Designing Schedules for Leagues and Tournaments

Jeffrey H. Dinitz
Mathematics and Statistics
University of Vermont
Burlington VT,
USA 05405

Abstract

In this note, I will summarize the one hour talk that I gave at *Graph Theory Day* 48, held at Mount Saint Mary College on Saturday, November 13.

The talk covered several main topics. I first spoke about my experience designing the schedule for the *XFL* Football League in 2000. I then showed how to construct the patterned tournament and how to construct a balanced tournament design from this patterned tournament. Then I spoke about schedules with more balance conditions and the connection between these schedules and certain higher dimensional arrays. Next I discussed how to assign the minimum number of referees to a round robin tournament schedule and I ended with a construction of a schedule for a league with 39 golfers playing in threesomes that I originally constructed for my mother-in-law, Joyce Cook.

1 The XFL schedule

In March of 2000, my friend and colleague Dalibor Froncek told me that he had heard that there was a new professional football league being formed, it was called the *XFL*. He said that if they are new, then they must need a schedule.

I called the general manager of the league, Rich Rose, and offered our services to him. On March 2, we sent a letter to Rich describing our qualifications and proposing that we be hired to construct the schedule of play for the new league. On March 8, we received a call from him confirming that the *XFL* was willing to hire us and that they were even willing to pay us for the schedule. It was agreed that he would send a list of requirements for the league schedule and that we would construct the best possible schedule satisfying these requirements.

Here are the guidelines and preferences as given to us by the *XFL*.

Guidelines

- There are two divisions *East* and *West* with four teams in each division.

- Each team will play ten games.
- Each team will play every other team in its own division twice.
- Each team will play every team in the other division once.
- Each team will have five home games and five away games.
- Every team will have one home and one away game against every team in its own division.
- Each team will have two home games and two away games against the four teams from the other division.

Preferences

- Teams that have non-divisional games along opposite coasts will play both non-divisional games in consecutive weeks to avoid scheduling competitive advantages.
- No three-week-long road trips
- Every team should have at least one home game by the third week of the season.

One can indeed see that some of the Guidelines and Preferences are redundant. For instance, Preference 3 is included in Preference 2. Nevertheless, we extracted the necessary information from the Guidelines and Preferences and started working.

At this point in the talk I defined the notions of a round robin tournament on n players, $RR(n)$, and a bipartite tournament with two teams each having n players, denoted by $BT(2n)$.

Definition. A *round-robin tournament* on $2n$ players is a tournament that consists of n games (each between two players) per round for $2n-1$ rounds in which each player plays each other player exactly once. A *bipartite tournament* on $2n$ players first partitions the players up into 2 *teams* (or divisions). The tournament then consists of n games per round for n rounds where each game is between two players from different teams. This tournament then satisfies the property that each player plays every player on the other team exactly once.

So we see from the guidelines above that that the XFL was requiring us to provide a tournament that consisted of an intradivisional double round robin tournament (with 4 teams) and an interdivisional bipartite tournament (4 teams in each division). In the talk, I then constructed a simple example of this and even showed that it satisfied Guideline 5 above, namely that each team had 5 home and 5 away games.

We sent the *XFL* several nice schedules and thought that our job with them was completed. A discussion of these schedules as well as a more in-depth discussion of the construction of the schedules can be found in [7]. Several months passed and we heard from the *XFL* again, but this time they had much more specific properties that the schedule needed to satisfy. For example, because the Chicago Auto Show was using Soldier Field, they needed Chicago to be *away* in Week 2 and Week 3. They also wanted Chicago to play Orlando in Orlando in Week 1 and New York to play at Las Vegas in Week 1. It was interesting working in a "real world" situation like this. Each time we would send them a nice schedule, they would thank us and then add a few more constraints that they had neglected to tell us about before. It was certainly different than just solving a well defined mathematical problem. Well, we did eventually come up with a very nice schedule, which we named X_5 and which they finally did indeed adopt.

I showed slides of the X_5 schedule as well as a photo of Dalibor and me at the *XFL* Championship Game (the *Million Dollar Game*). I also showed a slide of an article [2] in the *New York Times* about our work with the XFL. This article can also be accessed from my web page (http://www.emba.uvm.edu/ dinitz/). I lamented that it was unfortunate that the *XFL* folded after just one year (it was certainly my 15 minutes of fame). I told of discussion that I had with the president of the league, Basil Devito, while standing at midfield of the L.A. Coliseum just after the championship game ended. He said to me "Jeff, the only thing about the league that *nobody* every complained about was the schedule". I certainly felt good about that.

2 The patterned tournament and balanced tournament designs

In this part of the talk I discussed a particularly nice way to construct a round-robin tournament and showed how to then determine sites for the games so that the tournament is balanced for sites. We now give this construction. Let Z_n denote the cyclic group of order n.

Definition. The set of pairs $P = \{\{x, -x\} : x \in Z_{2n-1}, x \neq 0\}$ is called the *patterned starter* in the cyclic group of order $2n-1$. Notice that this only accounts for $2n-2$ players. We add another player, called ∞ and define $S_0 = P \cup \{0, \infty\}$.

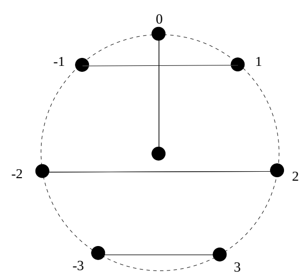
The n pairs in S_0 can be viewed as the games played in the first week of the round-robin tournament with $2n$ players. In general, for $0 \leq i \leq 2n-2$ the games played in the i^{th} week are the pairs in the set

$$S_i = S_0 + i = \{\{x+i, -x+i\} : x \in Z_{2n-1}, x \neq 0\} \cup \{i, \infty\}.$$

It not too hard to check that the set $\{S_0, S_1, \ldots, S_{2n-2}\}$ does indeed form the weeks of a round robin tournament on $2n$ players. The key point to note is that in P, every nonzero element of Z_{2n-1} occurs exactly once, and furthermore that every nonzero element of Z_{2n-1} also appears exactly once as a difference between the elements in the pairs in P. These two properties will guarantee that $\{S_0, S_1, \ldots, S_{2n-1}\}$ forms a round-robin tournament. Also notice that the pairs that are together in week i are those that *sum* to $2i$ modulo $2n-1$ (and also the pair $\{i, \infty\}$).

