NSERC Discovery Grant Proposal

This is the NSERC Discovery Grant Proposal submitted by Curtis Bright in 2020. It is being released publicly in the hope that it can help other researchers who are preparing grant proposals.

Notice of Intent

Fast search algorithms are at the heart of effective solutions to a huge number of industrial and theoretical problems—from search engines to resource allocation to mathematical conjecture verification. Some of the most effective general-purpose search techniques come from the field of satisfiability (SAT) checking. Indeed, programs called SAT solvers that search for solutions to logic problems are extremely effective at solving many kinds of search and optimization problems that arise in practice—though they tend to struggle with mathematically complex problems. In these kinds of problems it is typical to either use a computer algebra system (CAS) or a "SAT modulo theories" (SMT) solver which offer support for more complex mathematics. However, there are many problems that require *both* sophisticated mathematics (beyond what SAT or SMT solvers offer) and fast search routines (beyond what CASs offer).

This project aims to advance a new mathematical search paradigm that exploits both the search power of SAT solvers and the mathematical capabilities of CASs—thereby achieving the best in both the worlds of satisfiability checking and computer algebra. This "SAT+CAS" method is still in its infancy but has shown great promise in a number of preliminary results over the last few years. For example, the prototype SAT+CAS tool MathCheck has resolved several problems from combinatorics, number theory, and geometry that were not solvable using either SAT solvers or CASs alone. Thus, I am applying for an NSERC Discovery Grant in order to fully realize the potential of the SAT+CAS method. The funds will support extending MathCheck to new domains as well as applying the upgraded MathCheck to a wider variety of problems—such as searching for pooling schemes to more efficiently test a population for a virus. The ultimate goal of this research is to make the SAT+CAS method one of the most effective methods—if not the most effective method—for tackling general mathematical search problems.

To date, the SAT+CAS method has verified or improved the best bounds in several mathematical conjectures and in particular has resulted in significant increases to our understanding of Williamson matrices. In the 1960s, these matrices were used for developing codes to communicate with spacecraft and it was conjectured that Williamson matrices exist in all orders. This was disproven in 1993, but the smallest counterexample was unknown until it was determined by the SAT+CAS method as a part of my research in 2016. Moreover, we were able to search for Williamson matrices in even orders far higher than previously possible. Previous exhaustive searches had only completed up to order 18, while our SAT+CAS approach completed exhaustive searches up to order 70. This uncovered more than 100,000 new sets of Williamson matrices—whereas fewer than 200

sets of Williamson matrices were previously known.

We have also used the SAT+CAS method to show the nonexistence of combinatorial objects by generating nonexistence certificates that can be verified by a third party. For example, we recently solved Lam's problem from discrete geometry—the problem of proving the nonexistence of a finite projective plane of order ten. Projective planes have been widely studied since the 1600s and the first order for which it is theoretically uncertain if finite projective planes exist is ten. A massive computer search culminating in the 1980s determined that such planes do not exist. The SAT+CAS method solved Lam's problem significantly faster than previous searches (one particularly challenging subcase was solved over 500 times faster when compared with a recent 2011 verification) and produced the first nonexistence certificates for Lam's problem.

These initial successes speak to the potential of the SAT+CAS method for solving hard mathematical search problems that cannot feasibly be solved using existing tools. This research grant will support extending the SAT+CAS method to new kinds of problems, including those from new kinds of domains. As one example, consider the problem of testing samples for the presence of a virus. The most basic method is simply to test each sample individually, but this is costly. A more efficient strategy is to employ "pooled testing" where multiple samples are tested together. By judiciously choosing which samples to combine together one can develop testing schemes which are essentially as quick and accurate as the basic method but require significantly fewer tests. For example, disjunct matrices are combinatorial matrices that give rise to good pooling schemes. Disjunct matrices are known to exist once the size of the matrices are large enough, but these can be too large for many applications. However, by extending MathCheck into the domain of combinatorial group testing we can use the SAT+CAS method to search for new disjunct matrices—and therefore new pooling schemes. These could be useful to facilitate continuous testing for viruses like COVID-19, especially in regions where the medical system is being pushed to its limits.

This is just one of many new domains where progress is possible—other domains include physics (e.g., searching for Kochen–Specker systems), cryptography (e.g., searching for almost perfect nonlinear permutations), circuit complexity (e.g., searching for minimal circuits that compute certain functions), computational geometry (e.g., searching for optimal tripod packings), as well as additional problems in combinatorics (e.g., searching for mutually orthogonal Latin squares). The SAT+CAS method is perfectly positioned to attack such problems and the time is ripe to exploit the advances made by SAT solvers and CASs in order to solve problems previously considered infeasible.

