How to construct some simple sporting competition schedules

(Associate Prof. David Panton
School of Mathematics & Statistics, University of South Australia)

- Many sporting competitions operate as *round robins*, meaning that each team plays each other team exactly once. Since events are usually played at a *home* or *away* venue match schedules are often organised do that the teams play each other twice, once at home and once away. This is called a *double round robin.* In competitions where the number of teams and the physical nature of the game does not permit this, for example the AFL, we have to be content with a *partial round robin* in which only some of the return fixtures are played. For example in the AFL there are 16 teams and in the first round robin there are 15 rounds, since each team must play each other team exactly once. It is not feasible to have the full 30 rounds required in order that each team plays each other twice, once at home and once away.

- Here is an example of a round robin schedule with 6 teams and 5 rounds. Note that each team has a number and is represented in a *row*, while each round also has a number and is represented in a *column.* This is an example of a Basic Match Schedule (BMS) in which no team names have been assigned to the numbers.

Team	Round				
1	+2	-4	+6	+3	-5
2	-1	+3	-5	-6	+4
3	+5	-2	+4	-1	+6
4	+6	+1	-3	+4	-2
5	-3	-6	+2	-5	+1
6	-4	+5	-1	+2	-3

  The + alongside a team number says that the team is playing at home. For example in line 1 team 1 plays team 2 at home in round 1. A - means that the team is playing away. So in line 3 team 3 plays 2 away (at team 2's ground) in round 2.

- Many other constraints apply when constructing a schedule of sporting events of this type. Some of these relate to the structure of the schedule and are for example:

  1. each team must play each other team exactly once;
  2. there should be a roughly equal number of home (H) and away (A) games. For example with 6 teams there are 5 rounds and you would require either 2 or 3 H games and 3 or 2 A games;
  3. In each round you need to have the same number of H and A games;

- A number of other constraints may apply. Some of these relate to the quality of the schedule. For example we do not want to have too many (if any) home or away games in a row. These are called *breaks.* For example H H H or A A A are breaks of length 2. These should be avoided if possible, however it is not usually possible to avoid breaks of length 1, ie H H or A A. Other constraints relate to special events or special requests from teams. For example in the AFL, Carlton and Essendon always play each other on ANZAC day. In some cases grounds might not be available on certain days. In the Australian Soccer competition they do not want Melbourne teams to play on the weekend of the AFL grand final etc etc. These constraints make the construction of a good schedule difficult.

- How do we go about constructing these schedules? If you try even a small one like the one shown above, you will find that it is not easy. As the number of teams increase the task becomes exponentially harder. A number of mathematically based methods can be used to construct these schedules. While the methods used are beyond year 12 mathematics there are some relatively simple techniques which are fun to play around with.

- One of the first things you do when trying to solve a hard problem is to try to reduce it to a number of smaller problems, each of which are easier to solve. In this case we can first try to find what is called a *Home and Away Schedule (HAS)* which simply consists of the home and away pattern for all teams and rounds. An example of a HAS for the schedule of 6 teams and 5 rounds shown above is,

Team	Round				
1	1	0	1	1	0
2	0	1	0	0	1
3	1	0	1	0	1
4	1	1	0	1	0
5	0	0	1	0	1
6	0	1	0	1	0

  although this is one of many possible HASs which you can generate. Try it! Notice that two possible lines (in this case for teams 3 and 6) have alternating home and way games. It would be nice if these could be used for all the teams, but this isn't possible. Why? Notice also that breaks cannot be avoided when you have an even number of teams. See for example teams 1, 2, 4, and 5. The good news is that for an odd number of teams it is possible to easily generate a HAS schedule with no breaks. Remember that for an odd number of teams, in each round one team must have a *bye*. An example of a HAS for five teams is then

Team	Round				
1	B	1	0	1	0
2	0	B	1	0	1
3	1	0	B	1	0
4	0	1	0	B	1
5	1	0	1	0	B

  Can you see an easy way to create this? In this case why do we have the same number of rounds as teams?

