total Project Tasks Scope') ,'Net Work Done Build Stage' / ArrSum('Total Project Tasks Scope') ,'Net Work Done Commission Stage' / ArrSum('Total Project Tasks Scope')}
Probability Inter Stage Defect Operation	'Probability of Inter Stage Defect Operation Stage' /(1+'Actor Build-Operation Alignment'*'Actor Inter Stage Cumulative Communication'[2])
Probability Intra Stage Defect Build Stage	'Estimated Performance'*'Input Probability Intra Stage Defect Build Stage'
Probability Intra Stage Defect Commission Stage	'Estimated Performance'*'Input Probability Intra Stage Defect Operation Stage'
Probability Intra Stage Defect Design Stage	{35%, 45%, 38%, 38%, 50%, 38%, 40%, 15%}
Probability of Defects Build to Operation Stage	'Number of Defects Build to Operation Stage'/'Total Project Tasks Scope'

Probability of Defects Design to Build Stage	'Number of Defects Design to Build Stage'/'Total Project Tasks Scope'
Probability of Generating Additional Internal Defect when Receiving Downstream Tasks Build Stage	max(0,'Initial Probability of Generating Additional Internal Defect when Receiving Downstream Tasks Build Stage' -'Actor Inter Stage Cumulative Communication'[2]/'Total Project Tasks Scope')
Probability of Generating Additional Internal Defect when Receiving Downstream Tasks Design Stage	max(0,'Initial Probability of Generating an Additional Internal Defect when Receiving Downstream Tasks Design Stage' -'Actor Inter Stage Cumulative Communication'[1]/'Total Project Tasks Scope')
Probability of Generating Additional Internal Defect when Receiving Downstream Tasks Operation Stage	max(0,'Initial Probability of Generating an Additional Internal Defect when Receiving Downstream Tasks Commission Stage' -'Actor Inter Stage Cumulative Communication'[3]/'Total Project Tasks Scope')
Probability of Generating an Internal Defect in Iteration Build Stage	max(0,'Initial Probability of Generating an Internal Defect in Iteration Build Stage' -('Actor Inter Stage Cumulative Communication'[2]+'Actor Intra Stage Cumulative Communication'[2]) /'Total Project Tasks Scope')
Probability of Generating an Internal Defect in Iteration Design Stage	max(0,'Initial Probability of Generating an Internal Defect in Iteration Design Stage' -('Actor Intra Stage Cumulative Communication'[1]+'Actor Inter Stage Cumulative Communication'[1]) /'Total Project Tasks Scope')
Probability of Generating an Internal Defect in Iteration Operation Stage	max(0,'Initial Probability of Generating an Internal Defect in Iteration Commission Stage' -('Actor Inter Stage Cumulative Communication'[2]+'Actor Intra Stage Cumulative Communication'[3]) /'Total Project Tasks Scope')
Probability of Generating Internal Defect in Iteration Construction Stage	0
Probability of Generating Internal Defect in Iteration Design Stage	0
Probability of Generating Internal Defect in Iteration Operation Stage	0
Probability of Inter Stage Defect Build Stage	For (i=NofDefectsType|0)
Probability of Inter Stage Defect Design Stage	For (I=NofDefectsType|0)
Probability of Inter Stage Defect Operation Stage	For (i=NofDefectsType|0)
Probability of Intra Defect Discovery Build Stage	min(1, 'Initial Probability of Intra Defect Discovery Build Stage' *('Initial Probability of Intra Defect Discovery Build Stage'+ 'Switch Effect of Communication on Defect Detection'* ('Actor Inter Stage Cumulative Communication'[2]*'Actor Intra Stage Cumulative Communication'[2]) /'Total Project Tasks Scope'^2) divz0('Initial Probability of Intra Defect Discovery Build Stage'))
Probability of Intra Defect Discovery Design Stage	min(1,'Initial Probability of Intra Defect Discovery Design Stage' *('Initial Probability of Intra Defect Discovery Design Stage'+ 'Switch Effect of Communication on Defect Detection'* ('Actor Inter Stage Cumulative Communication'[1]*'Actor Intra Stage Cumulative Communication'[1]) /'Total Project Tasks Scope'^2) divz0('Initial Probability of Intra Defect Discovery Design Stage'))
Probability of Intra Defect Discovery Operation Stage	min(1,'Initial Probability of Intra Defect Discovery Commission Stage' *('Initial Probability of Intra Defect Discovery Commission Stage'+ 'Switch Effect of Communication on Defect Detection'* ('Actor Inter Stage Cumulative Communication'[2]*'Actor Intra Stage Cumulative Communication'[3]) /'Stage Scope Operation Stage'^2) divz0('Initial Probability of Intra Defect Discovery Commission Stage'))
Probability of Intra Stage Defect Design Stage	'Probability Intra Stage Defect Design Stage'*'Estimated Performance'
Project Changes Difficulty	'A lower asymptote'+('K upper asymptote'-'A lower asymptote') /(1+'Q Y0'*exp(-'B growth rate'*('Time value'-'M max growth t')))^1/v
Project Completed	For (i=NofTasks|0)
Project Quality %	ArrAverage(For (i=NofTasks|1-'Remaining Defects'[i]/'Total Project Tasks Scope'[i]))
Proportion of Build Defects Possible to Rework in Operation Stage	0
Proportion of Design Defects Possible to Rework in Build Stage	0
Q Y0	{100,100,100}
Quality Assurance Build Stage	max(0/1<<mo>>,'Actor Work Capacity'[2]-'Initial Completion Rate Build Stage')
Quality Assurance Design Stage	max(0/1<<mo>>,'Actor Work Capacity'[1]-'Initial Completion Rate Design Stage')
Quality Assurance Operation Stage	max(0/1<<mo>>, 'Actor Work Capacity'[3]-'Initial Completion Rate Operation Stage')
Rate_1	'Tasks to Build Stage'+'Change Task Rate Design Stage'
Rate_2	'Tasks to Operation Stage'+'Change Task Rate Build Stage'
Rate_3	'Tasks to Project Completion'+'Change Task Rate Commission Stage'
Relative Schedule Pressure	'Actor Schedule Pressure'divz0 'Initial Schedule Pressure'
Relative Time Crunch	'Required Time to Completion'/max(1<<mo>>,'Individual Actor Available Time to Completion')
Release Package Size Build Stage	0.1
Release Package Size Design Stage	0.1
Release Package Size Operation Stage	0.2