Suggested Applicant Category: Early Career Researcher (ECR)

I have only held independent academic positions since July 2020 and am therefore an early career researcher. I joined the School of Computer Science at the University of Windsor as an adjunct professor in July 2020 and officially start as a tenure-track assistant professor in January 2021—with the authority to supervise undergraduate students, graduate students, and postdoctoral fellows.

Summary of Proposal

Fast search algorithms are at the heart of effective solutions to a huge number of industrial and theoretical problems—from search engines to resource allocation to mathematical conjecture verification. Some of the most effective general-purpose search techniques come from the field of satisfiability (SAT) checking. Indeed, programs called SAT solvers that search for solutions to logic problems are extremely effective at solving many kinds of search and optimization problems that arise in practice—though they tend to struggle with mathematically complex problems. In these kinds of problems it is typical to either use a computer algebra system (CAS) or a "SAT modulo theories" (SMT) solver which offer support for more complex mathematics. However, there are many problems that require *both* sophisticated mathematics (beyond what SAT or SMT solvers offer) and fast search routines (beyond what CASs offer).

This proposal aims to advance a new mathematical search paradigm that exploits both the search power of SAT solvers and the mathematical capabilities of CASs—thereby achieving the best in both the worlds of satisfiability checking and computer algebra. This "SAT+CAS" method is still in its infancy but has shown great promise in a number of preliminary results over the last few years. This research grant will support extending the SAT+CAS method to a wider variety of problems, including those from new kinds of domains. The ultimate goal of this research is to make the SAT+CAS method one of the most effective methods—if not the most effective method—for tackling general mathematical search problems.

As one example, consider the problem of testing samples for the presence of a virus. The most basic method is simply to test each sample individually, but this is costly. A more efficient strategy is to employ "pooled testing" where multiple samples are tested together. By judiciously choosing which samples to combine together one can develop testing schemes which are essentially as quick and accurate as the basic method but require significantly fewer tests. For example, disjunct matrices are combinatorial matrices that give rise to good pooling schemes. Optimal disjunct matrices are known to exist once the size of the matrices are large enough, but these can be too large for many applications. However, by extending the SAT+CAS method into the domain of combinatorial group testing we can search for new disjunct matrices—and therefore new pooling schemes. These could be useful to facilitate continuous testing for viruses like COVID-19, especially in regions where the medical system is being pushed to its limits. The SAT+CAS method is perfectly positioned to attack such problems and the time is ripe to exploit the advances made by SAT solvers and CASs in order to solve problems previously considered infeasible.

	Qty	Amount/year	Qty	Amount/year	Qty	Amount/year	Qty	Amount/year	Qty	Amount/year
Salaries and benefits										
Undergraduate	1	$3,000	1	$3,000	1	$3,000	1	$3,000	1	$3,000
Master's	1	$6,000	1	$6,000	1	$6,000	1	$6,000	1	$6,000
Doctoral	2	$24,000	2	$24,000	3	$36,000	3	$36,000	3	$36,000
Subtotal		$33,000		$33,000						
Postdoctoral	1	$40,000	1	$40,000	1	$30,000	1	$30,000	1	$30,000
Subtotal		$40,000		$40,000		$30,000		$30,000		$30,000
Equipment or facility										
Purchase or rental		$7,500		$0		$9,000		$0		$0
User fees		$2,500		$2,500		$2,500		$2,500		$2,500
Subtotal		$10,000		$2,500		$11,500		$2,500		$2,500
Materials and supplies										
		$0		$0		$0		$0		$0
Subtotal		$0		$0		$0		$0		$0
Travel										
Conferences		$12,000		$12,000		$14,000		$14,000		$14,000
Subtotal		$12,000		$12,000		$14,000		$14,000		$14,000
Dissemination Costs										
		$0		$0		$0		$0		$0
Subtotal		$0		$0		$0		$0		$0
Other (specify)										
		$0		$0		$0		$0		$0
Subtotal		$0		$0		$0		$0		$0
TOTAL PROPOSED EXPENDITURES		$90,000		$97,000		$102,500		$93,500		$93,500
Total Cash Contribution from industry (if applicable)		$0		$0		$0		$0		$0
Total Cash Contribution from university (if applicable)		$14,000		$14,000		$14,000		$14,000		$14,000
Total Cash Contribution from other sources (if applicable)		$0		$0		$0		$0		$0
Total amount requested from NSERC		$76,000		$83,000		$88,500		$79,500		$79,500

Relationship to Other Research Support

I have no other sources of research support other than those accounted for in the proposed expenditures. In particular, as a part of my start-up grant the University of Windsor has agreed to support my research program at $14,000/year for the next five years. As outlined in the proposed expenditures this money will support the salary and travel of a single PhD student for five years. From NSERC I am requesting the funds necessary to hire additional students and postdoctoral researchers for five years (as outlined in the budget justification).

