Progressive Party Problem (party)

Problem: This problem was originally stated by Peter Hubbard (Southampton University), organizer of the yacht rally of 42 boats and their crews. A important evening event in the yacht rally is to organize a party where each crew socializes with as many other crews as possible. Some of the boats are selected to serve as "host boats" where the party takes place, the other boats are the "guest boats". The crews of the host boats stay on their boats for the whole party, while the crews of the guest boats visits the host boats. In order to socialize as much as possible, the guest crews only stay for 30 minutes at one host, then they visit another host boat. The whole party lasts 3 hours.

Each crew has a given size of members and they always stay together, and each boat has a given capacity. The problem now is to assign all the guest crews to the host boats for each of the six time slots, using the minimum number of host boats.

The partitioning of crews into guests and hosts is fixed throughout the whole party and no two crews should meet more than once. For organizing reasons the boats 1 to 3 must be host boats, and the three boats 40 to 42 are guest boats. The problem and its mathematical formulation is described in [1].

Modeling Steps

There is a set of 42 boats (crews), that is: $i, j, k \in \{1, \ldots, 42\}$ and a set of 6 time slots: $t \in \{1, \ldots, 6\}$.

1. The crew size is given by s_i, and the boat capacity – the maximal number of persons that a boat can accomodate including the crew itself – is given by C_i.

2. Hence, the number of guests that a boat can invite is: $c_i = \max\,(0, C_i - s_i)$.

3. We introduce a binary variable $x_{i,j,t}$. $x_{i,j,t} = 1$ if a guest crew j visits the host crew i in time slot t, otherwise $x_{i,j,t} = 0$.

4. Furthermore we use a binary variable to decide which are host boats. $h_i = 1$ if i is a host boat, otherwise i is a guest boat ($h_i = 0$).

5. There are only parties on host boats, or, if a guest crew j visits the boat i in a time period, then i is a host boat:

$$x_{i,j,t} \rightarrow h_i \quad (x_{i,j,t} \leq h_i) \quad \text{ for all } i, j, t (i \neq j)$$

6. The total guest capacity of a boat cannot be exceeded at any time:

$$\sum_{j|i \neq j} s_j x_{i,j,t} \leq c_i h_i \quad \text{ for all } i, t$$

7. No crews is "idle": A crew is either hosting a party, or visiting a party. So, if a crew j is hosting a party then $h_j = 1$ otherwide it is visiting another crew i, that is $\sum_i x_{i,j,t} = 1$. Hence we have:

$$h_j + \sum_i x_{i,j,t} = 1 \quad \text{ for all } j, t$$

8. A crew j is visiting another crew i at most once:

$$\sum_t x_{i,j,t} \le h_i \qquad \text{for all } i, j (i \ne j)$$

9. Certain boats (number 1, 2, and 3) must be hosts:

$$h_i = 1 \qquad \text{for } i = \{1, 2, 3\}$$

10. Certain boats (number 40, 41, and 42) must be guests (there capacity is zero):[1]

$$h_i = 0 \qquad \text{for } i = \{40, 41, 42\}$$

11. Guest crews cannot meet more than once. that means, over all periods t and over all hosts i, $x_{i,j,t}$ and $x_{i,k,t}$ cannot be true at the same time. Hence, a correct formulation is:

$$\sum_{i,t|i \ne j, i \ne k} (x_{i,j,t} \wedge x_{i,k,t}) \le 1 \qquad \text{for all } j, k (j \ne k)$$

Unfortunately, this constraints are difficult to solve. An alternative is to introduce a variable $m_{j,k,t}$ which is 1 if crews j and k meet at a party at time t (otherwise the variable is free). We then replace the constraints by:

$$m_{j,k,t} \ge x_{i,j,t} + x_{i,k,t} - 1 \qquad \text{for all } i, j, k, t, (j < k, i \ne j, i \ne k)$$
$$\sum_t m_{j,k,t} \le 1 \qquad \text{for all } j, k, (j < k)$$

The inequalities are interpreted as follows: If crew j and k meet at a boat i in t then it is clear that $x_{i,j,t}$ and $x_{i,k,t}$ are both true and then $m_{j,k,t} \ge 1$ (according to the first constraint). Since the sum over all periods of the $m_{j,k,t}$ cannot exceed 1 (according to the second constraint), the condition that j and k meet *only once* is fullfilled. On the other side, if guest crew j and k never meet, then $m_{j,k,t}$ will be zero (according to the first constraint), and the second is certainly fullfilled too.