Release Task Delay Build Stage	0.5<<mo>>
Release Task Delay Design Stage	0.5<<mo>>
Release Task Delay Operation Stage	0.5<<mo>>
Release Task Rate Build Stage	'Tasks Approved Build Stage' * 'Release Time Build Stage'/'Release Task Delay Build Stage'
Release Task Rate Design Stage	'Tasks Approved Design Stage'*'Release Time Design Stage'/'Release Task Delay Design Stage'
Release Task Rate Operation Stage	'Tasks Approved Operation Stage' * 'Release Time Commission Stage'/'Release Task Delay Operation Stage'
Release Time Build Stage	If ('Release Package Size Build Stage'/('Scope Build Stage'-'Tasks Released Build Stage') <='Tasks Approved Build Stage',1,0)
Release Time Commission Stage	If ('Release Package Size Operation Stage'/('Stage Scope Operation Stage'-'Tasks Released Operation Stage')<='Tasks Approved Operation Stage',1,0)
Release Time Design Stage	If ('Release Package Size Design Stage'/('Total Project Tasks Scope'-'Tasks Released Design Stage') <='Tasks Approved Design Stage',1,0)
Remaining Defects	(sum('Known Defects Build Stage', 'Known Defects Operation Stage', 'Known Defects Design Stage', 'Undiscovered Defects Build Stage', 'Undiscovered Defects Operation Stage' ,'Undiscovered Defects Design Stage' ,'Undiscovered Defects Post Commission Stage'))
Remaining Tasks Build Stage	'Total Project Tasks Scope'-'Net Work Done Build Stage'
Remaining Tasks Commission Stage	'Total Project Tasks Scope'-'Net Work Done Commission Stage'
Remaining Tasks Design Stage	'Total Project Tasks Scope'-'Net Work Done Design Stage'
Required Time to Completion	max(0<<mo>>, {ArrSum('Remaining Tasks Design Stage')*'Avg Initial Completion Duration Design Stage' ,ArrSum('Remaining Tasks Build Stage')*'Avg Initial Completion Duration Build Stage' ,ArrSum('Remaining Tasks Commission Stage')*'Avg Initial Completion Duration Commission Stage'} divz0 Number(max('Additional Required Capacity','Actor Work Capacity')))
Resistance to Move Deadline	5<<mo>>
Return Defects to Build Stage	'Discover Inter Stage Defects in Operation Stage'/'Defect Knock on Effect Build Stage'
Return Defects to Design Stage	'Discover Inter Stage Defects in Build Stage'/'Defect Knock on Effect Design Stage'
Schedule Pressure Adjustment Delay	1<<mo>>
Scope Build Stage	'Total Project Tasks Scope'
Sensitivity to Time Pressure	1+(1-'Individual Actor Available Time to Completion'/'Individual Actor Deadlines')
Stage Scope Operation Stage	'Scope Build Stage'
Strength of Defect Effect Build Stage	{14.8669944, 4.972929, 16.6743387, 8.090426, 9.55744500, 7.65207226, 3.98072393, 24.1888906}
Strength of Defect Effect Design Stage	{7.62984163, 8.5019940, 7.99256187, 8.32075941, 5.05057588, 7.82428271, 2.58957034, 9.81862568}
Switch Deadline Extension	0
Switch Effect of Communication on Defect Detection	1
Switch Effect of Communication on Defect Generation	1
Switch for Alignment	1
Switch for Com	1
Switch Project Overtime	0
Tasks Approved Build Stage	For (i=NofTasks|0)
Tasks Approved Design Stage	For (i=NofTasks|0)
Tasks Approved Operation Stage	For (i=NofTasks|0)
Tasks Available for Completion Build Stage	'Scope Build Stage'*min('% Avail Internal Concurrence Build Stage','% Avail External Concurrence Build Stage')
Tasks Available for Completion Design Stage	'Total Project Tasks Scope'*min('% Avail Internal Concurrence Design Stage','% Available External Concurrence Design Stage')
Tasks Available for Completion Operation Stage	'Stage Scope Operation Stage'*min('% Avail Internal Concurrence Operation Stage','% Avail External Concurrence Operation Stage')
Tasks Completed not Checked Build Stage	For (i=NofTasks|0)
Tasks Completed Not Checked Design Stage	For (i=NofTasks|0)
Tasks Completed not Checked Operation Stage	For (i=NofTasks|0)
Tasks Coordinate Build Stage Commission Stage	min(1,'Actor Build-Operation Alignment')*'Tasks to Coordinate Operation Stage'/'Avg Coord Duration Commission Stage' *(1-'Project Changes Difficulty'[2])
Tasks Coordinate Design Stage Build Stage	min(1,'Actor Design-Build Alignment')*'Tasks to Coordinate Build Stage'/'Avg Coord Duration Build Stage' *(1-'Project Changes Difficulty'[1])