HQP Training Plan

Training Philosophy

My philosophy for training highly qualified personnel is based around the fact that individuals are diverse in their interests, strengths, and preferences. Accordingly, I strive to treat each researcher as an individual with unique qualities. In practice, this requires getting to personally know students in order to most effectively tailor the advice and projects provided to them. Fortunately, the flexibility of my research program supports this: as described in the proposal, there are plenty of branches of science and mathematics to which my research can effectively be applied—ensuring that each student can work in a domain of interest to them and thereby produce their best work.

My approach to training students is that I guide and mentor my students, offering projects and directions for their research. I also encourage students to set their own direction, especially as they become more acquainted with research. One skill that I strongly emphasize and look to bring out in to all my students is clarity of writing. This has innumerable benefits, both to disseminating research and producing quality research. Moreover, it is a skill that will serve them well in their future, regardless of their career path.

My philosophy also encourages interdisciplinary research and interactions with collaborators outside of computer science—in my past research I've collaborated with mathematicians, engineers, and research scientists in industry. Most of the projects in my research program involve applications (including industrial applications) to other fields and I encourage my students to take advantage of these synergies to improve their own "box of tools".

Equity, Diversity, and Inclusion

I strongly support increasing equity, diversity and inclusion in the sciences and this informs my training philosophy in order to promote these ideals. Firstly, I am conscious about being inclusive and eliminating stereotyping and biases in my training, research, and in the field at large. For example, I take care to use inclusive and unbiased language in my work and when appropriate have raised the issue with others. For example, there was a passage in Knuth's the Art of Computer Programming that used a masculine pronoun to refer to a generic individual. Knuth invites suggestions to improve his work, and I brought this to his attention. He decided to update the passage.

Additionally, my philosophy for evaluating candidates recognizes that many traditional methods of evaluation may unintentionally disadvantage individuals whose backgrounds are under-represented in the sciences. For example, I strive to only evaluate candidates based on how their interests align with the project and in their ability to see a research project to successful completion—as demonstrated by previous work (if available) or a concrete demonstration of their ability to complete work such as through a writing sample. I believe this provides a more accurate assessment of the likelihood of research success instead

of the crude metrics that are too often used—such as the most number of publications, awards, years of experience, or highest marks. I would rather select an applicant who can provide demonstrations of clear and well-written work (regardless of if the work has been formally published) over a candidate that has impressive credentials on paper. I believe that in addition to selecting the most likely candidates to succeed this also helps to level the playing field with respect to equity, diversity, and inclusion—since many qualified individuals from less advantaged positions may not have had the opportunities that would enable them to achieve the impressive credentials as measured "on paper".

In the same vein, I avoid selecting candidates based on their "fit" with my research group and environment—I recognize that there are good researchers from all backgrounds and those with the best "fit" are more likely from similar backgrounds. Rather than merely selecting candidates who naturally fit in, I instead strive to foster an environment where all individuals—regardless of their background—will feel like they fit in. To this end, I plan to proactively get to know all of the members in my research group along with their aspirations and struggles. I also have an open-door policy—I am always available to lend an ear and (when requested) provide guidance or assistance.

Challenges Related to Equity, Diversity, and Inclusion

Unfortunately, there are a number of challenges related to equity, diversity, and inclusion in computer science and at UWindsor. For example, the gender breakdown of students in the field of computer science skews towards male. The University of Windsor Office of Institutional Analysis reports that in the Winter 2020 term only 15% of the undergraduate students—and 40% of the graduate students—enrolled in a computer science program identified as women. Moreover, several studies of North American institutions have found that these numbers are fairly typical for the field of computer science.

A second challenge related to equity, diversity, and inclusion concerns the demographic breakdown of the University of Windsor and in computer science as a whole. Certain demographic populations—such as Indigenous populations—are not well represented at either UWindsor or in computer science. In the Winter 2020 term, the Office of Institutional Analysis at UWindsor reported that (of the students who chose to self-identify) about 85% of computer science graduate students were international students. Moreover, they have also reported an uneven distribution in the represented nationalities—with just three countries accounting for 87% of the international student body.