12. Minimize the number of host boats

$$\min \sum_i h_i$$

The complete model code in LPL for this model is as follows (see [2]):

Listing 1: The Model

```
model party "Progressive Party Problem";
  set i,j,k  "boat or crew number";
    t        "time slot";
  parameter
    C{i}  "max number of guests";
    s{i} "crew size";
    c{i} "gueast cap";
```

[1] In the initial problem there were 39 boats, three boats were created vitually to allow the teens to visit as "guests" other boats

```
  variable
    m{j,k,t|j<k} [0..1]  "Crews j and k meet at time t";
    binary x{i,j,t|i<>j} "Crew j visits i at time slot t";
           h{i}          "Boat i is a host boat";
  constraint
    H{i,j,t|i<>j}: x <= h;
    Ca{i,t}: sum{j|i<>j} s[j]*x <= c*h;
    Ch{j,t}: h + sum{i} x = 1;
    V{i,j|i<>j}: sum{t} x <= h;
    Ho{i|i<=3}: h=1;
    Gu{i|i>=40}: h=0;
    L{i,j,k,t|j<k and i<>j and i<>k}:
      m[j,k,t] >= x[i,j,t] + x[i,k,t] - 1;
    M{j,k|j<k}: sum{t} m <= 1;
  minimize obj: sum{i} h;

  parameter p; q; X; Y;
  Draw.Scale(30,30);
  {i,t|h} (p:=0,
    if(t=1,(q:=q+1, X:=((q+1)%2)*15, Y:=Ceil(q/2),
            Draw.Text('H'&i,-2+X,Y*#t+t+.75))),
    Draw.Text('t='&t,-1+X,Y*#t+t+.75),
    Draw.Rect(X,Y*#t+t,c,.75,2,0),
    {j|x} (Draw.Rect(j&'('&s[j]&')',p+X,Y*#t+t,s[j],.75,1,0),
           p:=p+s[j]));
  Write{j|~h}('%4s: %3s\n', j, {t,i|x} i);
  model data;
    i := [1..42];   t := [1..6];
    C{i}:=[6 8 12 12 12 12 12 10 10 10 10 10 8 8 8 12 8 8 8 8 8 8 7 7 7
        7 7 7 6 6 6 6 6 6 6 6 6 6 9 0 0 0];
    s{i}:=[2 2  2  2  4  4  4  1  2  2  2  3 4 2 3  6 2 2 4 2 4 5 4 4 2
        2 4 5 2 4 2 2 2 2 2 2 4 5 7 2 3 4];
    c{i} := Max(C-s , 0);
  end
end
```

Solution: A solution with 13 host boats is show in Figure 1. The Figure lists the 13 hosts and their guest crew in each period. The horizontal extension is the capacity dimension and the number in parentheses is the crew size. Note that the host itself is deduced from the capacity. The grey part shows the capacity left on the host boat. For example, the host 3 (with a total guest capacity of 10) receives guest crews 15 (size: 3), 32 (size: 2), 33 (size: 2), and 35 (size: 2) in the first time slot. There is capacity left for one more person.

Table 1 gives a view from the guest crews. All guest crews visit 6 different hosts in the six time slots. For example, the guest crew 13 visits the host 7 in the first and then the host 3 in the second time slot, etc.

References

[1] Kalvelagen E. On Solving The Progressive Party Problem as a MIP. *GAMS Development Corp., Washington DC*, 2001.

[2] T. Hürlimann. Reference Manual for the LPL Modelling Language, most recent version. www.virtual-optima.com.

Guest	Host t=1	Host t=2	Host t=3	Host t=4	Host t=5	Host t=6
13	7	3	9	4	6	8
14	4	8	16	11	2	3
15	3	9	2	4	12	11
17	12	3	8	11	9	16
18	9	4	1	6	2	11
19	2	10	16	8	3	7
20	2	11	8	7	6	5
21	16	10	3	11	8	5
22	4	9	7	8	5	12
23	11	3	6	5	1	4
24	7	2	11	3	16	6
25	1	16	7	4	2	9
26	8	1	3	16	11	4
27	5	6	12	3	11	16
28	12	7	4	6	3	2
29	11	1	4	2	9	6
30	6	16	8	2	10	1
31	1	6	5	16	7	8
32	3	2	1	9	5	4
33	3	7	6	1	11	12
34	11	4	9	16	12	7
35	3	11	4	10	16	7
36	4	12	9	1	7	5
37	5	8	11	9	7	10
38	10	4	5	7	8	9
39	8	5	10	12	4	3
40	16	8	12	10	6	11
41	9	12	2	10	4	8
42	6	11	3	5	9	10