Tasks Not Completed Build Stage	For (i=NofTasks|0)
Tasks Not Completed Design Stage	'Total Project Tasks Scope'
Tasks not Completed Operation Stage	For (i=NofTasks|0)
Tasks Not Coordinated Build Operation Stage	(1-min(1,'Actor Build-Operation Alignment')) *'Tasks to Coordinate Operation Stage'/'Avg Coord Duration Commission Stage'
Tasks Not Coordinated Design Build Stage	(1-min(1,'Actor Design-Build Alignment'))*'Tasks to Coordinate Build Stage'/'Avg Coord Duration Build Stage'
Tasks Released Design Stage	For (i=NofTasks|0)
Tasks Released Build Stage	For (i=NofTasks|0)
Tasks Released Operation Stage	For (i=NofTasks|0)
Tasks to be Iterated Build Stage	For (i=NofTasks|0)
Tasks to be Iterated Design Stage	For (i=NofTasks|0)
Tasks to be Iterated Operation Stage	For (i=NofTasks|0)
Tasks to Build Stage	'Tasks Released Design Stage'/'Release Task Delay Design Stage'
Tasks to Coordinate Build Stage	For (i=NofTasks|0)
Tasks to Coordinate Operation Stage	For (i=NofTasks|0)
Tasks to Operation Stage	'Tasks Released Build Stage'/'Release Task Delay Build Stage'
Tasks to Project Completion	'Tasks Released Operation Stage'/'Release Task Delay Operation Stage'
Time to Reduce Work Capacity	14<<mo>>
Time value	Number(Time - StartTime-'Actor Entry Timing')/'Deadlines in Months'
Total Project Tasks Scope	For (i=NofTasks|100)
Total Remaining Tasks	{ArrSum('Remaining Tasks Design Stage'),ArrSum('Remaining Tasks Build Stage'),ArrSum('Remaining Tasks Commission Stage')}
Total Undiscovered Defects post Commission	ArrSum('Undiscovered Defects Post Commission Stage')
Undiscovered Defects Build Stage	For (I=NofDefectsType|0)
Undiscovered Defects Design Stage	For (I=NofDefectsType|0)
Undiscovered Defects Operation Stage	For (I=NofDefectsType|0)
Undiscovered Defects Post Commission Stage	For (I=NofDefectsType|0)
v	0.1
Work Capacity Adjustment Time	0.25<<mo>>
Work Done Build Stage	0
Work Done Design stage	0
Work Done Operation Stage	0

Appendix B Case estimates

Expert 1: Stage Contribution to End Building Quality

	Conceptual Design			Detailed Design & Build			Operation		
Issues	Min	Best	Max	Min	Best	Max	Min	Best	Max
Heating System Efficiency	30%	40%	50%	30%	40%	50%	10%	20%	30%
Lighting energy use	30%	40%	50%	30%	40%	50%	10%	20%	30%
Office equipment power density	20%	25%	30%	20%	25%	30%	40%	50%	60%
Occupant density	20%	25%	30%	20%	25%	30%	40%	50%	60%
Heating Set point	40%	50%	60%	0%	0%	0%	40%	50%	60%
Occupancy hours	20%	25%	30%	20%	25%	30%	40%	50%	60%
Infiltration[1]									
Ventilation	10%	20%	30%	30%	40%	50%	30%	40%	50%

Expert 1: Testing Thoroughness Estimate

	Conceptual Design			Detailed Design & Build			Operation		
Issues	Min	Best	Max	Min	Best	Max	Min	Best	Max
Heating System Efficiency	40%	50%	60%	20%	30%	40%	70%	80%	90%
Lighting energy use	20%	30%	40%	20%	30%	40%	70%	80%	90%
Office equipment power density	40%	50%	60%	0%	0%	0%	40%	50%	60%
Occupant density	40%	50%	60%	0%	0%	0%	40%	50%	60%
Heating Set point	40%	50%	60%	0%	0%	0%	40%	50%	60%
Occupancy hours	40%	50%	60%	0%	0%	0%	40%	50%	60%
Infiltration[2]									
Ventilation	40%	50%	60%	20%	30%	40%	40%	50%	60%

[1] The expert did not provide an estimate for infiltration because he did not have access to the calibrated model for the case building.

[2] The expert did not provide an estimate for infiltration because he did not have access to the calibrated model for the case building.

Expert 2 Stage Contribution to End Building Quality

	Conceptual Design			Detailed Design & Build			Operation		
	Min	Best	Max	Min	Best	Max	Min	Best	Max
Heating System Efficiency	20%	30%	35%	25%	30%	40%	30%	40%	50%
Lighting power density	45%	50%	55%	0%	0%	0%	45%	50%	65%
Office equipment power density	45%	50%	55%	0%	0%	0%	45%	50%	65%
Occupant density	45%	50%	65%	0%	0%	0%	45%	50%	55%
Heating Set point	45%	50%	55%	0%	0%	0%	45%	50%	65%
Occupancy hours	45%	50%	65%	0%	0%	0%	45%	50%	55%
Infiltration	40%	40%	60%	40%	40%	60%	10%	20%	20%
Ventilation	5%	10%	15%	15%	20%	25%	60%	70%	90%