These numbers speak to the challenge of recruiting women and people from under-represented demographics but also to the great opportunity that we have. Changing these numbers is not something that can not change overnight, but there are encouraging signs. For example, the fact that UWindsor's graduate program has a much higher proportion of women than the undergraduate program makes it clear that there are a significant number of women interested pursuing research in this field. One potential way to help improve the demographic parity is to support early outreach efforts. For example, I would like to get even high school students interested in the field for example by giving presentations that

are accessible to those with no background in the field—avoiding the minutiae overload that is all-to-common among research presentations.

Research Training Plan

My research training plan is tailored to produce significant research as well as to impart research skills to my trainees at each stage of their careers. As detailed in the proposal, my research plan supports numerous projects that are appropriate to individuals of varying skills and for varying amounts of commitment.

Undergraduate projects are chosen from an area (recreational mathematics and discrete tomography) that is accessible and has many projects that are feasible in a short amount of time. Master's projects are chosen from an area (circuit minimization) that is less accessible but does not require advanced mathematical knowledge. PhD projects are chosen from areas that require deeper analytic or algebraic knowledge (such as physics and combinatorial matrix theory)—but once that knowledge is acquired can be applied to other similar projects, resulting in projects leading to multiple papers. Postdoctoral projects require the most amount of mathematical background and programming ability.

The projects are also chosen to impart valuable skills to the highly qualified personnel. For example, how to organize, execute, and optimize combinatorial searches—skills which generalize to a great many other kinds of problems. Additionally, the projects have also been chosen to have direct applications and value beyond mathematics. For example, the combinatorial matrices project has applications to pooled testing and therefore increasing the efficiency of testing for viruses like COVID-19.

In addition, the training plan has been chosen to be feasible for myself to have an active role in each project. Some students may be co-supervised, but I plan to have two PhD students and a Master's student for the first two years and I will start supervising a third PhD student after two years. Additionally, I plan to work with undergraduates on summer projects as well as supporting one postdoctoral researcher. The postdoctoral researcher will also assist with student supervision. Not only do the students receive the benefit of having an additional source of guidance, this is also of significant benefit to the postdoctoral researcher—giving them valuable supervisory experience that will be useful to them in their future career.

Past Contributions to HQP Training

Because I am an early career researcher (starting as a tenure-track professor in January 2021) my past contributions to HQP training have been in a mentorship or unofficial advisory role.

Firstly, as a PhD student I had the opportunity to co-supervise the student Abhinav Baid in a summer project funded by Google as a part of their Summer of Code program. Abhinav was a Bachelor of Engineering student from the Birla Institute of Technology and Science (India) and was co-supervised by myself and professor William Hart from Warwick University (United Kingdom). The project was to implement and optimize a lattice basis reduction algorithm as a part of the FLINT (Fast Library for Number Theory) research project.

The project was successful, and produced an optimized lattice basis reduction implementation that is available in FLINT from version 2.5 onwards—this also includes a novel variant proposed in my Master's work. As lattice basis reduction has an amazing wealth of mathematical applications it is of significant benefit to researchers worldwide to have a free, open-source, and publicly-available optimized lattice basis reduction algorithm available. In particular, the novel variant in FLINT allows solving linear systems (one of the most fundamental computations pervasive in science and engineering) more efficiently.

I provided most of Abhinav's supervision by answering questions about the algorithm and lattice theory, providing direction of what tasks to prioritize, and giving advice about implementation issues. I also made a point to encourage precise documentation and clear writing. William Hart, as the director of the FLINT project provided advice about how the implementation should fit into the rest of the FLINT library.

Abhinav said that the experience was "a tremendous learning experience" and has since thanked me for imparting coding skills—such as how to document code—that have been useful in his career. The next year he successfully participated in the Summer of Code program again and implemented an algorithm for factoring linear ordinary differential operators. In the next two years he was a software engineering intern with Google and a research intern with the Indian Institute of Science. He then graduated from his program and took his current position as a software engineer with Bloomberg LP in London (United Kingdom).

Secondly, as a postdoctoral fellow I mentored several graduate students at the University of Waterloo, including Saeed Nejati, Chunxiao Li, and Sebastian Verschoor. I helped them edit drafts of their papers, provided feedback on their work, and helped them use satisfiability (SAT) solvers in their research. I also prepared benchmarks that could be used in order to test a SAT solver's performance—resulting in a report in the 2020 Proceedings of the SAT Competition (co-written by Saeed Nejati, Vijay Ganesh, and myself). Additionally, Chunxiao published a paper in the 2020 SAT conference, Saeed published three SAT-related papers since 2018, and Sebastian has written two papers in the area of SAT solving—one of which was published in Nature's *Scientific Reports* in 2020.

