Research Statement

I work primarily in Combinatorics and Combinatorial Optimization, but am also active in Mathematical Biology. I like diversity in my research and am always looking for ways to expand the range of my interests and to work with other scientists or industrial partners. I am predominantly attracted by combinatorial problems that have mathematical depth as well as potential applications, and where implementation issues raise exciting mathematical questions. I have found that diverse problems from science and technology require broad knowledge and I have always been excited to learn new techniques. My main combinatorial work is specifically in graph, design and coding theory and Gray code realizations. I am interested in industrial optimization, scheduling and software testing problems. My research in biology involves demography and the evolution of amphimixis. I shall now describe each of these four research areas.

1 Graphs, Designs, and Codes

My thesis work focused on two problems: covering arrays and packing arrays. These are relaxations of orthogonal arrays to covering and packing hypotheses. My thesis work was spent generalizing Wilson's construction, developing new constructions and establishing upper and lower bounds. My continuing work in covering arrays is related more to combinatorial optimization and industrial mathematics and so will be discussed in Section 3.

Packing arrays can be viewed as error correcting codes with $d = n - 1$. These also have industrial applications in multi-disk allocation as discussed by Abadi and Abdel-Ghaffar. My research extends the result of Abdel-Ghaffar, showing that equality holds in the Plotkin bound for holey codes when $n \geq 2g - 1$. As part of this proof, I completely solved the edge colourability of complete graphs with one complete hole. Multiple holes are a natural extension of this research. For particular values in the range $n < 2g - 1$ equality in this bound would have strong implications about the existence of finite projective planes and I am currently investigating this deep connection. When $n < 2g - 1$ and there are no holes in the code, then the dual structure of a packing array is a Class-Uniformly Resolvable Design. These designs were introduced by Lamken, Rees and Vanstone. Peter Danziger and I have written one paper investigating basic constructions and necessary conditions, deriving extremal conditions (rare in design theory) and providing several complete infinite families, and another investigating Class-Uniformly Resolvable Group Divisible Designs. We are continuing this research by applying frame methods and constructions from finite geometries. Class-Uniformly Resolvable structures have applications in the statistical design of experiments, tournament design and scheduling, construction of high order difference arrays, and error correcting codes.

Lastly, I have solved the pancyclic 2-factorization problem for undirected complete graphs, undirected complete bipartite graphs and directed complete graphs of odd order. Directed graphs permit 2-cycles, which increases the difficulty of the problem substantially. The techniques used are closely related to Piotrowski's approach to the Oberwolfach problem. Pavol Gvozdjak and I are working to combine my pancyclic techniques with his Oberwolfach solutions to produce arbitrary 2-factorizations with one or two cycles per factor. Dalibor Froncek and I are investigating another problem complementary to the Oberwolfach problem: 2-factorizations of complete graphs with each 2-factor having a different number of connected components. We have solved this problem for undirected complete graphs and are turning our attention to other graphs.

2 Gray Codes and Superstring Problems

Gray Codes are very useful for digital to analogue conversion and many other applications. Part of their usefulness derives from the fact that they prescribe efficient generation procedures for classes of combinatorial objects and therefore function as a kind of compression. This compression utility would be further enhanced if the Gray code realization of a set of objects were compatible with standard data structures. It is easy to see that the standard binary reflected Gray code can be realized in two stacks. The question of which Gray Codes can be realized as successive states of a queue is important but little is known. I have proved that 1,2,5 and 6 bit cyclic Gray codes can be realized in queues and that 3 and 4 bit cyclic codes cannot. I am looking for a recursive construction.

The related question of which Gray code realizations of k-sets of an n-set can be realized in a queue is an outstanding conjecture of Chung, Diaconis and Graham. I recently showed that $k = n - 2$ is always impossible even when the necessary divisibility conditions are satisfied. This was shown by establishing strong internal structural lemmas. I have now turned my attention to establishing similar structural results for the case $k = n - 3$. Randomized searches will be important in both of the previous problems.

These two problems are loosely related to two other superstring problems. The first question is what is the shortest cyclic string on an n letter alphabet that contains every combination of letters as a contiguous subword. And affirmative answer to the existence of queue realizable n-bit Gray codes would give an upper bound of 2^{n-1} for this question. Our best result to date is $n2^{n-1}$. The second similar problem is to find the shortest superword that contains a given set of words as subsequences. This arises out of my industrial work with MITACS.

3 Combinatorial Optimization

My work with IBM and MITACS involves combinatorial optimization and scheduling. We have completed some projects and have several active projects underway. Our most significant completed project was an industrial scheduling problem for utility technicians. We needed to efficiently solve the multi-van traveling salesman problem with time windows, but our client's major constraint was running time. We adapted very fast matching algorithms and sparse data structures to produce suboptimal but very good solutions within their time limitations. The code that we produced is being integrated into their software package and will be used operationally by the company.

Another active project, finding the shortest superword containing a fixed set of words as subsequences over an α-alphabet, is the formalization of a job scheduling problem with degrees of parallelism and precedence constraints. We have shown the problem to be NP-hard and have been able to prove that the greedy algorithm is exactly α bad. We are currently looking for more complicated heuristics that will produce closer to optimal solutions. The second project is nurse scheduling. The issue here is to achieve some kind of long term average paraeto optimality or envy-free condition.

A third project is to optimize the manufacture of high-performance multi-layer ceramic packages (MLCs) for electronic chips. MLCs are enjoying increasing popularity, production demand is rising sharply, and so optimizing production is commercially important. In each layer approximately 50,000 "via" holes must be punched. The specialized punch tool bears an array of solenoid-driven pins, and the tool is repeatedly moved to punch all the required holes. Efficient operation entails minimizing the number of tool positions (a "minimum set cover" problem) and minimizing the travel time between them (a "Traveling Salesman Problem"). My work focused on geometric computations and precision of the set cover problem. It is expected that the software I contributed will enter production use soon.

In addition to scheduling problems, the covering arrays from my thesis are directly applicable to software testing problems and derandomization. I propose to program an online covering array generator that could provide the best known arrays to software developers and be kept up to date with the latest developments in the theory. I am also investigating the use of graph products and homomorphisms to optimize covering arrays when there are known non-interactions between variables or nodes on a network.

4 Mathematical Biology

The utility of amphimixis over parthenogenesis as a breeding strategy is both controversial and difficult. Amphimixis must confer at least a two times advantage to overcome the double breeding capacity of parthenogenetically reproducing species. I have extended Kondrashov's deterministic work to include environmental variation and determined that varying environments extend the utility of amphimixis. Towards a stochastic solution I have proven an upper bound on the click rate of Muller's Ratchet by bounding the expected stopping time of the stochastic process that describes mutation–free parthenogens. This work continues with the aim of generating analytical results on the stochastic model. Simulation results have also been produced, suggesting a possible interpretation of the benefits of cyclical parthenogenesis.

I am also interested in demography, specifically in a graph equivalence question derived from the problem of finding regular inbreeding patterns for livestock herds that avoid rapid approaches to genetic homogeneity. Inbreeding patterns impose a condition looser than pure isomorphism on families of n-partite graphs. This question has only been solved for small breeding stocks.