Expert 2: Testing Thoroughness Estimate

	Conceptual Design			Detailed Design & Build			Operation		
	Min	Best	Max	Min	Best	Max	Min	Best	Max
Heating System Efficiency	30%	45%	60%	10%	25%	40%	20%	25%	30%
Lighting power density	50%	60%	70%	0%	0%	0%	10%	25%	40%
Office equipment power density	30%	50%	70%	0%	0%	0%	10%	25%	40%
Occupant density	20%	35%	50%	0%	0%	0%	0%	0%	0%
Heating Set point	20%	40%	60%	0%	0%	0%	20%	40%	60%
Occupancy hours	20%	35%	50%	0%	0%	0%	0%	0%	0%
Infiltration	30%	45%	60%	30%	40%	50%	10%	25%	40%
Ventilation	40%	55%	70%	40%	55%	70%	10%	25%	40%

Appendix C Model Calibration

Model setup: Initial Alignment Values

The variables setting the initial levels of alignment are *Initial Level of Inter Stage Alignment* and *Initial Level of Intra Stage Alignment*. These are set to a value of 0.5 based on interviews but also on consideration of what the extremes of 0 and 1 represent in reality. 0 represents a state of total fragmentation in the construction chain where project partners are willing only to do the bare minimum to deliver the project. Clearly this is not representative of the conditions of the case project building. The importance of the project to all the involved project partners was established through the interviews and the workshop. 1 represents a state of alignment with a minimum fragmentation. In effect the project partners operate in a single organization from project start to finish. They share the same organizational routines and procedures and as such there is a common perception of problem solving related knowledge. This is also not representative of the conditions of the case project building. Project partners organizations had previously collaborated in different projects. Nevertheless, as is common practice in project based organizations, the make up of teams assigned to projects is varied both to diffuse knowledge in organizations, due to turnover, and seniority. Thus, the alignment values chosen for the initial level variables was 0.5. This was varied to explore the effect of alignment on building project quality. Results are presented and discussed in the paper.

Logistic Curve Calibration

The general form of the logistic curve is given by

$$Y_t = A + \frac{K - A}{(1 + Qe^{-B(t-M)})^{1/v}}$$

Where K is the upper asymptote, A is the lower asymptote, v affects near which asymptote maximum growth occurs, M is the starting time, B is the growth rate. The values given to B: {1, 1.2, 1.4} based on expert estimates on the shape of the curves. The rest of the variables were given the following values: Q = 100, N=1, K=1, A=0, M=0 for all three curves. Each curve is shifted in time so that it starts at the project stage start (Figure 1).

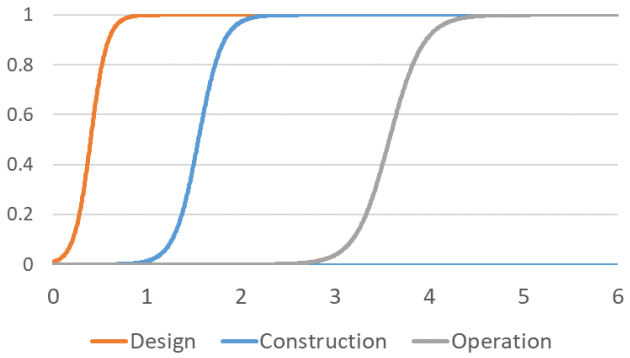

Figure 1 S-curves implemented in SD model

Knock-on effects between stages

The effect of defects in upstream project stages on quality of work in downstream stages is important in construction projects. In order to account for this effect between design and build stages and between build and operation stages two variables are used: *Strength of Defect Effect Design Stage* and *Strength of Defect Effect Build Stage*. They represent the effect of defects in a stage on downstream work. They are modelled with an 9-element array to account for the 9 performance areas in the case building project. These two variables are used to further calibrate the SD model using the powersim solver. The objective is to minimize the error between the model

output from the *Absolute Deviation per Defect Type* variable and the actual performance gap documented in Jain et al. (2017).

Optimization set up

The optimization was carried out with powersim evolutionary based search algorithm with the following settings: generations 300, 20 parents, 100 offsprings, minimum convergence 0.001, seed 100. The average of the best estimates of building performance experts are used as inputs for the following variables:
Defect Testing Thoroughness Design Stage, Defect Testing Thoroughness Build Stage, Defect Testing Thoroughness Operation Stage, Contribution of Intra Stage Defect Design Stage, Contribution of Intra Stage Defect Build Stage, Contribution of Intra Stage Defect Operation Stage

The optimization algorithm is set with two variables as the decision variables: *Strength of Defect Effect Design Stage* and *Strength of Defect Effect Build Stage*. Their value range was {2..22}.

Optimization output

The output values of the optimization for *Strength of Defect Effect Design Stage* are:
{7.923475175386, 7.681473816419, 6.454101033645, 4.299443377058, 15.80109170805, 4.67366321947, 2.077952670154, 4.814456028285, 5.444649198402}

The output values of the optimization for *Strength of Defect Effect Build Stage* are:
{10.71257953759, 1.959047860853, 14.88852276293, 3.606528720739, 13.67199484315, 4.493698498603, 1.524016193645, 12.60935922408, 2.740817243553}

These values give the following array values for the *Absolute Deviation per Defect Type* with a timestep of 0.125 months and simulation time of 5 years with Euler integration, and a bank calendar setting:
{0.02052034980391, 0.002039130274223, 0.001075194932733, 0.00006836309092473, 0.01827165989677, 0.006200356094376, 0.0004277520084877, 0.00001543118211078, 0.00}

The average error is 0.005402026364837.

Appendix Model Documentation
Task flow structure
Initial Completion Rate Design Stage
min('Tasks Not Completed Design Stage'/'Avg Initial Completion Duration Design Stage',
'Actor Work Capacity'[1], 'Tasks Available for Completion Design Stage'/'Avg Initial Completion Duration Design Stage')

Logic: Project work is completed subject to capacity constraints. This is provided that there are tasks available for completion. This logic applies also to *Initial Completion Rate Build Stage* and *Initial Completion Rate Operation Stage*.