Thirdly, as a postdoctoral fellow at Carleton University I have been unofficially co-

supervising the undergraduate student Noah Rubin along with professors Kevin Cheung and Brett Stevens. The project is still in the early stages but he has already implemented a hybrid integer and constraint programming technique for searching for mathematical objects. I have been providing direction to Noah as the project unfolds in addition to advice about how to successfully devise and structure efficient mathematical searches. More research is necessary to see if this approach can scale and beat more conventional approaches, but has already produced promising results. Moreover, the background in integer and constraint programming that Noah has now developed should serve him well in his future endeavours.

Lastly, I recently started supervising the Waterloo undergraduate student Madhur Sharma. Madhur is interested in extending my PhD work in order to search for other kinds of combinatorial sequences. This supervision is currently informal though there is the potential of him officially becoming a graduate student of mine at a later date.

[All students mentioned here have given their consent to be named.]

Most Significant Contributions

Over the past six years I have made a number of important practical and theoretical contributions in my research. Most notably, I helped spearhead a new paradigm for solving mathematical search problems through combining two previously separate areas of computer science—satisfiability checking and computer algebra. My work has clearly demonstrated the power and flexibility of this paradigm by effectively employing it to solve a number of problems in combinatorics [C8–9], geometry [C1–3], graph theory [C4], and number theory [C7]. In each case, my research has shown that problems can be solved more quickly, rigorously, and verifiably when compared to previous approaches. The paradigm has already received significant interest from both academia and industry: the "SC-square" (satisfiability checking and symbolic computation) project was started in 2016 order to advance the paradigm and now has associates from over 40 universities and 15 companies.

As the lead developer of MathCheck—the first system to combine satisfiability (SAT) solvers and computer algebra systems (CAS)—I have been at the forefront of this emerging area. My work on MathCheck has resulted in three invited talks at conferences and workshops: at Applications of Computer Algebra in 2018, at the SC-square workshop in 2019, and at the upcoming Canadian Mathematics Society meeting in 2020. I also presented MathCheck in the "sister conference best paper track" of the International Joint Conference on Artificial Intelligence in 2016. My work on MathCheck also appears in two invited papers: in the SC-square track of Computer Algebra in Scientific Computing in 2016 [C9] and in the Journal of Automated Reasoning in 2017 [J8].

1. Williamson Matrices

First defined in 1944, Williamson matrices have been extensively studied for both their theoretical and practical properties and my work has led to a number of new discoveries concerning them. In the 1960s, researchers at NASA studied Williamson matrices in the process of developing codes for communicating with spacecraft and they conjectured that Williamson matrices exist in all orders. This was disproven in 1993, but the smallest counterexample was unknown until I determined it through extensive searches performed by MathCheck in 2016 [C9]. Fewer than 200 Williamson matrices had previously been discovered but MathCheck uncovered over 100,000 new Williamson matrices [C8,J3]. We were able to run searches in much higher orders than had ever previously been accomplished: for example, the even orders had only previously been exhaustively searched up to order 18 while we exhaustively searched the orders up to 70. Moreover, these searches revealed unexpected patterns that led to the discovery of new infinite classes of Williamson matrices that had eluded researchers for over 75 years. For example, in Williamson's original paper on the subject he found Williamson matrices exist in orders that are powers of two up to 32. Nothing more was known until MathCheck discovered examples in order 64 and I generalized the patterns appearing in these matrices to prove that Williamson matrices exist for all orders that are powers of two [J4].

2. Projective Geometry

Projective geometry was developed in the 1600s in order to formalize the techniques of drawing a three dimensional scene onto a two dimensional canvas. My work on projective geometry has produced the fastest and most rigorous solution of Lam's problem, one of the most celebrated and long-standing problems in the field [C1,J2]. All projective geometries that have a finite number of points are completely classified—with the exception of projective planes, i.e., those having exactly two dimensions. The smallest projective plane whose existence is theoretically uncertain is known as a "projective plane of order ten" and Lam's problem is to determine if such a plane exists or not. Open since the 1800s, Lam's problem was resolved in the 1980s by a massive search requiring months on the fastest supercomputers of the era.

