数理解析研究所講究録
607 巻 1987 年 33-38

Directing and Orienting Triple Systems

Charles J. Colbourn

Department of Computer Science
University of Waterloo
Waterloo, Ontario, N2L 3G1
CANADA

ABSTRACT

We outline current knowledge about producing directed analogues of the usual undirected triple systems. Algorithms for directing and orienting block designs are emphasized. We also mention some open problems involving orientation of designs.

1. Background

A *triple system* B[3,λ;v] is a pair (V,B); V is a v-set of *elements,* and B is a collection of 3-element subsets of V called *blocks* or *triples;* each 2-subset of elements appears in precisely λ triples. Triple systems have been very widely studied in combinatorial design theory. Mendelsohn [11,13] suggested the extension to designs which are balanced for directed pairs, instead of the usual balance required for undirected pairs of elements. With this in mind, define a *Mendelsohn triple system (MTS)* MB[3,λ;v] to be a collection B of "cyclic triples" on a v-set V of elements; a cyclic triple (a,b,c) is said to contain the ordered pairs (a,b),(b,c),(c,a), and each ordered pair is contained in precisely λ cyclic triples. Such systems are also called "cyclic triple systems", but we adopt the current notation, suggested by Mathon and Rosa [12], to avoid confusion with systems having a cyclic automorphism. Observe that graph-theoretically, blocks in a MTS are cyclic tournaments of order 3; a different directed analogue, suggested by Hung and Mendelsohn [11], takes blocks to be transitive tournaments of order 3. A *directed triple system (DTS)* DB[3,λ;v] is a collection of "transitive triples" on a v-set V of elements; a transitive triple $<a,b,c>$ is said to contain the ordered pairs (a,b),(a,c),(b,c), and every ordered pair is contained in precisely λ

transitive triples. These are also called "transitive triple systems", but again we prefer the present name to avoid confusion with triple systems having transitive automorphism group.

Existence questions for Mendelsohn and directed triple systems have been settled [11,13], and hence one might consider extending the large body of current knowledge about triple systems to their ordered analogues. However, it seems more sensible to explore the relationship between triple systems and the ordered versions, in the hope that a number of extensions may be straightforward. With this in mind, observe that if one simply pretends that cyclic triples or transitive triples are just 3-subsets, one produces a $B[3,2\lambda;v]$ from a $MB[3,\lambda;v]$ or a $DB[3,\lambda;v]$; the triple system so obtained is called the *underlying* triple system. Naturally, for some systems this process can be reversed. When a triple system $B[3,2\lambda;v]$ can be rewritten as cyclic triples to produce a $MB[3,\lambda;v]$, we say the system is *orientable,* and we refer to the process of producing the required cyclic triples as *orienting.* Similarly, when a triple system can be written as transitive triples to produce a $DB[3,\lambda;v]$, we say the system is *directable,* and we refer to the production of transitive triples as *directing.*

Mendelsohn [13] showed that there exist $B[3,2;v]$'s which are not orientable; in fact, for every $v\equiv 0,1 \pmod 3$, $v\geq 9$, there is an orientable $B[3,2;v]$ and a non-orientable $B[3,2;v]$ [1]. Turning to directing, however, the situation is dramatically different: every $B[3,2;v]$ is directable [3]. In the remainder of this paper, we study algorithms for directing and orienting, and at the same time consider the extension to higher λ.

2. Directing triple systems

Colbourn and Colbourn [3] established that every triple system with $\lambda=2$ can be directed, by describing an efficient algorithm for directing such a triple system. This algorithm was later modified by Colbourn and Harms [5] to show that every $B[3,2\lambda;v]$ is directable. The idea in the algorithm is straightforward. First, modify each 3-subset to form a transitive triple in any way at all. In the configuration which results, each ordered pair appears between 0 and 2λ times. An ordered pair appearing s times for $s\neq\lambda$ is termed a *conflict,* and the *severity* of the conflict is $max(s-\lambda,\lambda-s)$. The directability of the system is established by showing that the severity of some conflict can be reduced, while no conflicts are introduced and none become more severe. Repetition of this process eventually eliminates all conflicts. As a simple example, suppose $<a,b,c>$ is a transitive triple and (a,b) appears more than λ times; replacing $<a,b,c>$ with $<b,a,c>$ reduces the severity of the conflict. Not all substitutions are this trivial, but Colbourn and Harms show that in every case there is a sequence of substitutions which reduce the severity.

