

Conclusions and Future Work

- Proposed an approach to solve the RCPSP with logical constraints:
 - OR precedence relations;
 - Bi-directional precedence relations;
- The method uses an extension of multi-mode, where the realization of each activity-mode is controlled by SAT, with more flexibility over the MRCPSP procedures, and the schedule is done by a RCPSP procedure
- The method, applied to MRCPSP instances, reveal to be competitive.
- Future work we plan:
 - Study these type of logical constraints without resources, and with multiple mode;
 - Study mode identity constraints in multiple mode.

A novel approach to solve various resource-constrained project scheduling problems using multi-mode

José Coelho* and Mario Vanhoucke**

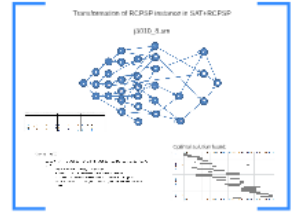
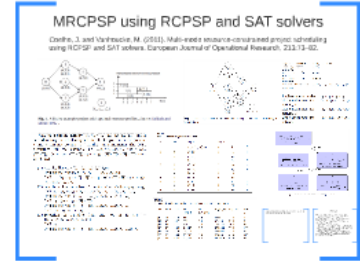
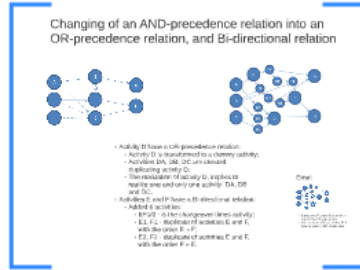
* Technical University of Lisbon, Lisbon (Portugal)
 ** Universidade Alberta, Lethbridge (Portugal) José.Coelho@ualb.ca
 *** Faculty of Economics and Business Administration, Ghent University, (Belgium) mario.vanhoucke@ugent.be
 **** Technology and Operations Management Area, Vlerick Business School (Belgium) mario.vanhoucke@vlerick.com
 ***** Department of Management Science and Innovation, University College London (United Kingdom) m.vanhoucke@ucl.ac.uk

Using MRCPSP instances to create OR-precedence relations and Bi-directional relations instances.

In a MRCPSP with RCPSP resources:

1. Select a number of activities, a part of a MRCPSP instance, or a MRCPSP instance.
2. Select a number of activities with AND precedence relations, and generate an OR precedence relation between them.
3. Select a number of activities with AND precedence relations, and generate a Bi-directional precedence relation between them.

Also, I observed that any activity, as a part of a MRCPSP instance, can be used as a resource for other activities, if the activity is not used before the project starts. This is a new type of precedence relation, and it is called Bi-directional precedence relation.



Tests and Results

- Run without CR-precedence relations and Bi-directional relations:
 - Check if our procedure is close to the state-of-the-art in MRCPSP instance sets
- Run with different density of OR-precedence relations:
 - Quantify the increase needed on the number of activities
 - Quantify the gain in makespan by consider OR-precedence relations, relative to treat those precedence relations as AND-precedence relations
- Run with bi-directional relations.

***A novel approach to solve
various resource-constrained
project scheduling problems***

using multi-mode

José Coelho* and Mario Vanhoucke**

* - Technical University of Lisbon, Lisbon (Portugal)

- Universidade Aberta, Lisbon (Portugal) Jose.Coelho@uab.pt

** - Faculty of Economics and Business Administration, Ghent University,
(Belgium) mario.vanhoucke@ugent.be

- Technology and Operations Management Area, Vlerick Business School
(Belgium) mario.vanhoucke@vlerick.com

- Department of Management Science and Innovation, University College
London (United Kingdom) m.vanhoucke@ucl.ac.uk

MRCPSP using RCPSP and SAT solvers

Coelho, J. and Vanhoucke, M. (2011). Multi-mode resource-constrained project scheduling using RCPSP and SAT solvers. *European Journal of Operational Research*, 213:73–82.

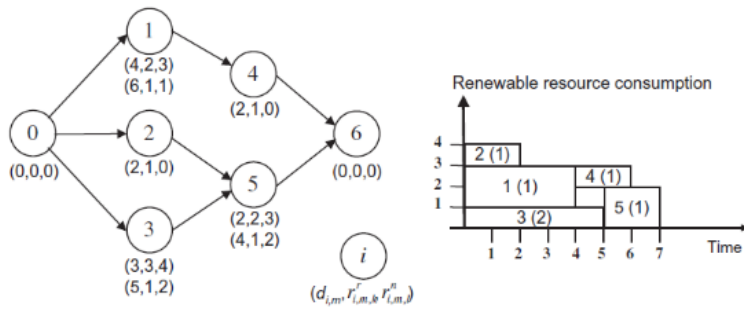


Fig. 1. A fictive example project with optimal resource profiles.. Source: Kolisch and Drexel (1997).

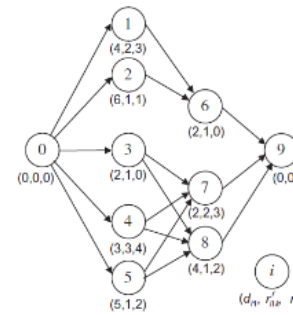


Fig. 2. The single-mode RCPSP network of Fig. 1 without any activity mode restrictions.

Single-mode activities: $x_0 + x_3 + x_6 + x_9 = 4$
 Mode assignment for activity 1: $x_1 + x_2 = 1$
 Mode assignment for activity 3: $x_4 + x_5 = 1$
 Mode assignment for activity 5: $x_7 + x_8 = 1$
 Non-renewable resource constraint: $3x_1 + x_2 + 4x_4 + 2x_5 + 3x_7 + 2x_8 \leq 8$

Single-mode activities: $x_0 \wedge x_3 \wedge x_6 \wedge x_9$
 Mode assignment for activity 1: $(x_1 \vee x_2) \wedge (\bar{x}_1 \vee \bar{x}_2)$
 Mode assignment for activity 3: $(x_4 \vee x_5) \wedge (\bar{x}_4 \vee \bar{x}_5)$
 Mode assignment for activity 5: $(x_7 \vee x_8) \wedge (\bar{x}_7 \vee \bar{x}_8)$

Assume a simple activity list $AL = \{0, 1, 2, 3, 4, 5, 6, 7, 8, 9\}$ and a set T denoting the set of assigned literals at a given moment (where x and \bar{x} can not belong to T at the same moment). The DPLL algorithm for the example $CNF = x_0 \wedge x_3 \wedge x_6 \wedge x_9 \wedge (x_1 \vee x_2) \wedge (\bar{x}_1 \vee \bar{x}_2) \wedge (x_4 \vee x_5) \wedge (\bar{x}_4 \vee \bar{x}_5) \wedge (x_7 \vee x_8) \wedge (\bar{x}_7 \vee \bar{x}_8) \wedge (\bar{x}_1 \vee \bar{x}_4)$ runs as follows:

- Activity list $AL = \{0, 1, 2, 3, 4, 5, 6, 7, 8, 9\}$.
 Unit clause rule: $x_0, x_3, x_6, x_9: T = \{x_0, x_3, x_6, x_9\}$.
 CNF: $(x_1 \vee x_2) \wedge (\bar{x}_1 \vee \bar{x}_2) \wedge (x_4 \vee x_5) \wedge (\bar{x}_4 \vee \bar{x}_5) \wedge (x_7 \vee x_8) \wedge (\bar{x}_7 \vee \bar{x}_8) \wedge (\bar{x}_1 \vee \bar{x}_4)$.
- Selection of variable from AL , not in T : $x_1: T = \{x_0, x_3, x_6, x_9, x_1\}$.
 CNF: $\bar{x}_2 \wedge (x_4 \vee x_5) \wedge (\bar{x}_4 \vee \bar{x}_5) \wedge (x_7 \vee x_8) \wedge (\bar{x}_7 \vee \bar{x}_8) \wedge \bar{x}_4$.
 Unit clause rule: $\bar{x}_2, \bar{x}_4: T = \{x_0, x_3, x_6, x_9, x_1, \bar{x}_2, \bar{x}_4\}$.
 CNF: $x_5 \wedge (x_7 \vee x_8) \wedge (\bar{x}_7 \vee \bar{x}_8)$.
 Unit clause rule: $x_5: T = \{x_0, x_3, x_6, x_9, x_1, \bar{x}_2, \bar{x}_4, x_5\}$.
 CNF: $(x_7 \vee x_8) \wedge (\bar{x}_7 \vee \bar{x}_8)$.
- Selection of variable from AL , not in T : $x_7: T = \{x_0, x_3, x_6, x_9, x_1, \bar{x}_2, \bar{x}_4, x_5, x_7\}$.
 CNF: \bar{x}_8 .
 Unit clause rule: $\bar{x}_8: T = \{x_0, x_3, x_6, x_9, x_1, \bar{x}_2, \bar{x}_4, x_5, x_7, \bar{x}_8\}$.

Table 1
The SAT mode assignment solution.

i	m	Fig. 1 $(d_{i,m}, r_{i,m,1}^r, r_{i,m,2}^r, r_{i,m,3}^r)$	Fig. 2 i	T	d_{i1}
0	1	(0,0,0)	0	x_0	0
1	1	(4,2,3)	1	x_1	4
1	2	(6,1,1)	2	\bar{x}_2	0
2	1	(2,1,0)	3	x_3	2
3	1	(3,3,4)	4	\bar{x}_4	0
3	2	(5,1,2)	5	x_5	5
4	1	(2,1,0)	6	x_6	2
5	1	(2,2,3)	7	x_7	2
5	2	(4,1,2)	8	\bar{x}_8	0
6	1	(0,0,0)	9	x_9	0

Table 2
A comparison between the pure SAT and the SAT(k) approach.

Set	#Var	#Ins	M (SAT(3))	M (SAT(4))	Avg.Cl(3)	Avg.Cl(4)	#Ins-M
J10	32	536	119	1.5	5883	21.4	0
J12	38	547	801	2.1	32,435	25.4	0
J14	44	551	5342	2.4	186,436	29.4	24
J16	50	550	14,445	2.7	454,055	33.4	162
J18	56	552	23,112	3.0	653,028	37.4	320
J20	62	554	26,759	3.4	694,061	41.4	378
J30	92	640	49,554	5.8	750,030	61.5	480

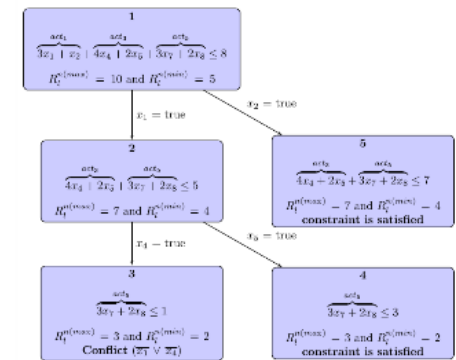


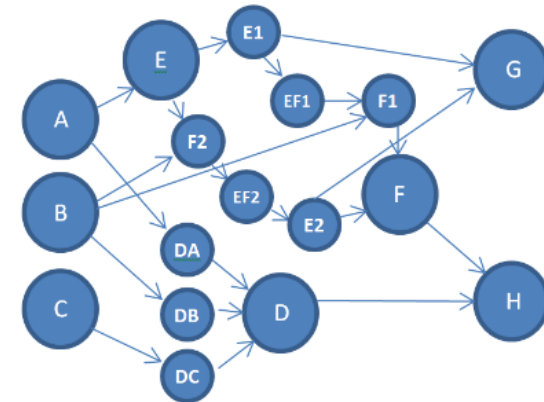
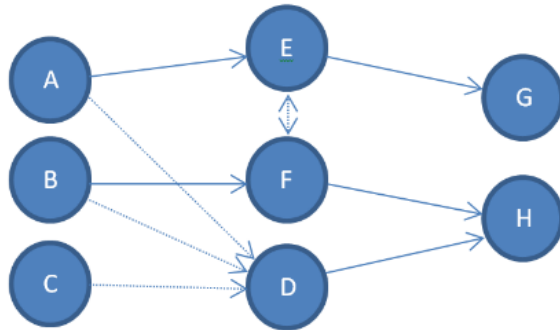
Fig. 3. Enumeration scheme of the example project.