The round-robin tournament defined above is called the *patterned tournament* on $2n$ players or in graph theoretic terms it is the *patterned one-factorization of* K_{2n}. In Example 2.1 below the games of the patterned tournament on 8 players are listed. Each row is a week of the tournament.

This tournament can be nicely visualized. In the diagram below (shown here for $2n = 8$) the edges represent the games in the 0^{th} round (the pairs in S_0). We associate the symbol ∞ with the vertex in the center of the circle and note that the symbols are elements of Z_7 (so $-1 = 6, -2 = 5$, and $-3 = 4$). Subsequent rounds are obtained by rotating the diagram. This is equivalent to adding 1 to each element in each pair of the prior round (noting that $\infty + 1 = \infty$).



Note that this tournament is easily adaptable to having an odd number of players. Merely delete player ∞. In this way in week i player i will have a bye week and still throughout the tournament, each player will play each other player exactly once.

Now from the patterned tournament we can design a schedule of play for 8 teams at 4 sites. In the example below the rows represents the weeks, while the columns represent the sites.

Example 2.1 *The patterned tournament on 8 players at 4 sites. The sites (columns) are labeled by the differences between the elements in the pairs contained in the column. So in this case the columns are* $\infty, \pm 1, \pm 2$ *and* ± 3*, respectively.*

	∞	± 1	± 2	± 3
week 0	$0, \infty$	3,4	6,1	2,5
week 1	$1, \infty$	4,5	0,2	3,6
week 2	$2, \infty$	5,6	1,3	4,0
week 3	$3, \infty$	6,0	2,4	5,1
week 4	$4, \infty$	0,1	3,5	6,2
week 5	$5, \infty$	1,2	4,6	0,3
week 6	$6, \infty$	2,3	5,0	1,4

We can see that the balance on the sites is not very good as team ∞ plays *all* its games at site ∞. Nonetheless, it is a nice easy construction for a round-robin tournament on 8 players played at 4 sites, and in general for a round-robin tournament on $2n$ players played at n sites.

We now wish to get the *sites* as balanced as possible. The best we can do is have every player play at each site exactly twice, except for one site where they only play once. The following definition is for such a tournament.

Definition. A *balanced tournament design*, BTD(n), defined on a $2n$-set V is an arrangement of the pairs of elements in V into a $2n-1 \times n$ array such that:

1. every element of V is contained in precisely one cell of each row,
2. every element of V is contained in at most two cells in any column,

Example 2.2 *A BTD(5). Note again that the weeks of the tournamenent are the rows, while the columns represent the sites. Here we have 10 players (labeled 0 – 9).*

8 4	9 3	5 6	1 2	0 7
9 2	8 5	0 3	4 7	1 6
1 3	4 6	8 7	9 0	2 5
5 7	0 2	9 1	8 6	3 4
0 6	1 7	4 2	5 3	8 9
2 3	9 4	6 7	8 0	1 5
4 5	8 2	0 1	9 6	3 7
9 7	0 5	8 3	1 4	2 6
8 1	6 3	9 5	7 2	0 4

I will now describe an easy construction of a balanced tournament design from the patterned tournament on $2n$ players. This construction works when $2n \equiv 0$ or $2 \pmod 3$.

By example, let $2n = 14$ (all arithmetic mod 13). We begin with the patterned tournament on 14 teams with the columns reordered as indicated by the column headings.

∞	± 2	± 6	± 3	± 1	± 5	± 4
$\infty, 0$	12,1	10,3	5,8	6,7	4,9	11,2
$\infty, 1$	0,2	11,4	6,9	7,8	5,10	12,3
$\infty, 2$	1,3	12,5	7,10	8,9	6,11	0,4
$\infty, 3$	2,4	0,6	8,11	9,10	7,12	1,5
$\infty, 4$	3,5	1,7	9,12	10,11	8,0	2,6
$\infty, 5$	4,6	2,8	10,0	11,12	9,1	3,7
$\infty, 6$	5,7	3,9	11,1	12,0	10,2	4,8
$\infty, 7$	6,8	4,10	12,2	0,1	11,3	5,9
$\infty, 8$	7,9	5,11	0,3	1,2	12,4	6,10
$\infty, 9$	8,10	6,12	1,4	2,3	0,5	7,11
$\infty, 10$	9,11	7,0	2,5	3,4	1,6	8,12
$\infty, 11$	10,12	8,1	3,6	4,5	2,7	9,0
$\infty, 12$	11,0	9,2	4,7	5,6	3,8	10,1

Note again that every player is in every non-∞ site exactly twice, but that player ∞ plays every game at site ∞. We wish to switch the cells from the ∞ column with cells from a non-∞ column in the same row. Our goal is that each non-∞ column receives two of the ∞ cells. Such a pattern was first found by Robert Gray in 1977 and reported in Hasselgrove and Leech [12]. The pattern is as follows: in week i ($i \neq n-1$) switch the pairs $\{\infty, i\}$ and $\{3i+1, -i-1\}$.