Computational experience with this algorithm showed that, despite its polynomial running time, it does a lot of unnecessary work -- triples are reordered at one step only to be returned to their original order at some later step. It is therefore of interest to direct the system one block at a time, never backtracking to correct an earlier incorrect decision. Harms and Colbourn [8,9] developed an algorithm with this behaviour. We sketch this simpler method here.

Given a $B[3,2\lambda;v]$, first fix an ordering for the elements; for convenience, let the elements be $\{0,1,...,v-1\}$ and use the natural ordering. Partition the blocks into segments; the segment $S(i)$ for element i contains all blocks in which i is the smallest element. The algorithm operates by processing the segments $S(v-1)$ down to $S(0)$, forming transitive triples for each segment in turn. The key observation is that in processing $S(i)$, no pairs involving i are directed yet, and moreover that the pairs appearing with i in blocks of $S(i)$ induce a multigraph of maximum degree 2λ. Directions for the edges in this multigraph can be chosen arbitrarily to maintain "balance" on edges $\{j,k\}$ with $j,k>i$, and it is then relatively straightforward to handle all pairs containing i. The details appear in [8], and a worked example in [9].

Directing a triple system by either algorithm gives many selections for edge directions which can be made arbitrarily; hence, it appears plausible that a triple system underlies very many different directed triple systems. One avenue that remains unexplored here is to use probabilistic techniques to show that every triple system is directable -- this may be generalizable to producing many distinct directed triple systems on the same underlying system. Little progress in this direction has been made, although there is one very appealing related conjecture: every triple system underlies six disjoint directed triple systems [10]. If true, this is of course best possible, since any triple can only be directed as a transitive triple in six different ways. This conjecture seems plausible, but is far from settled.

Finally, it deserves mention that every triple system having a cyclic automorphism underlies a directed triple system having a cyclic automorphism [7]. It seems reasonable to expect that one can require other group properties to carry over in the directing process.

3. Orienting triple systems

Orienting triple systems is a completely different matter. In this case, there are systems which cannot be oriented. Take, for example, the unique B[3,2;6] with blocks {012, 013, 024, 035, 045, 125, 134, 145, 234, 235}. Consider any block, say {012}. In an orientation, this block must appear as (0,1,2) or (0,2,1); there are only two choices. Moreover, if there is an orientation containing, say, (0,1,2), there is a second orientation obtained by changing each cyclic triple (a,b,c) to (a,c,b); hence, there is an orientation containing (0,2,1). Thus we assume without loss of generality that {012} is oriented as (0,1,2). Given this, {013} must be oriented as (0,3,1); similarly, {024} as (0,2,4) and {125} as (1,5,2). These in turn imply the orientation of more triples: (0,5,3), (1,3,4), (0,4,5), and (2,3,4). But now we require that {235} be oriented both as (2,3,5) and as (2,5,3), an obvious impossibility. At no step was a choice made, and hence the original design cannot be oriented. This process applies in general, and is an efficient method for determining whether a B[3,2;v] is orientable; in the general case, three outcomes of the forcing procedure outlined above are possible: all triples are oriented (design is orientable), a contradiction as above is encountered (design is not orientable), or some triples are oriented without contradiction but there remain blocks whose orientation is not forced. It is easy to see in the latter case that we can omit the blocks already oriented; the orientability of the whole design depends entirely on the orientability of those which remain. This is an interesting method because it involves no backtracking, despite the initial appearance that backtracking may be required. In this regard, it is very similar to determining satisfiability of logical formulas with two literals per clause; M. Colbourn [6] has observed that orientability can be checked easily using an algorithm for 2-satisfiability.