Discover Intra Stage Defects Design Stage
min('Quality Assurance Design Stage','Tasks Completed Not Checked Design Stage'
'Probability of Intra Defect Discovery Design Stage''Probability of Intra Stage Defect Design Stage')/1<<mo>>)

Logic: The rate of discovering intra stage defects is the min of quality assurance capacity, or the available tasks completed times the probability of discovering defects when there is a defect in the stage. This is influenced by product of the probability of defect discovery and the probability of generating a defect in the stage in the first place.This logic applies also to *Initial Completion Rate Build Stage* and *Initial Completion Rate Operation Stage*.

Discover Intra Stage Defects Build Stage
min('Quality Assurance Build Stage','Tasks Completed not Checked Build Stage'/1<<mo>>
'Probability of Intra Defect Discovery Build Stage''Probability Intra Stage Defect Build Stage')
+('Tasks Completed not Checked Build Stage'*
'Proportion of Design Defects Possible to Rework in Build Stage'
*'Probability of Defects Design to Build Stage'
*'Probability of Intra Defect Discovery Build Stage'/1<<mo>>)

Logic: The rate of discovering intra stage defects is the min of quality assurance capacity, or the available tasks completed times the probability of discovering defects when there is a defect in the stage. The difference to design stage is that there is a proportion of design defects that is not possible to rework in the build stage. This is what the plus term accounts for. This logic is the same for *Discover Intra Stage Defects Operation Stage*.

Change Task Rate Design Stage
min('Actor Work Capacity'[1]-'Initial Completion Rate Design Stage' ,
'Tasks to be Iterated Design Stage'/'Avg Iteration Duration Design Stage')

Logic: Work on new tasks gets precedence over rework. Rework is done only if there is surplus of capacity. The same logic applies to *Change Task Rate Build Stage* and *Change Task Rate Commission Stage*.

Quality Assurance Design Stage
max(0/1<<mo>>,'Actor Work Capacity'[1]-'Initial Completion Rate Design Stage')

Logic: Quality assurance is used for testing tasks. It is assumed that total actor capacity can be allocated to task completion or testing of competed tasks. It is equal to total capacity-capacity used to complete tasks. The same logic applies to *Quality Assurance Build Stage* and *Quality Assurance Commission Stage*.

Discover Inter Stage Defects Build Stage

min('Tasks Completed not Checked Build Stage'/1<<mo>>,
'Quality Assurance Build Stage'*max(0,('Probability of Generating Inter Stage Defect in Design Stage'-'Probability Intra Stage Defect Build Stage')))
*(1-'Proportion of Design Defects Possible to Rework in Build Stage')

Logic: It is assumed that developers will correct defects generated within their own phase before bringing those tasks to coordination so as to avoid revealing to colleagues that they have made errors. This is based on the assumption that they are not willing to accept work of a lower quality level than what they are willing to deliver. That is *if* 'Probability of Inter Stage Defect in Design Stage' <'Probability Intra Stage Defect Build Stage' *then* all defects are kept in the stage. The rate at which this is done is subject to quality assurance capacity constraints. Here priority is on quality assurance for intra stage task testing. The same logic applies to *Discover Inter Stage Defects Commission Stage*.

Approve Tasks Rate Design Stage

min('Tasks Completed Not Checked Design Stage'/1<<mo>>,
('Quality Assurance Design Stage'-'Discover Intra Stage Defects Design Stage')
*(1-'Probability of Intra Defect Discovery Design Stage'))
+'Defect Correction Rate Design Stage'

Logic: The tasks that are approved are the minimum of those completed and checked in quality assurance and no defects were found times the probability of not finding defects.. the plus term is there to account for the extreme that all tasks are defective and the approval rate then is the correction rate.

Release Task Rate Design Stage

'Tasks Approved Design Stage'*'Release Time Design Stage'/'Release Task Delay Design Stage'

Logic: the logic is that tasks are released to subsequent stages at an appropriate release time. The same logic applies to *Release Task Rate Build Stage* and *Release Task Rate Operation Stage*.

Release Time Design Stage

If ('Release Package Size Design Stage'/('Scope Design Stage'-'Tasks Released Design Stage')
<='Tasks Approved Design Stage',1,0)

Logic: This functions like a switch. Tasks in project are released in packages e.g. in software projects. The *Release Package Size Design Stage* is a constant. The logic is that there must be enough tasks approved so that their number is higher than the first part of the if function. The same logic applies to *Release Time Build Stage* and *Release Time Commission Stage*.

Tasks Coordinate Design Stage Build Stage

min(1,'Actor Design-Build Alignment')*'Tasks to Coordinate Build Stage'/'Avg Coord Duration Build Stage'*(1-'Project Changes Difficulty'[1])

Logic: Alignment functions as a switch. With no alignment no tasks are returned to previous stages. Coordination takes place as long as the 'Project Changes Difficulty' remains less than one. This variable is implemented with a general logistic function. See calibration for details.

The same logic applies to *Tasks Coordinate Build Stage Commission Stage* and *Coordinate Operation Post Commision*.

Tasks Not Coordinated Design Build Stage
(1-min(1,'Actor Design-Build Alignment'))*'Tasks to Coordinate Build Stage'/'Avg Coord Duration Build Stage'

Logic: This is the complementary variable to *Tasks Coordinate Design Stage Build Stage*. Alignment functions as a switch. With no alignment between CSC partners all tasks send downstream. The same logic applies to *Tasks Not Coordinated Build Operation Stage*.

