TABLE OF CONTENTS

I. Problem Statement

- The problem is to find a schedule that minimizes the time to build a five-segment bridge (see Figure 1). The project contains a set of 46 tasks (see Table 1) and a set of constraints between these tasks. Beside the usual precedence constraints, there are disjunctive constraints due to the restricted resources which must be shared by several tasks. For example, the caterpillar is used by tasks V1 and V2, which therefore cannot be performed at the same time.

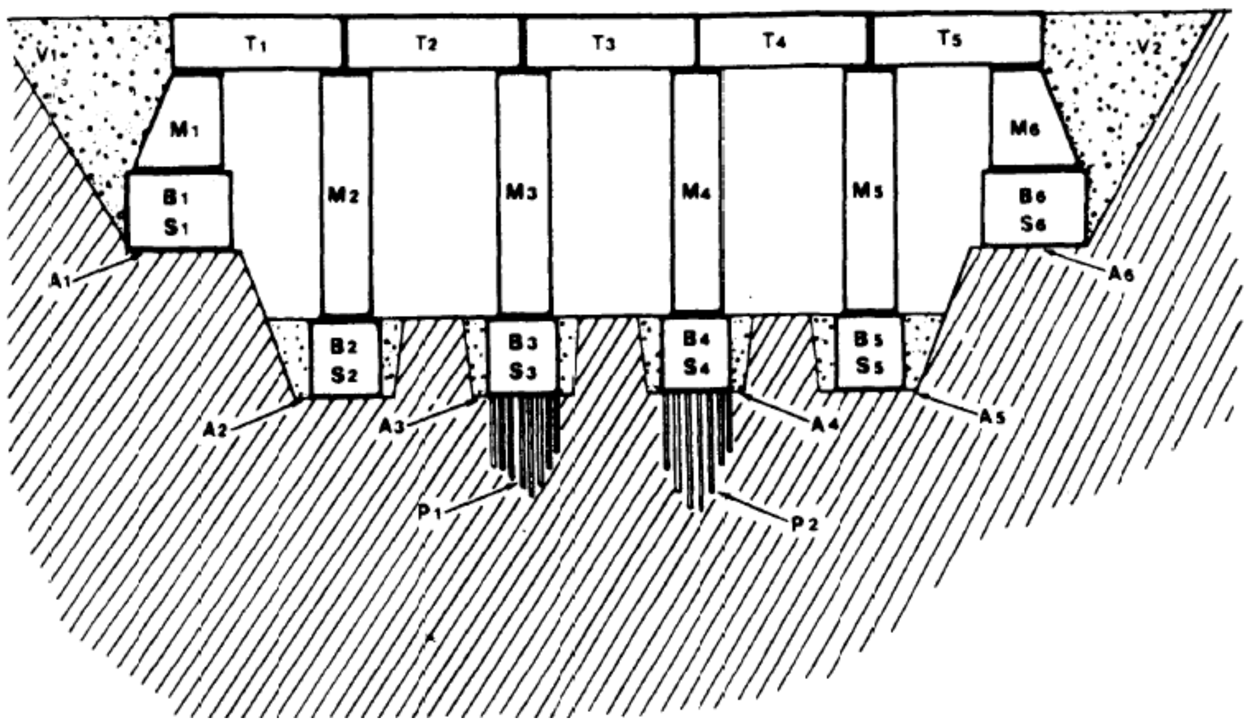

Figure 1. Five-segment bridge

N	Name	Description	Duration	Precedence	Resource
1	PA	beginning of project	0	-	-
2	A1	excavation (abument 1)	4	PA	excavator
3	A2	excavation (pillar 1)	2	PA	excavator
4	A3	excavation (pillar 2)	2	PA	excavator
5	A4	excavation (pillar 3)	2	PA	excavator
6	A5	excavation (pillar 4)	2	PA	excavator
7	A6	excavation (abument 2)	5	PA	excavator
8	P1	foundation piles 2	20	A3	pile-driver
9	P2	foundation piles 3	13	A4	pile-driver
10	UE	erection of temporary housing	10	PA	-
11	S1	formwork (abutment 1)	8	A1	carpentry
12	S2	formwork (pillar 1)	4	A2	carpentry
13	S3	formwork (pillar 2)	4	P1	carpentry
14	S4	formwork (pillar 3)	4	P2	carpentry
15	S5	formwork (pillar 4)	4	A5	carpentry
16	S6	formwork (abutment 2)	10	A6	carpentry
17	B1	concrete foundation (abutment 1)	1	S1	concrete-mixer
18	B2	concrete foundation (pillar 1)	1	S2	concrete-mixer
19	B3	concrete foundation (pillar 2)	1	S3	concrete-mixer
20	B4	concrete foundation (pillar 3)	1	S4	concrete-mixer
21	B5	concrete foundation (pillar 4)	1	S5	concrete-mixer
22	B6	concrete foundation (abutment 2)	1	S6	concrete-mixer
23	AB1	concrete setting time (abutment 1)	1	B1	-
24	AB2	concrete setting time (pillar 1)	1	B2	-
25	AB3	concrete setting time (pillar 2)	1	B3	-
26	AB4	concrete setting time (pillar 3)	1	B4	-
27	AB5	concrete setting time (pillar 4)	1	B5	-
28	AB6	concrete setting time (abutment 2)	1	B6	-
29	M1	masonry work (abutment 1)	16	AB1	bricklaying
30	M2	masonry work (pillar 1)	8	AB2	bricklaying
31	M3	masonry work (pillar 2)	8	AB3	bricklaying
32	M4	masonry work (pillar 3)	8	AB4	bricklaying
33	M5	masonry work (pillar 4)	8	AB5	bricklaying
34	M6	masonry work (abutment 2)	20	AB6	bricklaying
35	L	delivery of preformed bearers	2	-	crane
36	T1	positioning (preformed bearer 1)	12	M1, M2, L	crane
37	T2	positioning (preformed bearer 2)	12	M2, M3, L	crane
38	T3	positioning (preformed bearer 3)	12	M3, M4, L	crane
39	T4	positioning (preformed bearer 4	12	M4, M5, L	crane
40	T5	positioning (preformed bearer 5)	12	M5, M6, L	crane
41	UA	removal of the temporary housing	10	-	-