All previous approaches to solving Lam's problem relied on special-purpose ad-hoc computer code that is difficult to write and almost impossible to verify. In contrast, my work on Lam's problem provides certificates of nonexistence that a third party can use to convince themselves of the result [C2,C3]. Not only does this resolve Lam's problem to a more rigorous standard, it is also a much simpler solution, requiring no error-prone special-purpose search code. Indeed, my work uncovered previously undetected mistakes in not only the original search results from 1989 but also in the results of an independent verification from 2011 [C1,J2,J5]. In addition to these benefits, my solution of Lam's problem required less computing time than the previous solutions—one particularly challenging subcase was solved over 500 times faster when compared with the 2011 verification [C2].

3. Graph Theory

My work has led to dramatic improvements in industrial solvers of graph theoretic problems. Maplesoft funded my research in order to improve the efficiency of commands in Maple, the computer algebra system that they develop. In particular, I significantly sped up a number of commands in Maple's graph theory package by using SAT-based techniques. A number of problems that previously required hours to solve can now be solved in seconds in the latest version of Maple [C4]. For example, Maple can now solve much larger examples of the clique finding and graph colouring problems—problems that have important applications to bioinformatics, computational chemistry, map drawing, and resource scheduling.

4. Complementary Sequences

Complementary sequences are widely studied for their use in fields such as signal processing and number theory. My work has discovered new examples of such sequences and confirmed or refuted previous conjectures. For example, my work confirmed a 2002 conjecture that length 23 complex Golay sequences do not exist [C7,J1]. Moreover, I have published the only freely available code and data in the search for complex Golay sequences. This allows other researchers to use our technology and results in their own work.

I have also successfully applied MathCheck to several classes of matrices defined by

complementary sequences [C6,J6]. For example, I discovered three new counterexamples to the conjecture that good matrices always exist in odd orders, discovered two new sets of good matrices (including one that was overlooked by at least two previous searches), found the largest currently known best matrices, and showed that a conjecture about best matrices is true for larger orders than was previously known.

5. Minimal Primes

My work has also resolved some open problems in the area of minimal primes [J9]. A prime number is minimal if removing digits from its decimal representation cannot yield any smaller prime number. Surprisingly, it can be shown that there are only finitely many minimal primes—not only in decimal (base 10) but in any base. However, there is no known algorithm that can find all minimal primes in a given base and solving this problem in bases other than 10 was open since 2000. Despite the fact that the problem is not known to be decidable in general bases, I devised a heuristic approach that was able to resolve the question in all bases up to 16, and in all bases up to 30 with a possibly a small number of missed minimal primes. The results were simply astonishing: certain bases have some enormous minimal primes. The largest minimal prime in base 23 has over a million digits when represented in decimal. This number was checked using a probabilistic prime test and at the time of its discovery in 2016 was the tenth largest probable prime ever discovered. This research was only intended to be recreationally interesting, but it turns out that computing the "minimal strings" of DNA strings (by generalizing the minimality concept to strings in an arbitrary language) has applications to DNA strand design [ref 4].

Additional Information on Contributions

All of the research described in my most significant contributions was implemented and written by myself while taking into account feedback from my collaborators. In all of my papers the author order reflects the contribution amount and I have been listed as the first author in all of my papers with a single exception.

The open-source MathCheck project was started by professor Vijay Ganesh. As a PhD student under his supervision I became its lead developer and I implemented the routines used to search for combinatorial and number theoretic matrices and sequences. We joined with professor Ilias Kotsireas and he provided domain-specific knowledge that I incorporated into the system in order to make it more effective.

Because the research on MathCheck combined both the fields of satisfiability (SAT) and computer algebra we chose to publish these results in some of the top venues in these fields, such as ISSAC [C7] and CASC [C9] (two of the top conferences in computer algebra), as well as twice at the top-tier artificial intelligence (AI) conference AAAI [C6,C8]. As the authors of the first hybrid SAT and computer algebra system we also chose to publish in the first SC-square workshop [C10] (a workshop devoted to combinations of SAT solving and computer algebra) and were invited to publish our work in a special "SC-square" issue of the Journal of Symbolic Computation [J3] (the top journal in the field) and a special "best papers at CADE 2015" issue of the Journal of Automated Reasoning [J8].

After finishing my PhD, I further developed MathCheck in order to handle problems from other domains of mathematics. I started collaborating with the mathematicians Dragomir Djokovic, Kevin Cheung, and Brett Stevens. We decided to publish our work in some well-known mathematical journals that focus on applications of AI or computer algebra, such as AMAI [J6] and AAECC [J5], as well as in the top-tier AI conference IJCAI [C2]. We also published a construction for Williamson matrices in the IEEE Transactions of Information Theory [J4]—chosen because it is a prestigious journal that has published many other constructions for matrices and sequences over its long history.