Fraction Perceived Satisfactory Design Stage
'Net Work Done Design Stage'/'Scope Design Stage'

Logic: the stock of net work done is tasks send to build stage minus returns of defects. The same logic applies to *Fraction Perceived Satisfactory Build Stage* and *Fraction Perceived Satisfactory Operation Stage*.

% Avail Internal Concurrence Design Stage
For (i=NofTasks|Graph('Fraction Perceived Satisfactory Design Stage'[i],0,1,
'Internal Process Concurrence Relation Graph Input Design Stage'[i]))

Logic: This is taken directly from Ford and Sterman 1998. The internal process concurrence relation capture the degree of sequentiality or concurrence of the tasks aggregated together within a phase, including possible changes in the degree of concurrence as the work progresses. The same logic applies to *% Avail Internal Concurrence Build Stage* and *% Avail Internal Concurrence Operation Stage*.

Tasks Available for Completion Design Stage
'Scope Design Stage'*min('% Avail Internal Concurrence Design Stage','% Available External Concurrence Design Stage')

Logic: The tasks available depends on whether additional intra stage work is required to proceed or more upstream tasks have to be delivered first. the same logic applies in *Tasks Available for Completion Build Stage* and *Tasks Available for Completion Operation Stage*.

% Available External Concurrence Design Stage
For (i=NofTasks|Graph('Fraction of Tasks Released from Design Stage'[i]
,0,1,'External Process Concurrence Relation Design Stage'[i]))

Logic: This is from Ford and Sterman (1998), meant to capture the work overlap relation between project phases. The same logic applies to *% Avail External Concurrence Build Stage* and *% Avail External Concurrence Operation Stage*.

Fraction of Tasks Released from Design Stage
'Tasks Not Completed Design Stage'/'Scope Design Stage'

Logic: the fraction is a function of the total work scope in design stage. The same logic applies in *Fraction of Tasks Released from Build Stage* and *Fraction of Tasks Released from Operation Stage*.

Defect flow structure
Generate Internal Defects in Completion Design Stage
'Initial Completion Rate Design Stage'*'Probability of Intra Stage Defect Design Stage'
*(1-'Actor Inter Stage Cumulative Communication'[1]/'Total Actor Tasks')

Logic: This is the internal defect generation rate for each stage. It is assumed that additional information processing between stages results in less defects being generated in a stage. the same logic applies to *Generate Internal Defects in Initial Completion Build Stage* and *Generate Internal Defects in Initial Completion Commission Stage*.

Defects Released Design Stage

max(0/1<<mo>>, min('Approve Tasks Rate Design Stage',
'Undiscovered Defects Design Stage'/1<<mo>>-'Discover Internal Defects Design Stage'))

Logic: The logic is that no defective tasks are released, if defective tasks discovered equal the number of undiscovered defects. The same logic applies to *Defects Released Build Stage* and *Approve Defects Operation Stage*.

Defects to Build Stage

'Defect Knock on Effect Design Stage' *('Defects Released Design Stage'+'Defects Released with Workarounds to Build Stage')

Logic: This flow is separate from *Defects Released Design Stage*. This enables to account for knock-on effects of defects in downstream stages using 'Defect Knock on Effect Design Stage'. The same logic applies to *Defects to Operation Stage*.

Defects Released with Workarounds to Build Stage

('Known Defects Design Stage'/1<<mo>>-'Defect Correction Rate Design Stage')
*'Project Changes Difficulty'[1]

Logic: This flow enables to account for defects that are treated with workarounds, when the difficulty in implementing changes to the project becomes greater. This is modelled in an s-curve. Defects with workarounds can have knock on effects on subsequent project stages. The same logic applies to *Defects Released with Workarounds to Build Stage*.

Defect Knock on Effect Design Stage

(1+('Undiscovered Defects Design Stage'+'Known Defects Design Stage')/'Scope Design Stage')^'Strength of Defect Effect Design Stage'

Logic: Undiscovered and known defects in a stage have knock on effects of greater magnitude in subsequent project stages. This magnitude is captured by *Strength of Defect Effect Design Stage*. The same logic applies to *Defect Knock on Effect Build Stage*.

Discover Internal Defects Design Stage

min('Undiscovered Defects Design Stage'/1<<mo>>,
'Discover Intra Stage Defects Design Stage')*'Probability of Intra Defect Discovery Design Stage'

Logic: This is the rate of discovering defects in each stage. Defects discovered in the stage are the minimum of the defects that exist or the probability of discovering defects in the stage. the same logic applies to *Discover Internal Defects Build Stage* and *Discover Internal Defects Operation Stage*.

Probability of Intra Defect Discovery Design Stage

min(1,'Initial Probability of Intra Defect Discovery Design Stage'
*('Initial Probability of Intra Defect Discovery Design Stage'
+'Switch Effect of Communication on Defect Detection'*

('Actor Inter Stage Cumulative Communication'[1]*'Actor Intra Stage Cumulative Communication'[1])/'Total Project Tasks Scope'^2)
divz0('Initial Probability of Intra Defect Discovery Design Stage'))

Logic: The probability of discovering defects depends on cumulative inter and intra actor communication. The division with total actor tasks is made to keep probabilities between 0 and 1. It is assumed that a greater number of tasks requires a greater amount of information to ensure minimum defects occurring in any project stage. The same logic applies to *Probability of Intra Defect Discovery Build Stage* and *Probability of Intra Defect Discovery Operation Stage*.