Notice that in week i, the pair with the difference $\pm((3i+1)-(-i-1)) = \pm(4i+2)$ is switched with the pair $\{\infty, i\}$ (this prompted our relabeling of the columns of the original array). We also check that the sum of the pairs that get switched is $(3i+1)+(-i-1) = 2i$ so this is indeed a pair that plays in week i. This gives a very nice pattern for the switches when the columns have been reordered as above. We observe this pattern in the next example.

Here is the pattern of switches (the pairs to be switched with $\{\infty, i\}$ in row i are in boldface):

∞	± 2	± 6	± 3	± 1	± 5	± 4
$\infty, 0$	**12,1**	10,3	5,8	6,7	4,9	11,2
$\infty, 1$	0,2	**11,4**	6,9	7,8	5,10	12,3
$\infty, 2$	1,3	12,5	**7,10**	8,9	6,11	0,4
$\infty, 3$	2,4	0,6	8,11	**9,10**	7,12	1,5
$\infty, 4$	3,5	1,7	9,12	10,11	**8,0**	2,6
$\infty, 5$	4,6	2,8	10,0	11,12	9,1	**3,7**
$\infty, 6$	5,7	3,9	11,1	12,0	10,2	4,8
$\infty, 7$	6,8	4,10	12,2	0,1	11,3	**5,9**
$\infty, 8$	7,9	5,11	0,3	1,2	**12,4**	6,10
$\infty, 9$	8,10	6,12	1,4	**2,3**	0,5	7,11
$\infty, 10$	9,11	7,0	**2,5**	3,4	1,6	8,12
$\infty, 11$	10,12	**8,1**	3,6	4,5	2,7	9,0
$\infty, 12$	**11,0**	9,2	4,7	5,6	3,8	10,1

We note that after each bold pair in row i is switched with the pair $\{\infty, i\}$ each row will be unchanged and every column will contain each symbol either 1 or 2 times. The final tournament (a BTD(7)) is given below.

∞	± 2	± 6	± 3	± 1	± 5	± 4
12,1	$\infty, 0$	10,3	5,8	6,7	4,9	11,2
11,4	0,2	$\infty, 1$	6,9	7,8	5,10	12,3
7,10	1,3	12,5	$\infty, 2$	8,9	6,11	0,4
9,10	2,4	0,6	8,11	$\infty, 3$	7,12	1,5
8,0	3,5	1,7	9,12	10,11	$\infty, 4$	2,6
3,7	4,6	2,8	10,0	11,12	9,1	$\infty, 5$
$\infty, 6$	5,7	3,9	11,1	12,0	10,2	4,8
5,9	6,8	4,10	12,2	0,1	11,3	$\infty, 7$
12,4	7,9	5,11	0,3	1,2	$\infty, 8$	6,10
2,3	8,10	6,12	1,4	$\infty, 9$	0,5	7,11
2,5	9,11	7,0	$\infty, 10$	3,4	1,6	8,12
8,1	10,12	$\infty, 11$	3,6	4,5	2,7	9,0
11,0	$\infty, 12$	9,2	4,7	5,6	3,8	10,1

Unfortunately this construction fails when $2n \equiv 1 \pmod 3$ and there is no known direct construction for a BTD in this case. However, using recursive constructions and other techniques from combinatorial design theory it is not too hard to make BTD's for all orders. This was first proven by Schellenberg, van Rees and Vanstone [18] in 1977. We state this theorem below.

Theorem:[18] There exists a BTD(n) for all positive integers $n \neq 2$.

We next consider BTD's on $2n$ players which satisfy a *very* restrictive property, namely that every player plays at every site exactly once in the first n weeks and exactly once in the last n weeks. These are called partitioned balanced tournament designs. Here is the formal definition.

Definition. A BTD is a *partitioned* balanced tournament design (PBTD) if it also satisfies:

1. in the first n rows, each element of V occurs in each column *exactly once,*
2. in the last n rows, each element of V occurs in each column *exactly once.*

The following example is of a partitioned BTD on 10 players. Notice that in the first 5 rows (weeks) every symbol occurs exactly once in each column and that this also holds true for the last 5 weeks.

Example 2.3 *A PBTD(5).*

8 4	9 3	5 6	1 2	0 7
9 2	8 5	0 3	4 7	1 6
1 3	4 6	8 7	9 0	2 5
5 7	0 2	9 1	8 6	3 4
0 6	1 7	4 2	5 3	8 9
2 3	9 4	6 7	8 0	1 5
4 5	8 2	0 1	9 6	3 7
9 7	0 5	8 3	1 4	2 6
8 1	6 3	9 5	7 2	0 4

It is much more difficult to construct partitioned BTD's than BTD's. However in a series of papers, Lamken [13, 14] has proven the existence of PBTD's for all but three possible orders. We have:

Theorem 2.4 *(Lamken 1987, 1996) There exists a PBTD(n) for all positive integers* $n \geq 3$*, with the possible exceptions of* $n = 9, 11$ *and* 15*.*

We will have a use for these PBTD's later in this paper.

2.1 Some enumeration results

In this section we mention some results concerning the enumeration of round-robin tournaments and BTD's. Two round-robin tournaments on $2n$ players (one-factorizations of K_n) are *isomorphic* if one can be obtained from the other by interchanging weeks or symbols. The exact number of nonisomorphic round robin tournament on $2n$ players is known only up to $2n = 12$. Table 1 gives these values as well as a estimate on the number of nonisomorphic round-robin tournaments on 14 and 16 players. As is evident, this number grows very rapidly. It is a great example of *combinatorial explosion*.

$2n$	number	reference
2,4,6	1	
8	6	Dickson,Safford, 1906
10	396	Gelling, 1973
12	526,915,620	Dinitz,Garnick,McKay [8], 1994
14	1.132×10^{18} (est.)	Dinitz,Garnick,McKay [8], 1994
16	7.07×10^{30} (est.)	Dinitz,Garnick,McKay [8], 1994

Table 1. The number of nonisomorphic round-robin tournaments on $2n$ players

We also give some results on the number of nonisomorphic BTD's of order $2n \leq 10$.