I have also worked with industry—I met with Jürgen Gerhard (Maplesoft director of research) and pitched to him the idea of applying my research in SAT solvers to speed up some commands in the computer algebra system Maple. He agreed to fund this research and over the period of about two months I greatly improved the efficiency of several satisfiability and graph theory commands in Maple—functionality that is now being used by engineers and scientists around the world to complete their own work more effectively. This research was published at the Maple conference in 2020 [C4]—chosen because this venue would be able to reach a large number of educators and researchers in mathematics and computer algebra.

Satisfiability Checking and Computer Algebra: Proposal

Efficient search methods are essential to solve a huge number of problems like searching the internet, allocating resources, and finding the most efficient route when travelling. Some of the best general-purpose search methods come from the field of satisfiability (SAT) solving, but these struggle with complicated mathematical constraints. Conversely, some of the best methods for solving mathematical problems come from the field of computer algebra. However, there are many problems that require both sophisticated mathematics and powerful general-purpose search. My research program will combine both satisfiability and computer algebra techniques in order to advance a search method achieving the best of both worlds—for fast and mathematically powerful searches.

Recent Progress. Since 2016, I have been the lead developer of the MathCheck prototype tool[C9]—the first tool that performs mathematical searches by combining the functionality of SAT solvers with computer algebra systems (CAS). Using MathCheck I have resolved and made progress on problems from combinatorics,[J3] number theory,[J1] geometry,[C3] and graph theory[C4]—described in detail under my "most significant contributions".

Objectives. The long-term vision of my research program is to make the SAT+CAS method one of the most effective—if not the most effective—methods for solving mathematical search problems. The short-term objectives are to work towards making this vision a reality—by extending MathCheck to new domains in order to apply its unique capabilities to problems that cannot be solved by any existing methods. Using MathCheck we can implement more efficient, verifiable, and rigorous searches. Unlike most traditional search techniques, MathCheck provides certificates that can be used to rigorously show nonexistence when a solution to a problem doesn't exist.[C2]

Literature Review. The SAT+CAS paradigm was proposed by Ábrahám[1] and independently by Zulkoski et al.[27] in 2015. They pointed out that while "SAT modulo theories" solvers offer more mathematical functionality than SAT solvers they lack the kind of expressive mathematical functionality available in CASs. Since 2015, a small but budding community known as "SC-square" (for satisfiability checking and symbolic computation[7]) has started to organize around applications of this paradigm—such as solving polynomial systems over real numbers,[19] cryptanalysis,[13] program synthesis,[15] combinatorial object construction,[14] and circuit verification.[17,20] Despite the impressive variety of applications discovered so far, the potential of the paradigm has yet to be fully realized—there are a number of branches of science, engineering, and mathematics ripe with problems for which the SAT+CAS paradigm has not yet been applied. My research program will remedy this situation by extending the paradigm to a number of new domains of interest.

Methodology. The majority of my research will be accomplished by the highly qualified personnel who will be trained by myself (and other professors who have indicated

their interest) in satisfiability and computer algebraic methodologies—and how they can be judiciously combined. The professors involved include Ahmad Biniaz (Windsor), Vijay Ganesh (Waterloo), Kevin Cheung and Brett Stevens (Carleton), Ilias Kotsireas (Wilfrid Laurier), Supratik Chakraborty (IIT Bombay, India), and Oliver Kullman (Swansea, UK). The students will be graduates or undergraduates from the University of Windsor (or will be co-supervised students at Carleton, where I will hold an adjunct professorship).

The personnel will be matched to projects based on their interests, background, and qualifications. There is no shortage of problems that mathematicians, scientists, and engineers care about where SAT+CAS methods can be impactful—I describe five new application domains below. In each case I provide details of an important problem in the area, how SAT+CAS methods will be used to make progress on the problem, a plan of how the research will be conducted by my research team, and a description of the benefits that will result once this research has been completed.