Defect Correction Rate Design Stage

'Known Defects Design Stage' divz0 'Tasks to be Iterated Design Stage'
*(1-('Probability of Generating an Additional Internal Defect when Receiving Downstream Tasks Design Stage'))
'Change Task Rate Design Stage'(1-'Project Changes Difficulty'[1])

Logic: Defects are corrected as long as they are discovered, there are still tasks to be iterated, and the *Project Changes Difficulty* is less than 1. The correcting rate also depends on no additional defects being generated. The same logic applies to *Correcting Defects Build Stage* and *Correcting Defects Operation Stage*.

Generate Defects in Iteration Design Stage

'Known Defects Design Stage' divz0 'Tasks to be Iterated Design Stage'
*'Probability of Generating an Internal Defect in Iteration Design Stage'
*'Change Task Rate Design Stage'

Logic: The rate of defect generation in iteration, when trying to fix discovered defects. The divz0 is used as there are no *Tasks to be Iterated Design Stage* the defect generation rate must be zero. The same logic applies to *Generate Defects in Iteration Build Stage* and *Generate Defects in Iteration Operation Stage*.

Probability of Generating an Internal Defect in Iteration Design Stage

max(0,'Initial Probability of Generating an Internal Defect in Iteration Design Stage'
-('Actor Intra Stage Cumulative Communication'[1]
+'Actor Inter Stage Cumulative Communication'[1])
/'Total Actor Tasks')

Logic: That additional intra and inter stage information should decrease errors in iteration in the design stage. the same logic applies to *Probability of Generating an Internal Defect in Iteration Build Stage* and *Probability of Generating an Internal Defect in Iteration Operation Stage*.

Probability of Generating Additional Internal Defect when Receiving Downstream Tasks Design Stage

max(0,'Initial Probability of Generating an Additional Internal Defect when Receiving Downstream Tasks Design Stage'
-'Actor Inter Stage Cumulative Communication'[1]/'Total Actor Tasks')

Logic: The probability of correcting known defects is one unless additional defects are generated. The logic is that the greater the number of total actor tasks, the greater the amount of information exchange required to have an impact on this probability. The same logic applies to *Probability of Generating Additional Internal Defect when Receiving Downstream Tasks Build Stage* and

Probability of Generating Additional Internal Defect when Receiving Downstream Tasks Operation Stage.

Discover Inter Stage Defects in Build Stage

min('Tasks Completed not Checked Build Stage'/1<<mo>>,
('Quality Assurance Build Stage'-'Discover IntraStage Defects Build Stage')
*max(0,('Probability of Defects Design to Build Stage'
-'Probability Intra Stage Defect Build Stage')*'Probability of Intra Defect Discovery Build Stage'))
*(1-'Proportion of Design Defects Possible to Rework in Build Stage')

Logic: This flow concerns the defects that are discovered in build stage but have not been generated or developed in these stages. This is the reason for the quality assurance – discover intra stage defects term. This term is multiplied by a probability term which is the probability of defects coming from design stage minus the intra staerge defect probability times the probability of discovering this. This term is multiplied by the proportion of defects that must be sent back to design stage. The same logic applies to *Discover Inter Stage Defects in Operation Stage*.

Return Defects to Design Stage

'Discover Inter Stage Defects in Build Stage'/'Defect Knock on Effect Design Stage'

Logic: It is necessary to have a separate flow to return defects to the actual stock of known defects in design. This is because defect number in build stage is inflated. E.g. a design defect might generate 2 build defects, but if it is returned to design stages it must still be a single design defect. This is the reason to divide with Defect Knock on Effect Build Stage. the same logic applies to *Return Defects to Build Stage*.

Capacity Structure

Capacity Increase

'Additional Required Capacity'
*For(i=NofActors|Step(1,StartTime+'Actor Entry Timing'[i]))
*min(1,'Relative Schedule Pressure')
/'Work Capacity Adjustment Time'

Logic: Capacity is added at all cost to the project to finish in time and not reduce its scope if required time is less than available time then capacity should be added.

Capacity Decrease

max(0/1<<mo>>, 'Actor Work Capacity'-ArrSum('Net Project Remaining Work Per Stage')/1<<mo>>)/'Time to Reduce Work Capacity'

Logic: If available capacity is in excess of what is required to complete the remaining tasks then it is reduced. So when the project ends, capacity reduces to zero.

Additional Required Capacity

For (i=NofActors|ArrSum('Net Project Remaining Work Per Stage'[i]))
*{'Avg Initial Completion Duration Design Stage','Avg Initial Completion Duration Build Stage','Avg Initial Completion Duration Commission Stage'}
divz0('Individual Actor Available Time to Completion')/1<<mo>>

Logic: Calculates the required capacity per stage. The assumption is that all of the tasks need to be worked on in every stage.

Net Project Remaining Work Per Stage
{'Total Actor Tasks'-'Net Work Done Design Stage',
'Total Actor Tasks'-'Net Work Done Build Stage',
'Total Actor Tasks'-'Net Work Done Commission Stage'}

Logic: Calculates the remaining tasks for each stage. when they are zero then capacity should also go to zero. The assumption is that all of the tasks need to be worked on in every stage.

Relative Schedule Pressure
'Actor Schedule Pressure'divz0 'Initial Schedule Pressure'

Logic: It is assumed that an initial pressure to perform exists.

Increase in Schedule Pressure
(max(0,('Required Time to Completion'divz0'Individual Actor Available Time to Completion')
*('Sensitivity to Time Pressure')))/'Schedule Pressure Adjustment Delay'
*For(i=NofActors|Step(1,StartTime+'Actor Entry Timing'[i]))

Logic: As available time to completion decreases, pressure increases if there is a change in deadline then time to completion increases and momentarily pressure should not increase or even decrease.