$2n$	number	reference
2,6	1	
8	47	Corriveau [4], (1988)
10	30,220,557	Dinitz, Dinitz [6], (2005)

Table 2. The number of nonisomorphic BTD(n) ($2n$ players)

We note that even though there are over 30 million BTD(5)'s that exactly *two* of them are partitioned BTD's. It is clear that each round robin tournament on $2n$ players gives many BTD(n)'s (i.e keep the rows as the weeks and arrange the columns to make the BTD). In particular for 10 players there are exactly 396 round robin tournaments, while there are 30,220,557 BTD's. We have the following information from [6] about the connection between the number of round robin tournaments and the number of BTD's in the case of 10 players.

- The greatest number of *distinct* BTD's from any round robin tournament on 10 players is 123,876, the least is 63,504 and the average is 89,998.8.
- The greatest number of *nonisomorphic* BTD's from any round robin tournament on 10 players is 103,912, the least is 293. In general, it is about about 90,000 divided by the order of the automorphism group of the round robin tournament.

3 Court Balanced Tournament Designs

It is certainly possible that there may be fewer than n sites available for a round robin tournament on $2n$ players. However, if certain numerical condidions are satisfied it is still possible to balance the tournament for sites.

Definition. A *Court Balanced Tournament Design*, CBTD(m, c), defined on an $m-$set V is an arrangement of the pairs of elements in V into an $\binom{m}{2}/c$ by c array such that

1. no cell is empty,
2. every element of v occcurs at most once in each row,
3. each element of v appears in the same number of columns, and
4. each pair of elements from V occurs together in exactly one cell of the array.

Here is an example of a court balanced tournament design on 10 players at 3 sites, a CBTD(10,3). Note that each player plays on each court exactly 3 times and that there are 15 weeks in this tournament.

Example 3.1 *A CBTD(10,3)*

1 0	3 6	2 8
1 5	3 7	2 9
2 6	1 8	3 4
2 7	1 9	3 5
3 8	2 4	1 6
3 9	2 0	1 7
2 3	5 8	6 7
5 0	6 9	7 8
6 4	1 3	8 9
7 5	8 4	9 0
8 6	9 5	4 0
7 9	6 0	4 5
8 0	4 7	1 2
1 4	2 5	3 0
4 9	7 0	5 6

Since each player plays $m-1$ games, if they are to play the same number at each court, then necessarily $c|(m-1)$. Also, since the total number of games is $\binom{m}{2}$ and since there are no empty cells it must be true that $c|\binom{m}{2}$. This allows us to compute the number of weeks for such a tournament as $\binom{m}{2}/c$. In 1994, it was shown [16] that these necessary condition also turns out to be sufficient.

Theorem 3.2 *(Mendelsohn, Rodney [16]) There exists a CBTD($2n, c$) if and only if $c|\binom{2n}{2}$, $c|(2n-1)$ and $1 \leq c \leq n$.*

4 Room squares and n-dimensional Room cubes

Room Squares are well studied objects in the area of combinatorial design theory. In the context of this paper, they provide another way to balance sites (and other measures) in a round-robin tournament. We begin with a definition.

Definition. Let n be an odd integer and let S be a set of size $n+1$ called *symbols*. A *Room square* of side n (denoted RS(n)) based on symbol set S is an $n \times n$ array, F, which satisfies the following properties:

1. every cell of F either is empty or contains an unordered pair of symbols from S,

2. every symbol $x \in S$ occurs once in every row and once in every column of F,

3. every unordered pair of symbols occurs in exactly one cell of F.

It is known that a Room square of side n exists if and only if n is an odd integer, $n \geq 1$, $n \neq 3, 5$. See Mullin and Wallis [17] for a proof. For additional information about Room squares and related structures, see [9] or [3]. A Room square of side 7 on the symbol set $\{\infty, 0, 1, \ldots 6\}$ is given in Example 4.1 below.

Example 4.1 *A Room square of side 7.*

∞0	3 4	6 1		5 2		
	∞1	4 5	0 2		6 3	
		∞2	5 6	1 3		0 4
1 5			∞3	6 0	2 4	
	2 6			∞4	0 1	3 5
4 6		3 0			∞5	1 2
2 3	5 0		1 4			∞6

A Room square of side n, say F, can be used to schedule a round robin tournament for $n+1$ teams. Again we let the rows of F be indexed by the n weeks (rounds), and the columns of F are indexed by n sites. The round robin tournament then satisfies the following properties:

1. every team plays once in every round and once at each site,
2. every pair of teams plays together exactly once during the tournament.

Notice that the round robin tournament formed from the rows of the Room square of side 7 in Example 4.1 is again the *patterned* round robin tournament (on 8 players). Also notice that this tournament is now balanced on sites as each team playes exactly once at each of the 7 sites.

Now suppose that we wish to add another balance condition, say for example referees. In the case of a round robin tournament on $2n$ players we would be asking that it satisfies properties 1 and 2 above and in addition, the following property:

3. there are $2n-1$ referees and each team sees each referee exactly once.

The following example describes a solution to this problem in the case of 8 players.