Appendix A. A simple SAT instance with a conflict generation, backtracking step and learning clause

```

c1: x1 v x2
c2: x1 v x3
c3: x2 v x3
c4: x1 v x4
c5: x2 v x4
c6: x3 v x4
c7: x1 v x5
c8: x2 v x5
c9: x3 v x5
c10: x4 v x5
c11: x1 v x6
c12: x2 v x6
c13: x3 v x6
c14: x4 v x6
c15: x5 v x6
c16: x1 v x7
c17: x2 v x7
c18: x3 v x7
c19: x4 v x7
c20: x5 v x7
c21: x1 v x8
c22: x2 v x8
c23: x3 v x8
c24: x4 v x8
c25: x5 v x8
c26: x1 v x9
c27: x2 v x9
c28: x3 v x9
c29: x4 v x9
c30: x5 v x9
c31: x1 v x10
c32: x2 v x10
c33: x3 v x10
c34: x4 v x10
c35: x5 v x10
c36: x1 v x11
c37: x2 v x11
c38: x3 v x11
c39: x4 v x11
c40: x5 v x11
c41: x1 v x12
c42: x2 v x12
c43: x3 v x12
c44: x4 v x12
c45: x5 v x12
c46: x1 v x13
c47: x2 v x13
c48: x3 v x13
c49: x4 v x13
c50: x5 v x13
c51: x1 v x14
c52: x2 v x14
c53: x3 v x14
c54: x4 v x14
c55: x5 v x14
c56: x1 v x15
c57: x2 v x15
c58: x3 v x15
c59: x4 v x15
c60: x5 v x15
c61: x1 v x16
c62: x2 v x16
c63: x3 v x16
c64: x4 v x16
c65: x5 v x16
c66: x1 v x17
c67: x2 v x17
c68: x3 v x17
c69: x4 v x17
c70: x5 v x17
c71: x1 v x18
c72: x2 v x18
c73: x3 v x18
c74: x4 v x18
c75: x5 v x18
c76: x1 v x19
c77: x2 v x19
c78: x3 v x19
c79: x4 v x19
c80: x5 v x19
c81: x1 v x20
c82: x2 v x20
c83: x3 v x20
c84: x4 v x20
c85: x5 v x20
c86: x1 v x21
c87: x2 v x21
c88: x3 v x21
c89: x4 v x21
c90: x5 v x21
c91: x1 v x22
c92: x2 v x22
c93: x3 v x22
c94: x4 v x22
c95: x5 v x22
c96: x1 v x23
c97: x2 v x23
c98: x3 v x23
c99: x4 v x23
c100: x5 v x23
c101: x1 v x24
c102: x2 v x24
c103: x3 v x24
c104: x4 v x24
c105: x5 v x24
c106: x1 v x25
c107: x2 v x25
c108: x3 v x25
c109: x4 v x25
c110: x5 v x25
c111: x1 v x26
c112: x2 v x26
c113: x3 v x26
c114: x4 v x26
c115: x5 v x26
c116: x1 v x27
c117: x2 v x27
c118: x3 v x27
c119: x4 v x27
c120: x5 v x27
c121: x1 v x28
c122: x2 v x28
c123: x3 v x28
c124: x4 v x28
c125: x5 v x28
c126: x1 v x29
c127: x2 v x29
c128: x3 v x29
c129: x4 v x29
c130: x5 v x29
c131: x1 v x30
c132: x2 v x30
c133: x3 v x30
c134: x4 v x30
c135: x5 v x30
c136: x1 v x31
c137: x2 v x31
c138: x3 v x31
c139: x4 v x31
c140: x5 v x31
c141: x1 v x32
c142: x2 v x32
c143: x3 v x32
c144: x4 v x32
c145: x5 v x32
c146: x1 v x33
c147: x2 v x33
c148: x3 v x33
c149: x4 v x33
c150: x5 v x33
c151: x1 v x34
c152: x2 v x34
c153: x3 v x34
c154: x4 v x34
c155: x5 v x34
c156: x1 v x35
c157: x2 v x35
c158: x3 v x35
c159: x4 v x35
c160: x5 v x35
c161: x1 v x36
c162: x2 v x36
c163: x3 v x36
c164: x4 v x36
c165: x5 v x36
c166: x1 v x37
c167: x2 v x37
c168: x3 v x37
c169: x4 v x37
c170: x5 v x37
c171: x1 v x38
c172: x2 v x38
c173: x3 v x38
c174: x4 v x38
c175: x5 v x38
c176: x1 v x39
c177: x2 v x39
c178: x3 v x39
c179: x4 v x39
c180: x5 v x39
c181: x1 v x40
c182: x2 v x40
c183: x3 v x40
c184: x4 v x40
c185: x5 v x40
c186: x1 v x41
c187: x2 v x41
c188: x3 v x41
c189: x4 v x41
c190: x5 v x41
c191: x1 v x42
c192: x2 v x42
c193: x3 v x42
c194: x4 v x42
c195: x5 v x42
c196: x1 v x43
c197: x2 v x43
c198: x3 v x43
c199: x4 v x43
c200: x5 v x43
c201: x1 v x44
c202: x2 v x44
c203: x3 v x44
c204: x4 v x44
c205: x5 v x44
c206: x1 v x45
c207: x2 v x45
c208: x3 v x45
c209: x4 v x45
c210: x5 v x45
c211: x1 v x46
c212: x2 v x46
c213: x3 v x46
c214: x4 v x46
c215: x5 v x46
c216: x1 v x47
c217: x2 v x47
c218: x3 v x47
c219: x4 v x47
c220: x5 v x47
c221: x1 v x48
c222: x2 v x48
c223: x3 v x48
c224: x4 v x48
c225: x5 v x48
c226: x1 v x49
c227: x2 v x49
c228: x3 v x49
c229: x4 v x49
c230: x5 v x49
c231: x1 v x50
c232: x2 v x50
c233: x3 v x50
c234: x4 v x50
c235: x5 v x50
c236: x1 v x51
c237: x2 v x51
c238: x3 v x51
c239: x4 v x51
c240: x5 v x51
c241: x1 v x52
c242: x2 v x52
c243: x3 v x52
c244: x4 v x52
c245: x5 v x52
c246: x1 v x53
c247: x2 v x53
c248: x3 v x53
c249: x4 v x53
c250: x5 v x53
c251: x1 v x54
c252: x2 v x54
c253: x3 v x54
c254: x4 v x54
c255: x5 v x54
c256: x1 v x55
c257: x2 v x55
c258: x3 v x55
c259: x4 v x55
c260: x5 v x55
c261: x1 v x56
c262: x2 v x56
c263: x3 v x56
c264: x4 v x56
c265: x5 v x56
c266: x1 v x57
c267: x2 v x57
c268: x3 v x57
c269: x4 v x57
c270: x5 v x57
c271: x1 v x58
c272: x2 v x58
c273: x3 v x58
c274: x4 v x58
c275: x5 v x58
c276: x1 v x59
c277: x2 v x59
c278: x3 v x59
c279: x4 v x59
c280: x5 v x59
c281: x1 v x60
c282: x2 v x60
c283: x3 v x60
c284: x4 v x60
c285: x5 v x60
c286: x1 v x61
c287: x2 v x61
c288: x3 v x61
c289: x4 v x61
c290: x5 v x61
c291: x1 v x62
c292: x2 v x62
c293: x3 v x62
c294: x4 v x62
c295: x5 v x62
c296: x1 v x63
c297: x2 v x63
c298: x3 v x63
c299: x4 v x63
c300: x5 v x63
c301: x1 v x64
c302: x2 v x64
c303: x3 v x64
c304: x4 v x64
c305: x5 v x64
c306: x1 v x65
c307: x2 v x65
c308: x3 v x65
c309: x4 v x65
c310: x5 v x65
c311: x1 v x66
c312: x2 v x66
c313: x3 v x66
c314: x4 v x66
c315: x5 v x66
c316: x1 v x67
c317: x2 v x67
c318: x3 v x67
c319: x4 v x67
c320: x5 v x67
c321: x1 v x68
c322: x2 v x68
c323: x3 v x68
c324: x4 v x68
c325: x5 v x68
c326: x1 v x69
c327: x2 v x69
c328: x3 v x69
c329: x4 v x69
c330: x5 v x69
c331: x1 v x70
c332: x2 v x70
c333: x3 v x70
c334: x4 v x70
c335: x5 v x70
c336: x1 v x71
c337: x2 v x71
c338: x3 v x71
c339: x4 v x71
c340: x5 v x71
c341: x1 v x72
c342: x2 v x72
c343: x3 v x72
c344: x4 v x72
c345: x5 v x72
c346: x1 v x73
c347: x2 v x73
c348: x3 v x73
c349: x4 v x73
c350: x5 v x73
c351: x1 v x74
c352: x2 v x74
c353: x3 v x74
c354: x4 v x74
c355: x5 v x74
c356: x1 v x75
c357: x2 v x75
c358: x3 v x75
c359: x4 v x75
c360: x5 v x75
c361: x1 v x76
c362: x2 v x76
c363: x3 v x76
c364: x4 v x76
c365: x5 v x76