42	V1	filling 1	15	T1	caterpillar

43	V2	filling 2	10	T5	caterpillar
44	K1	point 1 of cost function	0	-	-
45	K2	point 2 of cost function	0	-	-
46	PE	end of project	0	T2, T3, T4, V1, V2, UA	-

Table 1. Tasks of problem

- Table 1 includes information about n tasks where each task i is characterized by its duration d, and a set of precedence constraints with some other tasks.

II. Constrains

We denote that:

- S(i): the start day of task i, domain of this variable is [0 lg] (lg is total duration of all tasks, concretely in this problem is 271).
- d(i): duration of task i

1. Precedence constrain

- A precedence constraint between tasks i and j implies that task j must start after the completion of task i.

S(j) >= S(i) + d(i), with all task i must be completed before task j

- In this problem, we will follow column "Precedence" in Table 1. The task in this column must be completed before the main task in this row is started.

2. Distance constrain

- Task i must be completed before task j an interval time.

S(i) >= S(j) + d(j) + 10, task i must be started at least 10 days before completing task j

- In this problem, we have 5 distance constrain:

- The time between the completion of a particular formwork and the completion of its corresponding concrete foundation is at most 4 days.
- There are at most 3 days between the end of a particular excavation (or foundation piles) and the beginning of the corresponding formwork.
- The formworks must start at least 6 days after the beginning of the erection of the temporary housing.
- The removal of the temporary housing can start two days before the end of the last masonry work.
- The delivery of the performed bearers occurs exactly 30 days after the beginning of the project.

3. Disjunctive constrain

- A disjunctive constraint states that two tasks i and j cannot be performed simultaneously, i.e., either task i must precede task j or task j must precede task i. These constraints are due to the fact that tasks i and j use the same resource.

S(i) >= S(j) + d(j) or S(j) >= S(i) + d(i), if task i and j use the same resource

- In this problem, we will follow the column "Resource" in Table 1. If 2 task using the same resource, then consider it as Precedence constrain.

We will try to satisfy all these constrains.

III. Implementation in B-Prolog

- Declare function name and argument:

```
bridge(R):-
```

- Declare variables and domain values:

```
R=[A1,A2,A3,A4,A5,A6,P1,P2,UE,S1,S2,S3,S4,S5,S6,B1,B2,B3,B4,B5,B6,AB1,AB2,AB3,
   AB4,AB5,AB6,M1,M2,M3,M4,M5,M6,L,T1,T2,T3,T4,T5,UA,V1,V2,Send],
R in 0..271,
```