Example 4.2 *A schedule of play for a round robin tournament on 8 players at 7 sites with 7 referees so that each player plays exactly once at each site and sees each referee exactly once.*

	weeks				sites				referees			
0	∞0	16	25	34	∞0	13	26	45	∞0	15	23	46
1	∞1	20	36	45	∞1	24	30	56	∞1	26	34	50
2	∞2	31	40	56	∞2	35	41	60	∞2	30	45	61
3	∞3	42	51	60	∞3	46	52	01	∞3	41	56	02
4	∞4	53	62	01	∞4	50	63	12	∞4	52	60	13
5	∞5	64	03	12	∞5	61	04	23	∞5	63	01	24
6	∞6	05	14	23	∞6	02	15	34	∞6	04	12	35

To construct the tournament from the above, merely choose a pair of players and determine which week they play and at which site and with which referee. For example, say the pair is $\{5, 6\}$. We see that they play in week 2 at site 1 and have referee 3. It can be checked that this is indeed a schedule which satisfies our three balance conditions (weeks, sites, referees).

A *Room cube of side* n is a 3 dimensional array with the property that each of the 2-dimensional projections are Room squares of side n. It is interesting to note that a schedule satisfying three balance conditions is equivalent to a Room cube by the following (reversible) construction: put pair $\{i, j\}$ in cell (w, s, r) if pair $\{i, j\}$ plays in week w at site s and with referee r. So from Example 4.2 we can construct a Room cube of side 7.

Clearly there is no reason that one can't ask for even more balance conditions to be satisfied. (Although realistically it is getting a little far-fetched). The following example on 10 players adds another balance condition, say time-of-day.

Example 4.3 *A schedule of play for a round robin tournament on 10 players at 9 sites with 9 referees at 9 different times-of-day so that each player plays exactly once at each site, sees each referee exactly once and plays at each time-of-day exactly once.*

	weeks					sites					referees					times-of-day				
1	01	23	45	67	89	01	29	36	48	57	01	26	39	47	58	01	25	34	68	79
2	02	13	46	58	79	02	15	34	69	78	02	14	37	56	89	02	18	35	49	67
3	03	12	47	59	68	03	16	28	45	79	03	17	25	48	69	03	15	27	46	89
4	04	16	25	39	78	04	17	26	35	89	04	18	27	36	59	04	13	28	57	69
5	05	18	24	37	69	05	14	27	39	68	05	19	28	34	67	05	16	29	38	47
6	06	19	27	35	48	06	12	37	39	58	06	15	24	38	79	06	14	23	59	78
7	07	15	28	36	49	07	19	25	38	46	07	13	29	45	68	07	12	39	48	56
8	08	17	29	34	56	08	13	24	59	67	08	16	23	49	67	08	19	26	37	45
9	09	14	26	38	57	09	18	23	47	56	09	12	35	46	78	09	17	24	36	58

To construct the tournament from the above, again merely choose a pair of players and determine which week, site, referee and time-of-day they play. For example, say the pair $\{1, 9\}$. We see that they then play in week 6 at site 7 and have referee 5 and at the 8^{th} time-of-day. It can be checked that this is indeed a schedule which satisfies our four balance conditions (weeks, sites, referees, and time-of-day).

We can generalize the notion of a Room cube to that of a Room $t-$cube where a Room $t-$cube of side n is a t dimensional array with the property that each of the 2-dimensional projections are Room squares of side n. A Room $t-$cube of side n is equivalent to a round robin tournament satisfying t balance conditions. So the example above can be used to construct a Room cube of side 7 and a Room 4-cube of side 9. Theorem 4.4 gives a compilation of the best known existence theorems for Room $t-$cubes. See [9] for the original references for these results

Theorem 4.4 *There exists a Room cube of side 7 (Gross, Mullin, Wallis 1973); there exists a Room 4-cube of side 9 (Dinitz, Wallis 1985); and there exists a Room 5-cube for every odd* $n \geq 11$*, except possibly for* $n = 15$*. (Dinitz 1987). There exists a Room 4-cube of side 15 (Dinitz 1980)*

We are still a long way from knowing exactly the maximum number of conditions of balance that a round robin tournament on $2n$ players can have. The only values that are known explicitly are for $2n \leq 10$. It is conjectured [11] that the maximum number t of conditions of balance in a round robin tournament on $2n$ players satisfies $t \leq n-1$ but this far from being proven. See [9] for further more information on Room squares, Room $t-$cubes and related designs.

5 Assigning referees to round-robin tournaments

The material in this section comes from some recent work with Doug Stinson and can be found in its entirety in [10].

We will again use a Room square F of side n to schedule a rond robin tournament for $n+1$ teams. However, unlike what was done previously we will now let the rows of F be indexed by n playing fields, and the columns of F be indexed by n rounds. The round robin tournament still satisfies the following properties:

1. every team plays once on every field and every team plays once in every round,

2. every pair of teams plays together exactly once during the tournament.

It is easily seen that there are exactly $(n+1)/2$ games in every round, so clearly at least $(n+1)/2$ referees are required for this tournament. Unlike in the previous section where we had n referees for the tournament, in this application we do not want to have referees "sitting around" doing nothing so we wonder just how balanced the tournament can be with exactly $(n+1)/2$ referees. Can it even be scheduled?

In order to eliminate possible bias of referees, we would like to assign referees to games in such a way that every team receives each referee roughly the same number of times. More precisely, for every team T and for every referee R, it should be the case that R is assigned to exactly one or two games involving team T. A Room square for which referees can be assigned in this way will be called a *referee-minimal Room square*, denoted RMRS(n).

Given a Room square F of side n, a *column-transversal* in F is a set of n *filled* cells with the property that no two cells are in the same column and no symbol occurs more than twice in these cells. F will be an RMRS(n) if and only if there exists a set of $(n+1)/2$ disjoint column-transversals in F (each column transveral corresponds to an assigned referee).

Referee-minimal Room squares have a nice three-dimensional interpretation which gives a connection between balanced tournament designs and Room squares. Again for this application we will be considering the *transpose* of the BTD's of the earlier sections, so in this section a BTD(n) will be a $n \times (2n-1)$ array with the rows now representing the sites while the columns are the weeks.