c366: x1 v x77
c367: x2 v x77
c368: x3 v x77
c369: x4 v x77
c370: x5 v x77
c371: x1 v x78
c372: x2 v x78
c373: x3 v x78
c374: x4 v x78
c375: x5 v x78
c376: x1 v x79
c377: x2 v x79
c378: x3 v x79
c379: x4 v x79
c380: x5 v x79
c381: x1 v x80
c382: x2 v x80
c383: x3 v x80
c384: x4 v x80
c385: x5 v x80
c386: x1 v x81
c387: x2 v x81
c388: x3 v x81
c389: x4 v x81
c390: x5 v x81
c391: x1 v x82
c392: x2 v x82
c393: x3 v x82
c394: x4 v x82
c395: x5 v x82
c396: x1 v x83
c397: x2 v x83
c398: x3 v x83
c399: x4 v x83
c400: x5 v x83
c401: x1 v x84
c402: x2 v x84
c403: x3 v x84
c404: x4 v x84
c405: x5 v x84
c406: x1 v x85
c407: x2 v x85
c408: x3 v x85
c409: x4 v x85
c410: x5 v x85
c411: x1 v x86
c412: x2 v x86
c413: x3 v x86
c414: x4 v x86
c415: x5 v x86
c416: x1 v x87
c417: x2 v x87
c418: x3 v x87
c419: x4 v x87
c420: x5 v x87
c421: x1 v x88
c422: x2 v x88
c423: x3 v x88
c424: x4 v x88
c425: x5 v x88
c426: x1 v x89
c427: x2 v x89
c428: x3 v x89
c429: x4 v x89
c430: x5 v x89
c431: x1 v x90
c432: x2 v x90
c433: x3 v x90
c434: x4 v x90
c435: x5 v x90
c436: x1 v x91
c437: x2 v x91
c438: x3 v x91
c439: x4 v x91
c440: x5 v x91
c441: x1 v x92
c442: x2 v x92
c443: x3 v x92
c444: x4 v x92
c445: x5 v x92
c446: x1 v x93
c447: x2 v x93
c448: x3 v x93
c449: x4 v x93
c450: x5 v x93
c451: x1 v x94
c452: x2 v x94
c453: x3 v x94
c454: x4 v x94
c455: x5 v x94
c456: x1 v x95
c457: x2 v x95
c458: x3 v x95
c459: x4 v x95
c460: x5 v x95
c461: x1 v x96
c462: x2 v x96
c463: x3 v x96
c464: x4 v x96
c465: x5 v x96
c466: x1 v x97
c467: x2 v x97
c468: x3 v x97
c469: x4 v x97
c470: x5 v x97
c471: x1 v x98
c472: x2 v x98
c473: x3 v x98
c474: x4 v x98
c475: x5 v x98
c476: x1 v x99
c477: x2 v x99
c478: x3 v x99
c479: x4 v x99
c480: x5 v x99
c481: x1 v x100
c482: x2 v x100
c483: x3 v x100
c484: x4 v x100
c485: x5 v x100
c486: x1 v x101
c487: x2 v x101
c488: x3 v x101
c489: x4 v x101
c490: x5 v x101
c491: x1 v x102
c492: x2 v x102
c493: x3 v x102
c494: x4 v x102
c495: x5 v x102
c496: x1 v x103
c497: x2 v x103
c498: x3 v x103
c499: x4 v x103
c500: x5 v x103
c501: x1 v x104
c502: x2 v x104
c503: x3 v x104
c504: x4 v x104
c505: x5 v x104
c506: x1 v x105
c507: x2 v x105
c508: x3 v x105
c509: x4 v x105
c510: x5 v x105
c511: x1 v x106
c512: x2 v x106
c513: x3 v x106
c514: x4 v x106
c515: x5 v x106
c516: x1 v x107
c517: x2 v x107
c518: x3 v x107
c519: x4 v x107
c520: x5 v x107
c521: x1 v x108
c522: x2 v x108
c523: x3 v x108
c524: x4 v x108
c525: x5 v x108
c526: x1 v x109
c527: x2 v x109
c528: x3 v x109
c529: x4 v x109
c530: x5 v x109
c531: x1 v x110
c532: x2 v x110
c533: x3 v x110
c534: x4 v x110
c535: x5 v x110
c536: x1 v x111
c537: x2 v x111
c538: x3 v x111
c539: x4 v x111
c540: x5 v x111
c541: x1 v x112
c542: x2 v x112
c543: x3 v x112
c544: x4 v x112
c545: x5 v x112
c546: x1 v x113
c547: x2 v x113
c548: x3 v x113
c549: x4 v x113
c550: x5 v x113
c551: x1 v x114
c552: x2 v x114
c553: x3 v x114
c554: x4 v x114
c555: x5 v x114
c556: x1 v x115
c557: x2 v x115
c558: x3 v x115
c559: x4 v x115
c560: x5 v x115
c561: x1 v x116
c562: x2 v x116
c563: x3 v x116
c564: x4 v x116
c565: x5 v x116
c566: x1 v x117
c567: x2 v x117
c568: x3 v x117
c569: x4 v x117
c570: x5 v x117
c571: x1 v x118
c572: x2 v x118
c573: x3 v x118
c574: x4 v x118
c575: x5 v x118
c576: x1 v x119
c577: x2 v x119
c578: x3 v x119
c579: x4 v x119
c580: x5 v x119
c581: x1 v x120
c582: x2 v x120
c583: x3 v x120
c584: x4 v x120
c585: x5 v x120
c586: x1 v x121
c587: x2 v x121
c588: x3 v x121
c589: x4 v x121
c590: x5 v x121
c591: x1 v x122
c592: x2 v x122
c593: x3 v x122
c594: x4 v x122
c595: x5 v x122
c596: x1 v x123
c597: x2 v x123
c598: x3 v x123
c599: x4 v x123
c600: x5 v x123
c601: x1 v x124
c602: x2 v x124
c603: x3 v x124
c604: x4 v x124
c605: x5 v x124
c606: x1 v x125
c607: x2 v x125
c608: x3 v x125
c609: x4 v x125
c610: x5 v x125
c611: x1 v x126
c612: x2 v x126
c613: x3 v x126
c614: x4 v x126
c615: x5 v x126
c616: x1 v x127
c617: x2 v x127
c618: x3 v x127
c619: x4 v x127
c620: x5 v x127
c621: x1 v x128
c622: x2 v x128
c623: x3 v x128
c624: x4 v x128
c625: x5 v x128
c626: x1 v x129
c627: x2 v x129
c628: x3 v x129
c629: x4 v x129
c630: x5 v x129
c631: x1 v x130
c632: x2 v x130
c633: x3 v x130
c634: x4 v x130
c635: x5 v x130
c636: x1 v x131
c637: x2 v x131
c638: x3 v x131
c639: x4 v x131
c640: x5 v x131
c641: x1 v x132
c642: x2 v x132
c643: x3 v x132
c644: x4 v x132
c645: x5 v x132
c646: x1 v x133
c647: x2 v x133
c648: x3 v x133
c649: x4 v x133
c650: x5 v x133
c651: x1 v x134
c652: x2 v x134
c653: x3 v x134
c654: x4 v x134
c655: x5 v x134
c656: x1 v x135
c657: x2 v x135
c658: x3 v x135
c659: x4 v x135
c660: x5 v x135
c661: x1 v x136
c662: x2 v x136
c663: x3 v x136
c664: x4 v x136
c665: x5 v x136
c666: x1 v x137
c667: x2 v x137
c668: x3 v x137
c669: x4 v x137
c670: x5 v x137
c671: x1 v x138
c672: x2 v x138
c673: x3 v x138
c674: x4 v x138
c675: x5 v x138
c676: x1 v x139
c677: x2 v x139
c678: x3 v x139
c679: x4 v x139
c680: x5 v x139
c681: x1 v x140
c682: x2 v x140
c683: x3 v x140
c684: x4 v x140
c685: x5 v x140
c686: x1 v x141
c687: x2 v x141
c688: x3 v x141
c689: x4 v x141
c690: x5 v x141
c691: x1 v x142
c692: x2 v x142
c693: x3 v x142
c694: x4 v x142
c695: x5 v x142
c696: x1 v x143
c697: x2 v x143
c698: x3 v x143
c699: x4 v x143
c700: x5 v x143
c701: x1 v x144
c702: x2 v x144
c703: x3 v x144
c704: x4 v x144
c705: x5 v x144
c706: x1 v x145
c707: x2 v x145
c708: x3 v x145
c709: x4 v x145
c710: x5 v x145
c711: x1 v x146
c712: x2 v x146
c713: x
```