- Precedence constrain:

```
P1 #>= A3 + 2,
P2 #>= A4 + 2,
S1 #>= A1 + 4,
S2 #>= A2 + 2,
S3 #>= P1 + 20,
S4 #>= P2 + 13,
S5 #>= A5 + 2,
S6 #>= A6 + 5,
B1 #>= S1 + 8,
B2 #>= S2 + 4,
B3 #>= S3 + 4,
B4 #>= S4 + 4,
B5 #>= S5 + 4,
B6 #>= S6 + 10,
AB1 #>= B1 + 1,
AB2 #>= B2 + 1,
AB3 #>= B3 + 1,
AB4 #>= B4 + 1,
AB5 #>= B5 + 1,
AB6 #>= B6 + 1,
M1 #>= AB1 + 1,
M2 #>= AB2 + 1,
M3 #>= AB3 + 1,
M4 #>= AB4 + 1,
M5 #>= AB5 + 1,
M6 #>= AB6 + 1,
T1 #>= M1 + 16, T1 #>= M2 + 8, T1 #>= L + 2,
T2 #>= M2 + 8, T2 #>= M3 + 8, T2 #>= L + 2,
T3 #>= M3 + 8, T3 #>= M4 + 8, T3 #>= L + 2,
T4 #>= M4 + 8, T4 #>= M5 + 8, T4 #>= L + 2,
T5 #>= M5 + 8, T5 #>= M6 + 20, T5 #>= L + 2,
```

```
    V1 #>= T1 + 12,
    V2 #>= T5 + 12,
    Send #>= T2 + 12, Send #>= T3 + 12, Send #>= T4 + 12, Send #>= V1 + 15,
    Send #>= V2 + 10, Send #>= UA + 10,
```

- Distance constrain:

```
    S1 + 8 + 4 #>= B1 + 1,
    S2 + 4 + 4 #>= B2 + 1,
    S3 + 4 + 4 #>= B3 + 1,
    S4 + 4 + 4 #>= B4 + 1,
    S5 + 4 + 4 #>= B5 + 1,
    S6 + 10 + 4 #>= B6 + 1,
    A1 + 4 + 3 #>= S1,
    A2 + 2 + 3 #>= S2,
    P1 + 20 + 3 #>= S3,
    P2 + 13 + 3 #>= S4,
    A5 + 2 + 3 #>= S5,
    A6 + 5 + 3 #>= S6,
    S1 #>= UE + 6,
    S2 #>= UE + 6,
    S3 #>= UE + 6,
    S4 #>= UE + 6,
    S5 #>= UE + 6,
    S6 #>= UE + 6,
    UA #>= M1 + 16 - 2,
    UA #>= M2 + 8 - 2,
    UA #>= M3 + 8 - 2,
    UA #>= M4 + 8 - 2,
    UA #>= M5 + 8 - 2,
    UA #>= M6 + 20 - 2,
    L #= 30,
```

- Disjunctive constrain:

```
    A2 #>= A1 + 4,
    A3 #>= A2 + 2,
    A4 #>= A3 + 2,
    A5 #>= A4 + 2,
    A6 #>= A5 + 2,
    P2 #> P1 + 20,
    S2 #>= S1 + 8,
    S3 #>= S2 + 4,
    S4 #>= S3 + 4,
    S5 #>= S4 + 4,
```

```
    S6 #>= S5 + 4,
    B2 #>= B1 + 1,
    B3 #>= B2 + 1,
    B4 #>= B3 + 1,
    B5 #>= B4 + 1,
    B6 #>= B5 + 1,
    M2 #>= M1 + 16,
    M3 #>= M2 + 8,
    M4 #>= M3 + 8,
    M5 #>= M4 + 8,
    M6 #>= M5 + 8,
    T1 #>= L + 2,
    T2 #>= T1 + 12,
    T3 #>= T2 + 12,
    T4 #>= T3 + 12,
    T5 #>= T4 + 12,
    V2 #>= V1 + 15,
```

- Objective optimization:

```
minof(labeling(R),Send).
```

* Result:

```
C:\Users\guh3hc\Desktop\BProlog\bp.exe
B-Prolog Version 8.1, All rights reserved, (C) Afany Software 1994-2014.
| ?- cd('C:/Users/guh3hc/Desktop')

yes
| ?- consult(assignment)
consulting::assignment.pl

yes
| ?- bridge(L)
L = [0,9,11,13,46,48,13,34,0,6,14,33,47,51,55,14,18,37,51,55,65,15,19,38,52,56,66,16,32,40,53,61,69,30,40,52,64,76,89,87,52,101,111]
yes
| ?-
```

- There are 2 tasks which are started at day 0: excavation (abument 1) and erection of temporary housing.
- With the result above, all constrains are satisfied.
- Total day need to complete the bridge with all above constrains is 111 days.

IV. References

1. Paper “SOLVING LARGE COMBINATORIAL PROBLEMS IN LOGIC PROGRAMMING”

2. https://www.comp.nus.edu.sg/~henz/sma2000/slides/bridge.html

3. Source code: https://github.com/NguyenDucHuy14287/Bridge_scheduling