Now, it is not hard to see that an RMRS(n) is equivalent to a three-dimensional "brick", having dimensions $n \times n \times \frac{n+1}{2}$, that satisfies certain conditions. Suppose that we think of the three dimensions of the brick as corresponding to fields, rounds, and referees, respectively. If we collapse the third dimension (i.e., project onto the first two dimensions), then we obtain a Room square of side n. If we collapse the first dimension, then we obtain a BTD($(n+1)/2$).

5.1 A Construction for Referee-minimal Room Squares

We now describe a method of constructing RMRS(n) for almost all odd integers $n \geq 9$. We make use of a special type of Room square called a *maximum empty subarray Room square*, denoted MESRS(n), which was first defined by Stinson [19]. An MESRS(n) is an RS(n) containing an $\frac{n-1}{2} \times \frac{n-1}{2}$ subarray of empty cells. We present an MESRS(9) in Example 5.1.

Example 5.1 *A maximum empty subarray Room square of side* 9*, a MESRS(9).*

37					28	59	$4X$	16
	56				$1X$	47	29	38
		$2X$			67	18	35	49
			48		39	26	17	$5X$
				19	45	$3X$	68	27
12	$8X$	57	69	34				
46	13	89	$7X$	25				
58	79	14	23	$6X$				
$9X$	24	36	15	78				

It is not hard to see that the rows and columns of an MESRS(n), say F, can be permuted so that F has the form

$$\begin{pmatrix} A & B \\ C & D \end{pmatrix},$$

where A has dimensions $\frac{n+1}{2} \times \frac{n+1}{2}$, B has dimensions $\frac{n+1}{2} \times \frac{n-1}{2}$, C has dimensions $\frac{n-1}{2} \times \frac{n+1}{2}$, D has dimensions $\frac{n-1}{2} \times \frac{n-1}{2}$, B and C are filled, D is empty, and the only filled cells in A are the diagonal cells. An MESRS(n) that is displayed in this fashion is said to be in *standard form.* Note that the MESRS(9) in Example 5.1 is in standard form.

Now if one lines up all the filled cells of A in the column just to the left of B (column $\frac{n+1}{2}$), and puts C^T to the left of that column, the resulting $\frac{n+1}{2} \times n$ array is a partitioned balanced tournament design. This construction can be performed in reverse and hence we have the following theorem.

Theorem 5.2 *The existence of a MESRS$(2n-1)$ is equivalent to the existence of a PBTD(n).*

Example 5.3 shows this connection in the case when $n = 5$.

Example 5.3 *The partitioned BTD(5) that is equivalent to the maximum empty subarray Room square of side 9 from Example 5.1*

1 2	4 6	5 8	9 X	3 7	2 8	5 9	4 X	1 6
8 x	1 3	7 9	2 4	5 6	1 X	4 7	2 9	3 8
5 7	8 9	1 4	3 6	2 X	6 7	1 8	3 5	4 9
6 9	7 X	2 3	1 5	4 8	3 9	2 6	1 7	5 X
3 4	2 5	6 X	7 8	1 9	4 5	3 X	6 8	2 7

Given this connection between PBTD's and MESRS, we can appeal again to the result of Lamken [14] on the existence of PBTD's to get the following existence theorem for MESRS.

Theorem 5.4 *Suppose $n \geq 9$ is an odd integer, and $n \neq 17, 21, 29$. Then there exists an MESRS(n).*

Now, suppose we have an MESRS(n) in standard form. Let $m = (n+1)/2$. Suppose that the rows of B are denoted $B_1, \ldots, B_m$, and denote the rows of C by $C_1, \ldots, C_{m-1}$. Let the diagonal of A be denoted A_d. We now assign referees for the games. For $1 \leq i \leq m-1$, referee R_i is assigned to all the games in the cells in $B_i \cup C_i$. Referee R_m is assigned to the games in the cells in $A_d \cup B_m$.

We show that this assignment of referees to games yields an RMRS(n). Every set of cells C_i contains every team exactly once, and every set of cells B_i contains every team at most once. Therefore referees $R_1, \ldots, R_{m-1}$ are assigned to each team either once or twice. In addition, it is not hard to see that the set of cells A_d contains every team exactly once, and hence the desired property holds also for referee R_m. Hence, we have proven the following.

Theorem 5.5 *There exists an RMRS(n) for all odd integers $n \geq 9$ except possibly if $n = 17, 21, 29$.*

5.2 Referee Field Changes

A round robin tournament based on a Room square is set up so that every team plays on a different field during each round. However, there may be no reason why the referees should be required to change fields so often. On the contrary, it might be desirable for the referees to change fields as infrequently as possible.

Here is a small example to illustrate.

Example 5.6 *Consider the RMRS(9) constructed as we have described in Section 5.1. Referees R_1, R_2, R_3 and R_4 each change fields once, and referee R_5 changes fields four times. The total number of field changes is therefore eight.*

In general, if we construct an RMRS(n) as described in Section 5.1, then the total number of referee field changes is $n-1$. In [10] we prove that it is impossible to construct an RMRS(n) in which the total number of referee field changes is less than $n-1$. Hence we have the following theorem about referee field changes in RMRS(n).

Theorem 5.7 *Suppose $n \geq 9$ is an odd integer, and suppose $n \neq 17, 21, 29$. Then there exists an RMRS(n) in which the total number of referee field changes is equal to $n-1$. Moreover, for any odd integer n, there does not exist an RMRS(n) in which the total number of referee field changes is less than $n-1$.*

5.3 An Open Problem

We mentioned a three-dimensional interpretation of RMRS(n) in Section 5. In this interpretation, we have a three-dimensional brick such that one two-dimensional projection yields a Room square and another two-dimensional projection yields a balanced tournament design. It is conceivable that the third two-dimensional projection could also be a balanced tournament design; however, we do not have any examples where this occurs. Thus we pose the following open problem.