Changing of an AND-precedence relation into an OR-precedence relation, and Bi-directional relation



- Activity D have a OR-precedence relation:
 - Activity D is transformed to a dummy activity;
 - Activities DA, DB, DC are created, duplicating activity D;
 - The realization of activity D, implies to realize one and only one activity: DA, DB and DC.
- Activities E and F have a Bi-directional relation:
 - Added 6 activities:
 - EF1/2 - is the changeover times activity;
 - E1, F1 - duplicate of activities E and F, with the order E » F;
 - E2, F2 - duplicate of activities E and F, with the order F » E.

Error:



- B precedes F, but in F1 it need to wait for the EF activity also
- G is successor of E, but in the E1 it need to wait the EF activity also

Using (M)RCPSP instances to create OR-precedence relations and BI-directional relations instances

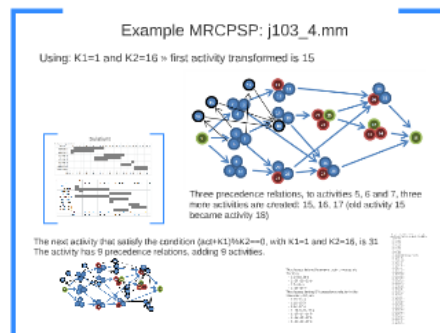
In a RCPSP or MRCPS instance:

- 1. Select a number of activities, to make all its precedence relations as OR precedence relations (like in activity D);
- 2. Select a number of activities with AND precedence relations, and select one of its preceding activity, and make it as a bi-directional precedence relation (like activity E and F).

Step 1 does not produce any violation, but is done only if the selected activity is not a dummy one. Step 2 does not produce any violation, if the activities are inserted between the maximal activity that is preceding activity B, and the minimal activity that is succeeding activity A, assuming a transformation of a precedence of A over B.

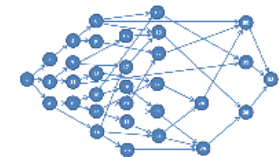
Rules:

- Add OR-precedence relations to the activities that satisfy: **(act+K1)%K2==0**
- Add BI-directional we use an identical rule

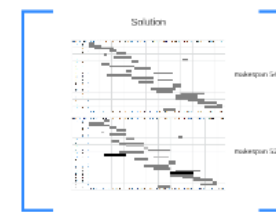
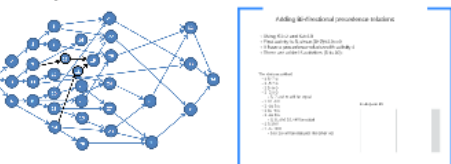


Example RCPSP: j3010_8.sm

- Add OR-precedence relations to the activities that satisfy: $(act+K1)\%K2=0$, with K1=3 and K2=10
- The first activity to satisfy the expression is activity number 7, since $(7+3)\%10=0$
 - Only one precedence relation: nothing to do
- The next activity that satisfies the expression, is activity 17, that have two precedence relations, to activity 7 and 14.
 - Inserted two new activities, 17 and 18 (old activity 17 became a dummy one: 19)
 - SAT clauses inserted:
 - 1 17 18 0
 - 1 -17 -18 0

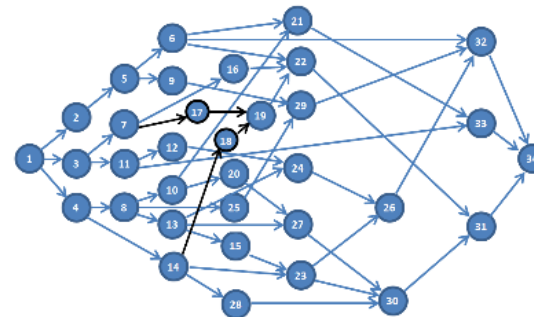
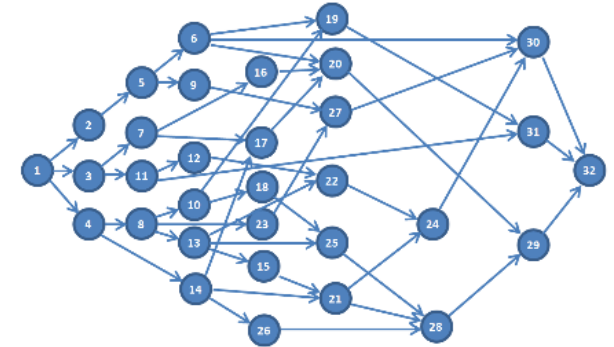


- The next activity is now the activity 27, that have two precedence relations, to activities 13 and 20, so two new activities are also inserted, 27 and 28, and also the clauses in SAT:
- 1 27 28 0
 - 1 -27 -28 0



Example RCPSP: j3010_8.sm

- Add OR-precedence relations to the activities that satisfy: $(act + K1) \% K2 == 0$, with $K1=3$ and $K2=10$
- The first activity to satisfy the expression is activity number 7, since $(7+3)\%10==0$
 - Only one precedence relation: nothing to do
- The next activity that satisfies the expression, is activity 17, that have two precedence relations, to activity 7 and 14.
 - Inserted two new activities, 17 and 18 (old activity 17 became a dummy one: 19)
- SAT causes inserted:
 - 1 17 18 0
 - 1 -17 -18 0



Adding Bi-directional precedence relations

- Using $K1=2$ and $K3=10$
- First activity is 8, since $(8+2)\%10==0$
- It have a precedence relation with activity 4
- There are added 6 activities (5 to 10):

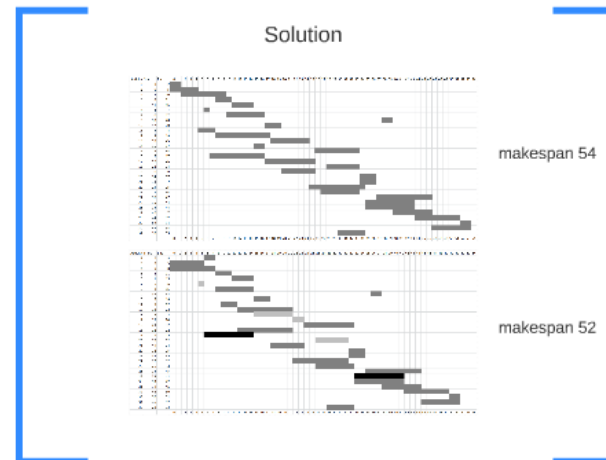
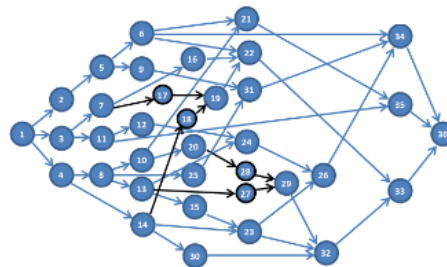
The classes added:

- 1-5-7-0
- 1-5-7-0
- 1-5-8-0
- 1-5-9-0
- 5-7 and 9, will be equal
- 1-10-8-0
- 1-10-8-0
- 1-10-9-0
- 1-10-9-0
- 6-8 and 10, will be equal
- 1-5-10-0
- 1-5-10-0
- 5 or 10 will be realized, the other not

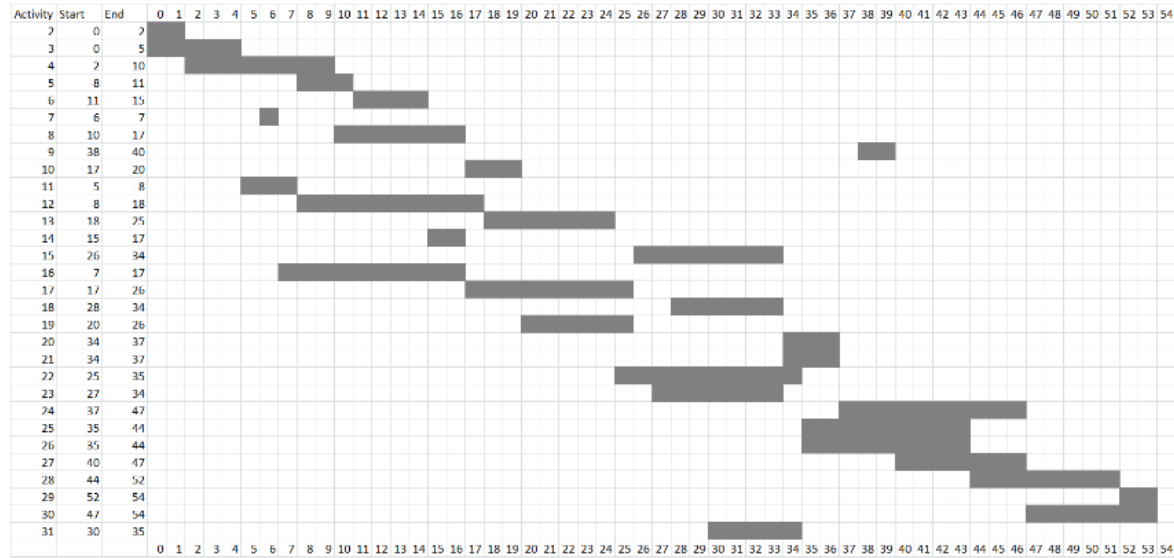
makespan: 45

The next activity is now the activity 27, that have two precedence relations, to activities 13 and 20, so two new activities are also inserted, 27 and 28, and also the clauses in SAT:

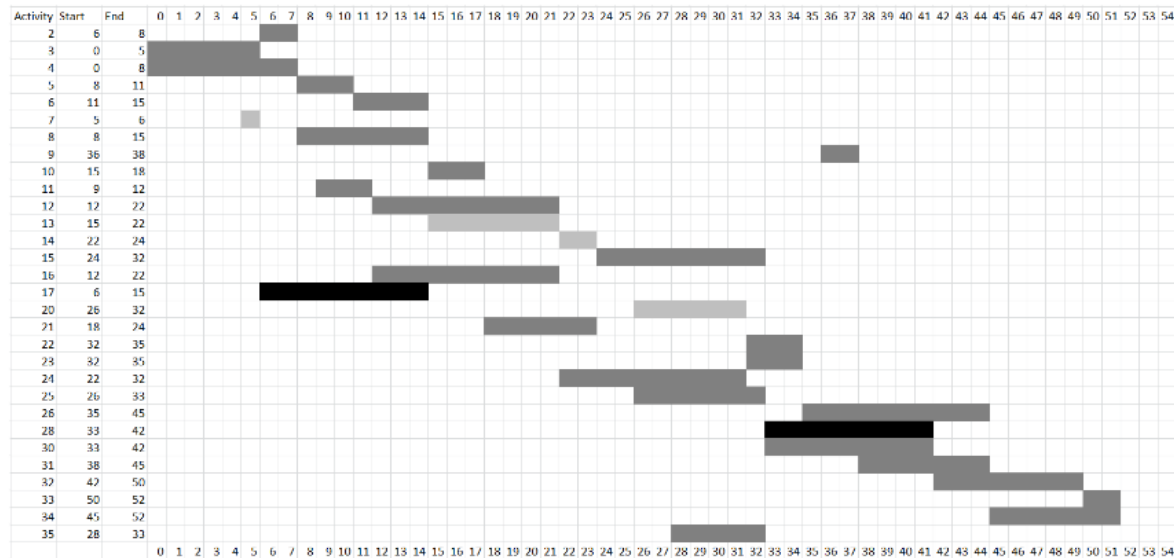
- 1 27 28 0
- 1 -27 -28 0



Solution



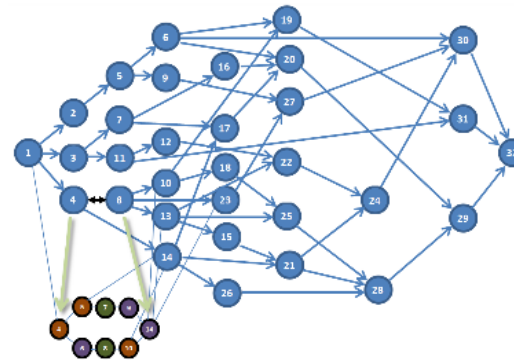
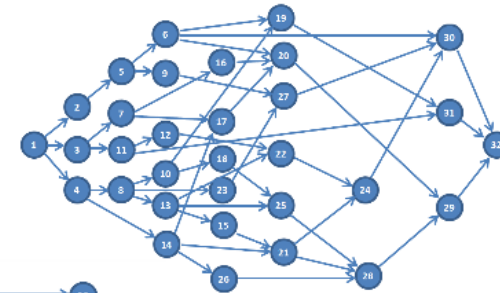
makespan 54



makespan 52

Adding Bi-directional precedence relations

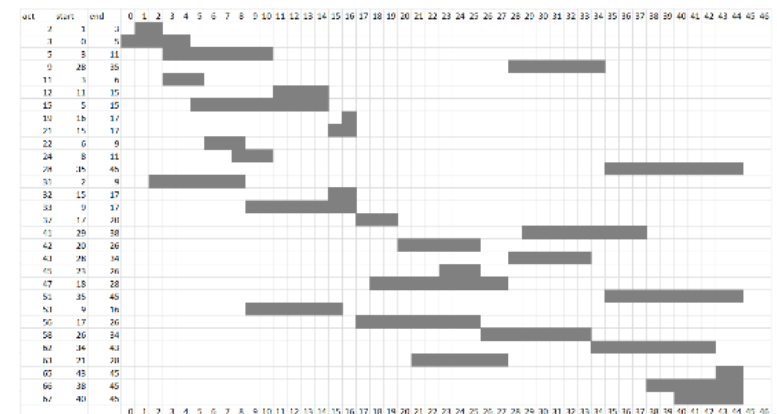
- Using $K1=2$ and $K3=10$
- First activity is 8, since $(8+2)\%10=0$
- It has a precedence relation with activity 4
- There are added 6 activities (5 to 10):



The clauses added:

- 1 5 -7 0
- 1 -5 7 0
- 1 5 -9 0
- 1 -5 9 0
 - 5, 7 and 9, will be equal
- 1 10 -8 0
- 1 -10 8 0
- 1 10 -6 0
- 1 -10 6 0
 - 6, 8, and 10, will be equal
- 1 5 10 0
- 1 -5 -10 0
 - 5 or 10 will be realized, the other not

makepsan 45



Tests and Results

1. Run without OR-precedence relations and Bi-directional relations:

- Check if our procedure is close to the state-of-the-art in MRCPSP instance sets

2. Run with different density of OR-precedence relations:

- Quantify the increase needed on the number of activities
- Quantify the gain in makespan by consider OR-precedence relations, relative to treat those precedence relations as AND-precedence relations

3. Run with bi-directional relations.

MMLIB results

Preliminary results: Results from V. Van Peteghem, M. Vanhoucke / European Journal of Operational Research vol. 202(3) 2008

Scenario	Limit	1000	5000	50000
MMLIB50	feasibility	83%	83%	83%
	dec. Eq.	20%	15%	15%
MMLIB100	feasibility	83%	83%	83%
	dec. Eq.	40%	24%	19%
MMLIB1	feasibility	n.a.	54%	48%
	dec. Eq.	n.a.	67%	54%
	dec. Eq.	n.a.	10%	2%

New results:

Scenario	Limit	100	1000	4000	5000
MMLIB50	feasibility	83%	100%	100%	100%
	dec. Eq.	30%	45%	30%	20%
MMLIB100	feasibility	83%	100%	100%	100%
	dec. Eq.	40%	55%	35%	25%
MMLIB1	feasibility	83%	100%	100%	100%
	dec. Eq.	40%	55%	35%	25%
Optimal		0%	1%	1%	1%

Feasibility:

- MMLIB: resource structure (cannot have resources with more than 127 units) solved
- MMLIB50/100: SAT backtracks limit reached
- Using an heuristic (BSAT)
- If still not solved, using a penalty in the makespan.

OR-precedence relations results

PSPLIB, RCPSP results

OR-precedence relations	PSPLIB	RCPSP	PSPLIB	RCPSP	PSPLIB	RCPSP	PSPLIB	RCPSP
100	100%	100%	100%	100%	100%	100%	100%	100%
200	100%	100%	100%	100%	100%	100%	100%	100%
300	100%	100%	100%	100%	100%	100%	100%	100%
400	100%	100%	100%	100%	100%	100%	100%	100%
500	100%	100%	100%	100%	100%	100%	100%	100%
600	100%	100%	100%	100%	100%	100%	100%	100%
700	100%	100%	100%	100%	100%	100%	100%	100%
800	100%	100%	100%	100%	100%	100%	100%	100%
900	100%	100%	100%	100%	100%	100%	100%	100%
1000	100%	100%	100%	100%	100%	100%	100%	100%

- Number of total activities, pass in the worst case, from 120 to 338 activities;
- Makespan with OR-precedence relations could improve up to 20%;
- The number of instances that became worst, decrease with the increase of CPU time;
- The SAT+RCPSP approach, can model these type of precedence relation for RCPSP instances.

BI-directional precedence relations results

PSPLIB, RCPSP results:

BI-directional precedence relations	PSPLIB	RCPSP	PSPLIB	RCPSP	PSPLIB	RCPSP	PSPLIB	RCPSP
100	100%	100%	100%	100%	100%	100%	100%	100%
200	100%	100%	100%	100%	100%	100%	100%	100%
300	100%	100%	100%	100%	100%	100%	100%	100%
400	100%	100%	100%	100%	100%	100%	100%	100%
500	100%	100%	100%	100%	100%	100%	100%	100%
600	100%	100%	100%	100%	100%	100%	100%	100%
700	100%	100%	100%	100%	100%	100%	100%	100%
800	100%	100%	100%	100%	100%	100%	100%	100%
900	100%	100%	100%	100%	100%	100%	100%	100%
1000	100%	100%	100%	100%	100%	100%	100%	100%

- Number of total activities, pass in the worst case, from 120 to 470 activities;
- Makespan with BI-directional precedence relations could improve also up to 20%;
- The number of instances that became worst, decrease with the increase of CPU time;
- The SAT+RCPSP approach, can model these type of precedence relation for RCPSP instances.

MMLIB results

Preliminary results:

Schedule Limit:		1,000	5,000	50,000
MMLIB50	feasibility	83%	83%	83%
	dev.LB	30%	21%	16%
	dev.Best	13%	5%	1%
MMLIB100	feasibility	83%	83%	82%
	dev.LB	40%	28%	19%
	dev.Best	19%	9%	2%
MMLIB+	feasibility	n.a.	50%	50%
	dev.LB	n.a.	67%	54%
	dev.Best	n.a.	10%	2%

Infeasibility:

- MMLIB+: resource structure (cannot have resources with more than 127 units): solved.
- MMLIB 50/100: SAT backtracks limit reached: solved.
 - Using an heuristic (GSAT);
 - If still not solved, using a penalty in the makespan.

Results from V. Van Peteghem, M. Vanhoucke / European Journal of Operational Research xxx (2013) xxx–xxx

Table 7
Solutions for MMLIB after 1000, 5000 and 50000 schedules.

Author	MS	Number of schedules		
		1000	5000	50000
BLP004	SA	51,853	63,262	62,400
BOCP95	SA	57,559	59,693	178,521
DELL05	SA	272,064	1,27,708	1,65,701
COE111	CA	83,373	83,333	83,152
ROJ000	CA	393,784	183,330	168,330
CH045	ACO	122,223	122,223	122,223
MB007	CA	272,223	183,333	46,881
ALC005	CA	56,000	43,200	46,880
WAL111	SA	180,274	180,411	11,649
JAK028	IS	49,208	38,888	32,011
ZHANG05	PS	48,515	75,641	70,771
TSE020	CA	65,117	37,002	20,444
WANG13	PS	75,117	31,005	28,770
DEW012	SS	28,440	32,216	28,220
JOH011	SA	48,046	15,811	22,011
ELL010	CA	43,844	32,447	20,911
LAF010	CA	75,000	70,000	70,000
LEV020	CA	34,106	28,810	28,810
DM008	PS	46,108	35,410	30,211
YANG16	CA	24,000	2,411	24,011
VANP11	SS	38,117	35,445	33,770

(Number): % of feasible solutions found.
Number: average % deviation from minimal CPILB.

Table 8
Solutions for MMLIB 100 after 1000, 5000 and 50000 schedules.

Author	MS	Number of schedules		
		1000	5000	50000
SLOW94	SA	118,891	123,811	121,481
REY796	SA	100,071	100,071	100,071
HEJ100	SA	160,871	160,881	160,411
OP049	CA	100,071	107,501	107,501
CAE111	CA	103,171	103,171	103,181
SG002	CA	107,000	103,620	103,221
CH040	ACO	103,171	103,171	103,171
WY0111	MSL	102,101	102,101	102,101
AF007	CA	61,800	57,000	60,000
JUH008	PS	191,881	88,411	81,221
LI000	CA	72,200	66,004	37,004
ZHANG05	PS	82,620	68,881	30,300
BAJ020	SS	40,116	37,000	34,117
WANG13	PS	57,844	38,555	37,455
XUE011	SA	53,000	30,000	30,220
TI010	CA	50,211	40,220	30,000
HAR011	CA	38,000	33,000	28,000
DEA	DEA	52,311	38,881	28,000
LOU008	CA	36,220	31,000	27,000
VANP10	CA	37,500	29,500	29,500
VANP11	SS	38,770	36,000	34,000

(Number): % of feasible solutions found.
Number: average % deviation from minimal CPILB.