Decrease in Schedule Pressure
max(0/1<<mo>>,'Actor Schedule Pressure'/('Actor Schedule Pressure Release Delay')
*For (I=1..3|(1-'Total Remaining Tasks'[i]/ArrSum('Total Actor Tasks'))))

Logic: The logic of the 1st line is it is harder to reduce pressure when there are a lot of remaining tasks. The logic of 3rd line is assuming that there is always less time to complete all the required tasks it is harder to reduce pressure if there are a lot of remaining tasks.

Total Remaining Tasks
{ArrSum('Remaining Tasks Design Stage'),ArrSum('Remaining Tasks Build Stage'),ArrSum('Remaining Tasks Commission Stage')}

Logic: It calculates the sum of remaining tasks per project stage.

Remaining Tasks Design Stage
'Total Actor Tasks'-'Net Work Done Design Stage'

Logic: It calculates the remaining tasks in a stage. the same logic applies to *Remaining Tasks Build Stage* and *Remaining Tasks Commission Stage*.

Required Time to Completion
max(0<<mo>>,
{ArrSum('Remaining Tasks Design Stage')*'Avg Initial Completion Duration Design Stage'
,ArrSum('Remaining Tasks Build Stage')*'Avg Initial Completion Duration Build Stage'
,ArrSum('Remaining Tasks Commission Stage')*'Avg Initial Completion Duration Commission Stage'}divz0 Number(max('Additional Required Capacity', 'Actor Work Capacity')))

Logic: The required time to complete all tasks in a stage is number of tasks over time required to complete one unit of task. The completion duration is brought in the equation to account for different task processing times in design, build, commission.

Actor Project Participation
For(I=NofActors|(Step(1,StartTime+'Actor Entry Timing'[i])-Step(1,StartTime+'Individual Calendar Deadlines'[i])))*(min(1,{ArrSum('Total Actor Tasks'-'Net Work Done Design Stage')
,ArrSum('Total Actor Tasks'-'Net Work Done Build Stage')
,ArrSum('Total Actor Tasks'-'Net Work Done Commission Stage')}))

Logic: Actor participation takes as long as it is required to complete the tasks per stage. The assumption is that as rework reduces network done, actors spend more time to fix defects.

Actor Alignment Increase
For(i=NofActors,j=NofActors|'Actor Project Net Increase in Overlap'[i,j]
divz0'Actor Project Participation Duration'[i])*1<<mo>>

Logic: It is assumed that pressure and actor overlap brings the actors into alignment. This variable calculates a matrix of actor alignment.

Actor Alignment Decrease
For (i=NofActors, j=NofActors|max(0,'Actor Alignment'[i,j])/'Individual Actor Deadlines'[i])

Logic: It is assumed that as opportunity cost, i.e. other projects available in the building sector raise the pressure, alignment decreases with time.

Actor Project Interaction Rate Increase
For(I=NofActors|(Step(1,StartTime+'Actor Project Interaction Timing'[i])
-Step(1,StartTime+'Individual Calendar Deadlines'[3])))

Logic: This is to allow for project partners to be engaged in the project before they commit to work e.g. start work on site.

Actor Design -Build Communication Rate
('Actor Design Communication'[2]*'Com Priority at Design Stage'
'Actor Build Communication'[1]'Com Priority at Build Stage')*'Actor Design-Build Alignment'

Logic: The multiplicative logic is used to model the fact that both actors must reciprocally engage in communication (Bendoly and Swink, 2007). Additional communication on any of the actors is enough to have an effect on the understanding of the work that is being carried out. The same logic applies to *Actor Design-Operation Communication Rate* and *Actor Build-Operation Communication Rate*.

Inter Stage Information Increase
min({'Actor Design -Build Communication Rate','Actor Build-Operation Communication Rate', 'Actor Design-Operation Communication Rate', ('Max Info per Kind of Task'-'Actor Inter Stage Cumulative Communication'))/'Information Increase Delay'

Logic: It is assumed that a maximum information quantity per kind of project task is required to complete the task without any defects.

Information Decrease
For (i=NofActors| 'Actor Inter Stage Cumulative Communication'[i]/'Individual Actor Deadlines'[i])

Logic: The logic is that information decline is proportional to the participation of each actor. It takes a time period of participation to decrease information by about 66.6%.

Intra Stage Information Increase

min({'Actor Communication Design Stage','Actor Communication Construction Stage'
,'Actor Communication Operation Stage'}
,('Max Info per Kind of Task'-'Actor Intra Stage Cumulative Communication'))
/'Information Increase Delay'

Logic: It is assumed that a maximum information quantity per kind of project task is required to complete the task without any defects.

Intra Stage Information Decrease

min('Actor Intra Stage Cumulative Communication','Actor Intra Stage Cumulative Communication'
*max(1,{'Actor Alignment Design Stage','Actor Alignment Construction Stage','Actor Alignment Operation Stage'}))/'Individual Actor Deadlines'

Logic: It is assumed that information decline is proportional to the participation of each actor.
It takes a time period of participation to decrease information by about 66.6%

Appendix D Project Timeline

Considerable time was spent in preplanning and approval according to the client. The project duration from start to finish was approximately 2.5 years. Figure 1 illustrates some of the main milestones of the project. The project started with pre-application submission. The scheme design was completed 10 months later and then the project entered the tender stage. The detailed design, value engineering and actual construction of the building were all concurrent activities with considerable overlap until the final completion of the building 33 months later. The finished building went through basic/seasonal commissioning, for two years and the two-year post commission period has been already completed as of January 2017.