> For which odd integers n does there exist a three-dimensional brick B having dimensions $n \times n \times \frac{n+1}{2}$, such that every two-dimensional projection of B is either a BTD($(n+1)/2$) or an RS(n)?

The existence of such a brick is equivalent to the existence of a Room square of side n which contains $(n+1)/2$ disjoint transversals, where each transversal consists of n filled cells with the property that no two cells are in the same row or column and no symbol occurs more than twice in these cells. We were unable to find any example of this object, even for small orders of n. However, we do not hesitate to conjecture that such an object exists for many orders.

6 The Joyce Cook golf league

I ended the talk with a discussion of a league of play for golfers that involves triples, instead of just pairs. These conditions were presented to me by my mother-in-law, Joyce Cook, who needed a schedule of play for her golf league. I was pleased that I could solve her problem as it turned out to be fairly *off the shelf.*

Here are the general conditions that needed to be satisfied:

- 39 golfers play in 13 threesomes (so each player plays every week),
- 13 different starting "places", 7 on the front nine, 6 on the back nine,
- league play lasts for 10 weeks,

Here are the balance conditions:

- no two golfers are together in a threesome more than once,
- no golfer starts in the same place twice.

First note that since each golfer plays with two other golfers per week and there are 10 weeks, that each golfer only plays with 20 of the 38 other golfers, so the tournament schedule is not particularly tight.

In order to construct this schedule we need another object from combinatorial design theory called a latin square.

Definition A *latin square* of side (order) n is an $n \times n$ array in which each cell contains a single element from an $n-$set S, and such that each element occurs exactly once in each row and exactly once in each column.

Example 6.1 *A latin square L of side 3 on symbol set $S = \{1, 2, 3\}$.*

$L =$

1	2	3
2	3	1
3	1	2

We note in passing that the existence of a latin square of side n is equivalent to the the existence of a bipartite tournament schedule for two teams each with n players. The rows represent one team, the columns represent the other team and the cells represent the rounds. More specifically, if $L(r, c) = s$, then person r from team 1 plays person c from team 2 in round s.

Definition Two latin squares L and R (both on the symbol set S) of the same order are *orthogonal* if when superimposed, every ordered pair of symbols from $S \times T$ occurs exactly once.

Example 6.2 *Two orthogonal latin squares L and R of side 3 on symbol set $S = \{1, 2, 3\}$ and the resulting superimposed square $L \times R$.*

$L =$

1	2	3
2	3	1
3	1	2

$R =$

1	2	3
3	1	2
2	3	1

$L \times R =$

1,1	2,2	3,3
2,3	3,1	1,2
3,2	1,3	2,1

Orthogonal latin squares have been studied since they were first constructed by Euler in 1782 in an attempt to solve the *36 officer problem.* There is extensive literature on Latin squares and the interested reader is referred to [5] and to [3] (Part 2). One of the primary questions in this area concerns the existence of sets of pairwise orthogonal latin squares for each order n. It is not hard to show that there are *at most* $n - 1$ pairwise orthogonal latin squares of side

n. However, this upper bound has only been achieved in the case when n is a prime power. Nonetheless, this will be sufficient for our purposes. We have the following theorem in this case.

Theorem 6.3 *When n is a prime power, there exist $n-1$ pairwise orthogonal latin squares of side n.*

We solve our golf scheduling problem by the use of 3 pairwise orthogonal latin squares of side 13. We denote these as A, B and C. These three squares are given in Example 6.4.

Example 6.4 *Three pairwise orthogonal latin squares of side 13.*

$A =$

0	1	2	3	4	5	6	7	8	9	10	11	12
1	2	3	4	5	6	7	8	9	10	11	12	0
2	3	4	5	6	7	8	9	10	11	12	0	1
3	4	5	6	7	8	9	10	11	12	0	1	2
4	5	6	7	8	9	10	11	12	0	1	2	3
5	6	7	8	9	10	11	12	0	1	2	3	4
6	7	8	9	10	11	12	0	1	2	3	4	5
7	8	9	10	11	12	0	1	2	3	4	5	6
8	9	10	11	12	0	1	2	3	4	5	6	7
9	10	11	12	0	1	2	3	4	5	6	7	8
10	11	12	0	1	2	3	4	5	6	7	8	9
11	12	0	1	2	3	4	5	6	7	8	9	10
12	0	1	2	3	4	5	6	7	8	9	10	11

$B =$

0	1	2	3	4	5	6	7	8	9	10	11	12
2	3	4	5	6	7	8	9	10	11	12	0	1
4	5	6	7	8	9	10	11	12	0	1	2	3
6	7	8	9	10	11	12	0	1	2	3	4	5
8	9	10	11	12	13	1	2	3	4	5	6	7
10	11	12	0	1	2	3	4	5	6	7	8	9
12	0	1	2	3	4	5	6	7	8	9	10	11
1	2	3	4	5	6	7	8	9	10	11	12	0
3	4	5	6	7	8	9	10	11	12	0	1	2
5	6	7	8	9	10	11	12	0	1	2	3	4
7	8	9	10	11	12	0	1	2	3	4	5	6
9	10	11	12	0	1	2	3	4	5	6	7	8
11	12	0	1	2	3	4	5	6	7	8	9	10

$C =$

0	1	2	3	4	5	6	7	8	9	10	11	12
3	4	5	6	7	8	9	10	11	12	0	1	2
6	7	8	9	10	11	12	0	1	2	3	4	5
9	10	11	12	0	1	2	3	4	5	6	7	8
12	0	1	2	3	4	5	6	7	8	9	10	11
2	3	4	5	6	7	8	9	10	11	12	0	1
5	6	7	8	9	10	11	12	0	1	2	3	4
8	9	10	11	12	0	1	2	3	4	5	6	7
11	12	0	1	2	3	4	5	6	7	8	9	10
1	2	3	4	5	6	7	8	9	10	11	12	0
4	5	6	7	8	9	10	11	12	0	1	2	3
7	8	9	10	11	12	0	1	2	3	4	5	6
10	11	12	0	1	2	3	4	5	6	7	8	9