Table 1: Guests at the Hosts

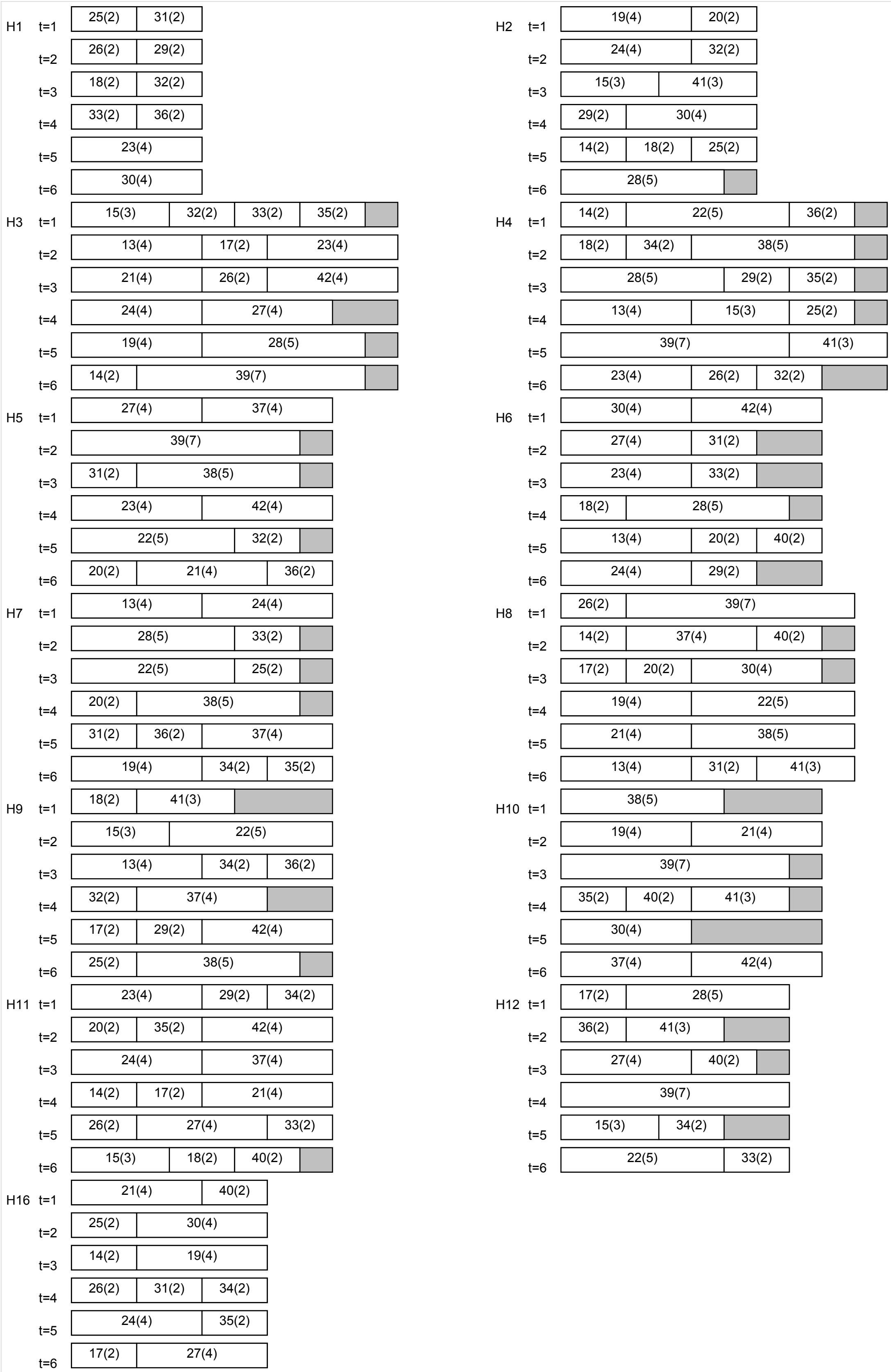

Figure 1: Guest to Host Assigment