One might hope again that the results for $\lambda=2$ generalize to higher λ; here, however, Colbourn [2] has proved that determining orientability is NP-complete, even for $\lambda=4$. This has two important consequences: it shows that orienting for higher λ differs substantially both from directing and from orienting with $\lambda=2$. The second consequence is that is establishes the existence of infinitely many triple systems with $\lambda=4$ which are non-orientable. The spectrum of non-orientable triple systems with $\lambda=4$ remains undetermined, however.

The NP-completeness result really limits what one might hope to do for higher λ; nevertheless, it appears possible that some reasonable necessary conditions or sufficient conditions might be developed for orientability.

4. Open Problems

The main point of this brief introduction is to outline the state of current knowledge. Perhaps the most important aspect is to state carefully some interesting open problems, some of which are currently being studied by the author and others.

1. Prove by probabilistic techniques that every triple system can be directed.
2. Extend #1 to prove that every triple system underlies a "large number" (exponentially many?) directed triple systems.
3. Prove that every triple system underlies six block-disjoint directed triple systems (see [10]).
4. Prove that every B[4,2;v] underlies a directed block design in which blocks are transitive tournaments of order four, *or* find a counterexample (see [4]).
5. Extend #4 to block size $k>4$ using transitive tournaments of order k (this seems very ambitious).
6. Extend #4 to higher even λ.
7. Show that orienting for every fixed even $\lambda>4$ is NP-complete.
8. Determine the spectrum of non-orientable B[3,λ;v] for all even $\lambda\geq 4$.
9. Develop reasonable necessary conditions for orientability for $\lambda\geq 4$.

Most of these problems will be difficult, but all are nice extensions to the current state of affairs.

Acknowledgements

Thanks to Rudi Mathon and Alex Rosa for helpful discussions, to Ryoh Fuji-Hara and Masakazu Jimbo for their great hospitality in Japan, and to NSERC Canada for financial support.

References

[1] F.E. Bennett and N.S. Mendelsohn, "Some remarks on 2-designs $S_2(2,3,v)$", *Proc. Ninth Southeastern Conference on Combinatorics, Graph Theory, and Computing* (1978) 119-127.

[2] C.J. Colbourn, "Orienting triple systems is NP-complete", *Ars Combinatoria,* to appear, 1986.

[3] C.J. Colbourn and M.J. Colbourn, "Every twofold triple system can be directed", *J. Comb. Theory* A34 (1983) 375-378.

[4] C.J. Colbourn and M.J. Colbourn, "Directing graph designs", *Proceedings of the Fourteenth Southeastern Conference on Combinatorics, Graph Theory, and Computing* (1983) 225-230.

[5] C.J. Colbourn and J.J. Harms, "Directing triple systems", *Ars Combinatoria* 15 (1983) 261-266.

[6] M.J. Colbourn, "Algorithmic aspects of combinatorial designs: a survey", *Annals of Discrete Mathematics* 26 (1985) 67-136.

[7] J.J. Harms, "Directing cyclic triple systems", *Annals of Discrete Mathematics* 26 (1985) 221-226.

[8] J.J. Harms and C.J. Colbourn, "An optimal algorithm for directing triple systems using Eulerian circuits", *Annals of Discrete Mathematics* 27 (1985) 433-438.

[9] J.J. Harms and C.J. Colbourn, "An efficient algorithm for directing triple systems", *Proceedings of the Fourteenth Southeastern Conference on Combinatorics, Graph Theory, and Computing* (1983) 455-462.

[10] J.J. Harms and C.J. Colbourn, "Partitions into directed triple systems", *Ars Combinatoria* 16 (1983) 21-25.

[11] S.H.Y. Hung and N.S. Mendelsohn, "Directed triple systems", *J. Comb. Theory* A14 (1973) 310-318.

[12] R.A. Mathon and A. Rosa, "A census of Mendelsohn triple systems of order nine", *Ars Combinatoria* 4 (1977) 309-315.

[13] N.S. Mendelsohn, "A natural generalization of Steiner triple systems", in: *Computers in Number Theory* (A. Atkin and B. Birch, eds.) Academic Press, 1971, pp. 323-338.