Now we can solve our golf scheduling problem. We superimpose these three squares to make a 13×13 array S: Each cell of S contains a triple (i, j, k) where i is from A, j is from B, and k is from C. Our 39 golfers consist of the 13 symbols from A plus the 13 symbols from B plus the 13 symbols from C. We let the weeks of the tournament be the rows of S and the 13 different starting places be the columns. (Note that we even have 3 extra weeks). If the triple in row r, column c is (i, j, k), then in week r, the triple consisting of golfer i (from the A group) and golfer j (from the B group) and golfer k (from the C group) will start at place c.

In Example 6.5 below we give the first three weeks of the tournament. We can observe that in week 3, player 2 from A and player 4 from B and player 3 from C form a threesome what starts at place 1 (probably hole #1).

Example 6.5 *The first 3 weeks of the tournament for 39 golfers (we have replaced 10 by a, 11 by b and 12 by c and supressed all commas) .*

$S =$

000	111	222	333	444	555	666	777	888	999	aaa	bbb	ccc
123	234	345	456	567	678	789	89a	9ab	abc	bc0	c01	012
246	357	468	579	68a	79b	8ac	9b0	ac1	b02	c13	024	135

We now check the conditions. Obviously there are 39 players playing in threesomes. There are also 13 different starting places and the league play is designed for 10 weeks (with three extra weeks possible). From the fact that A, B and C are latin squares we have that each player plays once in each week and that no golfer starts in the same place twice. From the fact that these latin squares are orthogonal we have that no two golfers are together in a threesome more than once. Thus we have our schedule.

7 Further information

Further information on our scheduling of the *XFL* can be found at [7]. A streaming video of parts of this talk as well as many of the slides from this talk can be found at www.msri.org/publications/ln/msri/2000/combdes/dinitz/1/. A discussion of tournament scheduling that pays particular attention to the home-away patterns can be found at [1]. An extensive survey of Room squares and related designs (including balanced tournament designs) is [9]. Finally, as noted above there is extensive literature on Latin squares and the interested reader is referred to [5] and to [3] (Part 2).

Large schedules such as Major League Baseball require integer programming techniques from operations research. These methods are beyond the scope of this talk, however the interested reader is referred to the web page of Michael Trick (http://mat.tepper.cmu.edu/sports/) for further information on this type of scheduling.

References

[1] E. Burke, D. de Werra, J. Kingston, Applications to timetabling, in *Handbook of Graph Theory* (J. Gross and J. Yellen, eds.) CRC Press, 2004, pp.445 - 474.

[2] P. Cohen, What good is math? An answer for jocks. *New York Times* (Feb. 3, 2001) (National Ed.) pp. A15, A17.

[3] C. J. Colbourn and J. H. Dinitz, eds. *The CRC Handbook of Combinatorial Designs*, CRC Press, Inc., 1996.

[4] J. Corriveau, Enumeration of balanced tournament designs. *Ars Combin* **25** (1988), 93–105.

[5] J. Dénes and A. D. Keedwell, Latin squares and their applications, Academic Press, New York-London, 1974, 547 pp.

[6] J. H. Dinitz and M. H. Dinitz, Enumeration of balanced tournament designs 10 points, *J. Combin. Math. Combin. Comput.*, to appear.

[7] J. H. Dinitz and D. Froncek, Scheduling the XFL, *Congr. Numer.* **147** (2000), pp. 5 – 15.

[8] J. H. Dinitz, D. K. Garnick and B. D. McKay, There are $526,915,620$ nonisomorphic one-factorizations of K_{12}. J. Combin. Des. **2** (1994), 273–285.

[9] J. H. Dinitz and D. R. Stinson. Room squares and related designs. In *Contemporary Design Theory: A Collection of Surveys*, John Wiley & Sons, Inc., 1992, pp. 137–204.

[10] J. H. Dinitz and D. R. Stinson, On assigning referees to tournament schedules, *Bull. Inst. Combin. and its Appl.*, to appear.

[11] K. B. Gross, R. C. Mullin and W. D. Wallis, The number of pairwise orthogonal symmetric Latin squares, *Util. Math.* **4** (1973), 239–251.

[12] J. Haselgrove and J. Leach, A tournament design problem *Amer. Math Monthly*, March 1977, 198 – 201

[13] E. R. Lamken, S. A. Vanstone, The existence of partitioned balanced tournament designs. in *Combinatorial design theory*, North-Holland Math. Stud., 149, North-Holland, Amsterdam, 1987. 339–352,

[14] E. R. Lamken. A few more partitioned balanced tournament designs. *Ars Combinatoria* **43** (1996), 121–134.

[15] E. R. Lamken and S. A. Vanstone. Balanced tournament designs and related topics. *Discrete Math.* **77** (1989), 159–176.

[16] E. Mendelsohn and P. Rodney, Mendelsohn, The existence of court balanced tournament designs, *Discrete Math.* **133** (1994), 207–216.

[17] R. C. Mullin and W. D. Wallis. The existence of Room squares. *Aequationes Math.* **13** (1975), 1–7.

[18] P. J. Schellenberg, G. H. J. van Rees and S. A. Vanstone. The existence of balanced tournament designs. *Ars Combin.* **3** (1977), 303–318.

[19] D. R. Stinson. Room squares with maximum empty subarrays. *Ars Combin.* **20** (1985), 159–166.