- The next task is to somehow assign team numbers to each of the home and away games, as shown in the example above. Let's consider the example above with five teams. The method below will always work with an odd number of teams. Figure 1 shows a circle with the five team numbers equally spaced on the circle. In round 1 the only teams pairings possible are $(2,3)$, $(2,5)$, $(3,4)$, or $(4,5)$. Why? In figure 1 we have a line joining teams 2 and 5 and a line joining teams 3 and 4. These are two possible *pairings*. We now label the teams with either B (for bye) or H (for home) or A (for away). Note that each pairing must link a home team with an away team. To get all the other legal pairings for rounds 2 to 5 all we have to do is to rotate these parallel lines and the position of the bye clockwise as shown in figures 2 to 5. You can now complete the schedule by replacing the 1s and 0s in the HAS above to give the Basic Match Schedule (BMS) below:

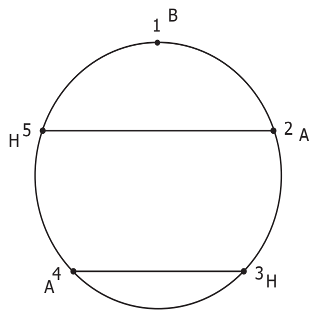

Figure 1: Pairing for round 1

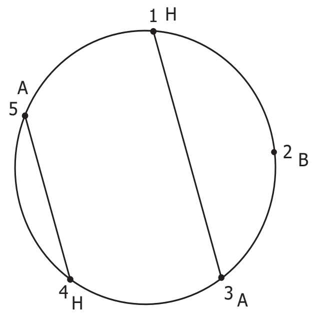

Figure 2: Pairing for round 2

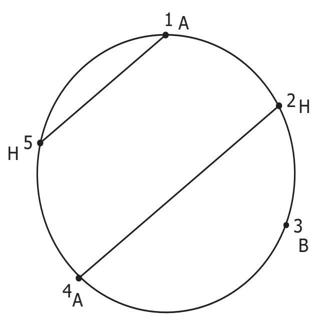

Figure 3: Pairing for round 3

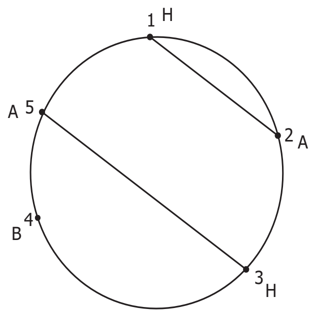

Figure 4: Pairing for round 4

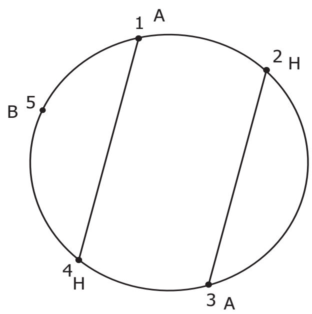

Figure 5: Pairing for round 5

Team	Round				
1	B	3	-5	2	-4
2	-5	B	4	-1	3
3	4	-1	B	5	-2
4	-3	5	-2	B	1
5	2	-4	1	-3	B

- Try this out for a seven team competition.
- The last but important step is to assign team names to the numbers. This is where you will need to start dealing with the real constraints.
- For an even number of teams the process is not quite so easy. Consider a simple four team competition where in a round robin we need 3 rounds. For an even number of teams it is impossible to avoid breaks, and a number of HASs may be possible. For example:-

Team	Round		
1	1	0	1
2	0	1	0
3	0	1	1
4	1	0	0

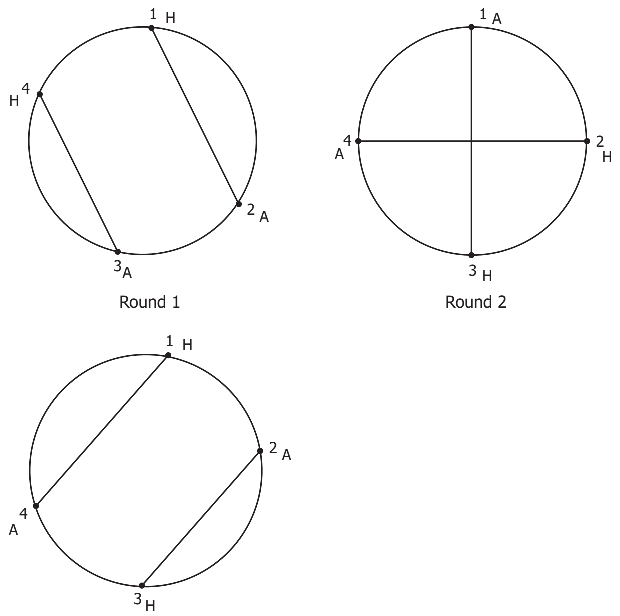

Figure 6: Pairing for four team example

- Note the breaks for teams 3 and 4. Assigning teams to the home and away games can be done using the wheel method, but this time it is not possible to find parallel pairings. In fact it is not always possible to find a BMS for a given HAS. Figure 6 shows three pairings which work for this HAS. Can you find others?

- Suppose now that the HAS is changed to:-

Team	Round		
1	1	0	1
2	0	1	0
3	1	1	0
4	0	0	1

- Can you find a schedule in this case?