Table 9
Solutions for MMLIB+ after 1000, 5000 and 50000 schedules.

Author	MS	Number of schedules	
		5000	50000
SLOW94	SA	4,781	4,781
COE111	CA	10,000	10,000
BOJ795	SA	10,480	10,577
CH045	ACO	10,807	10,807
EDM009	CA	10,668	10,677
BOJ010	SA	10,500	10,500
MB007	CA	10,500	10,500
VANP10	CA	10,500	10,500
ZHANG06	PS	10,811	10,811
ALC005	CA	10,555	10,487
TSE020	CA	10,000	10,216
JAK008	PS	10,420	10,399
WAL111	MSL	10,576	10,288
DEW012	SS	10,145	10,107
HAR011	CA	10,201	10,145
ROJ	CA	10,000	10,000
ELL010	CA	10,000	10,000
DM009	DEA	10,689	10,575
JOH011	SA	10,100	10,100
LEV020	CA	10,007	10,073
VANP11	SS	10,145	10,276

(Number): % of feasible solutions found.
Number: average % deviation from minimal CPILB.

New results:

Schedule limit:		100	1000	5000	50000
MMLIB50	feasibility	83%	100%	100%	100%
	dev.LB	58%	40%	30%	25%
	dev.Best	37%	13%	5%	1%
MMLIB100	Optimal	0,2%	7,2%	22,0%	35,2%
	Best results				10
	feasibility	83%	93%	100%	100%
MMLIB+	dev.LB	65%	46%	37%	27%
	dev.Best	41%	20%	10%	3%
	Optimal	0,0%	1,9%	8,3%	32,8%
MMLIB+	Best results			1	7
	feasibility	100%	100%	100%	100%
	dev.LB	195%	147%	121%	104%
MMLIB+	dev.Best	59%	32%	17%	7%
	Optimal	0,0%	0,5%	2,1%	6,6%
	Best results				216

OR-precedence relations results

PSPLIB, RCPSP results

OR-precedence relations		none		1/10					All				
Schedule limit	Instance set	%UB	%LB	%UB	%none	#best	#same	#worst	%UB	%none	#best	#same	#worst
5000	J30	0,06%	0,06%	-2,59%	-2,65%	215	261	4	-19,35%	-19,41%	476	4	0
	J60	0,45%	1,09%	-2,70%	-3,14%	277	184	19	-18,09%	-18,53%	470	6	4
	J90	0,69%	1,82%	-2,92%	-3,61%	296	158	26	-18,39%	-19,06%	457	10	13
	J120	2,13%	5,26%	0,51%	-1,61%	333	180	87	-4,79%	-6,89%	497	45	58
50000	J30	0,03%	0,03%	-2,70%	-2,72%	220	259	1	-20,86%	-20,89%	477	3	0
	J60	0,22%	0,85%	-3,07%	-3,29%	275	197	8	-19,84%	-20,05%	473	6	1
	J90	0,33%	1,43%	-3,38%	-3,71%	286	175	19	-20,27%	-20,59%	466	9	5
	J120	1,10%	4,16%	-0,73%	-1,83%	354	185	61	-6,62%	-7,69%	529	44	27
500000	J30	0,01%	0,01%	-2,76%	-2,77%	227	253	0	-22,19%	-22,19%	479	1	0
	J60	0,13%	0,77%	-3,23%	-3,36%	285	190	5	-21,14%	-21,27%	475	4	1
	J90	0,19%	1,29%	-3,59%	-3,77%	295	170	15	-21,62%	-21,80%	472	6	2
	J120	0,63%	3,66%	-1,30%	-1,93%	353	200	47	-7,60%	-8,22%	542	40	18

- Number of total activities, pass in the worst case, from 120 to 338 activities;
- Makespan with OR-precedence relations could improve up to 20%;
- The number of instances that became worst, decrease with the increase of CPU time;
- The SAT+RCPSP approach, can model these type of precedence relation for RCPSP instances.

MRCPSPP results (MMLIB)

Instance size increase

MMLIB50		5000 schedules limit		Maximal number of activities		Average number of activities	
K1	K2						
-	-			152	152,0		
3	10			731	207,1		
5	10			602	200,5		
3	25			716	200,0		
5	25			689	200,9		
3	50			222	200,2		
5	50			287	192,0		
3	1			2411	1829,1		

MMLIB50				5000 schedules limit	
K1	K2	Maximal number of activities	Average number of activities	Maximal number of activities	Average number of activities
3	10	731	207,1	322	202,0
5	10	602	200,5	118	114,0
3	25	716	200,0	222	212,0
5	25	689	200,9	227	218,0
3	50	222	200,2	287	192,0
5	50	287	192,0	287	192,0

Instances greater than 900 activity-modes, are not processed due data structure restrictions.

Makespan variation

MMLIB50		5000 schedules limit			
K1	K2	Better	Same makespan	Worst makespan	Total
3	10	191	132	147	450
5	10	183	134	133	450
3	25	172	157	121	450
5	25	170	152	128	450
3	50	131	192	127	450
5	50	122	191	141	451

Using only results on instances with feasible solutions

BI-directional precedence relations results

PSPLIB, RCPSP results:

BI-directional precedence relations		none		1/10					All				
Schedule limit	Instance set	%UB	%LB	%UB	%none	#best	#same	#worst	%UB	%none	#best	#same	#worst
5000	J30	0,06%	0,06%	-5,27%	-5,33%	301	138	41	-19,57%	-19,63%	469	5	6
	J60	0,45%	1,09%	-6,70%	-7,14%	372	90	18	-19,48%	-19,90%	475	1	4
	J90	0,69%	1,82%	-7,02%	-7,70%	392	70	18	-17,29%	-17,95%	466	5	9
	J120	2,13%	5,26%	-2,49%	-4,57%	501	54	45	-6,35%	-8,42%	522	19	59
50000	J30	0,03%	0,03%	-5,43%	-5,46%	309	138	33	-20,55%	-20,58%	472	2	6
	J60	0,22%	0,85%	-7,22%	-7,43%	381	81	18	-22,50%	-22,69%	479	0	1
	J90	0,33%	1,43%	-7,75%	-8,07%	400	64	16	-21,24%	-21,55%	477	1	2
	J120	1,10%	4,16%	-4,34%	-5,40%	553	30	17	-9,98%	-11,03%	577	6	17

- Number of total activities, pass in the worst case, from 120 to 470 activities;
- Makespan with BI-directional precedence relations could improve also up to 20%;
- The number of instances that became worst, decrease with the increase of CPU time;
- The SAT+RCPSP approach, can model these type of precedence relation for RCPSP instances.

Conclusions and Future Work

- Proposed an approach to solve the RCPSP with logical constraints:
 - OR precedence relations;
 - BI-directional precedence relations;
- The method uses an extension of multi-mode, where the realization of each activity-mode is controlled by SAT, with more flexibility over the MRCPSP procedures, and the schedule is done by a RCPSP procedure
- The method, applied to MRCPSP instances, reveal to be competitive.
- Future work we plan:
 - Study these type of logical constraints without resources, and with multiple mode;
 - Study mode identity constraints in multiple mode.