Application to Problems in Physics

Research in quantum physics relies on various combinatorial objects like Kochen–Specker (KS) systems—a set of three-dimensional vectors satisfying properties based on the laws of quantum physics. For example, using KS systems the mathematicians John Conway and Simon Kochen proved the "Free Will Theorem" that roughly says that if entities in the universe have free will then it follows that the fundamental particles that make up those entities also have free will.[5] Conway stressed the importance of finding smaller examples as the systems are important in devising experimental setups[3] and examples of small KS systems could provide insight into quantum theory and designing quantum computers.[21]

Opportunity. Despite extensive searches, we still don't know the minimal size of a KS system. The best known previous search relied on ad-hoc exhaustive search code utilizing the computer algebra library nauty to perform filtering.[24] We are now perfectly positioned to improve on this search by combining satisfiability checking and symbolic computation—exploiting the search power of a SAT solver while tailoring the search to incorporate the important filtering information provided by nauty.

Plan. This project will be worked on by a PhD student and will be directed by myself along with my collaborator Vijay Ganesh who has expensive experience using solvers to resolve problems in physics and mathematics. The project will result in searches for small KS systems—potentially leading to a smaller KS system or an improvement in the best known lower bound on the size of a KS system. This project marks the beginning of exploring the untapped potential of applying the SAT+CAS methods to problems in physics. Moreover, the skills and code developed by the student will be applied to further problems in quantum physics such as the nearest neighbor compliance problem which is important in designing quantum circuits.[25]

Application to Combinatorial Matrices and Pooled Testing

Consider the problem of testing for the presence of a virus in a set of samples. You can test each sample individually, but this is costly. To save resources one can pool together multiple samples and test them simultaneously, though this requires the development of efficient pooling schemes. Disjunct matrices from combinatorics give rise to one source of efficient pooling schemes.[23] Methods are known for constructing disjunct matrices that asymptotically provide the optimal possible number of tests[9]—however, these methods tend to produce pools which are too large for many applications.

Opportunity. The SAT+CAS approach provides an ideal method to search for disjunct matrices and therefore pooling schemes. Traditional brute-force search methods are not powerful enough to find disjunct matrices of the size necessary to be useful in pooled testing, but SAT search techniques and CAS filtering techniques such as isomorphism rejection have been found to be useful in many similar searches for matrices.[J6] However, SAT and isomorphism rejection have not yet been used in the search for disjunct matrices—providing a unique opportunity to make progress on this difficult problem.

Plan. This project will be worked on by a PhD student directed by myself in association with my collaborators Kevin Cheung and Brett Stevens, who have expertise in disjunct matrices. They believe it is likely that disjunct matrices exist in the sizes that would be useful in high-frequency pooled testing for viruses like COVID-19. This project will search for such matrices and will result in either a collection of matrices suitable for this purpose or a nonexistence proof that such matrices do not exist. Additionally, the skills and code developed by the student will form the basis of searches for other other classes of combinatorial matrices such as those defined by strongly regular graphs.[6]

Application to Graph Colouring Problems

In 1950, the mathematician Ed Nelson asked how many colours are necessary to colour every point on an infinite sheet of paper—supposing that points that are exactly separated by some fixed distance are coloured differently. Within a year it was shown that it is possible to colour all points up to this constraint using only seven colours and that at least four colours are necessary.[22] However, despite these early results the problem of determining the minimal number of colours necessary remains an open problem. No progress was made on improving the best known bounds until 2018, when Aubrey de Grey showed that it is necessary to use at least five colours by improving our understanding of certain kinds of graphs in graph theory.[8]

Opportunity. We now have a great opportunity to make further progress on Nelson's question by using modern SAT solvers combined with computer algebraic methods. SAT solvers excel at search in colouring problems—however, by themselves they cannot be used in this problem as they have no conception of distance or two-dimensional points. The

current best known result is a collection of 529 two-dimensional points that need at least five colours to be consistently coloured—found by some ad-hoc construction methods and using a SAT solver to verify the colouring.[11] Because CASs excel at algebraically dealing with points we now have an opportunity to incorporate CASs directly as a part of the SAT solver's search.

Plan. This project will be worked on by a postdoctoral fellow directed by myself in collaboration with Ahmad Biniaz. The postdoctoral fellow will implement a SAT+CAS method to search for collections of two-dimensional points that require at least five colours to colour consistently. I will supply the direction in terms how the SAT solver can be connected to a CAS, while Ahmad will supply the domain expertise in computational geometry. The project will run searches for collections of two-dimensional points that require five or more colours in order to consistently colour. The searches have the potential to improve the lower bound in Nelson's problem or find a collection of five-colourable points that improves on the current record. Either way, these searches will provide insight into developing and running searches for other kinds of colouring problems like those derived from Latin squares—which have numerous practical and theoretical applications.[26]

Application to Circuit Minimization

A circuit is a piece of electronics that produces a "bit" (a high or low electronic signal) as output when given a collection of bits of input. They are typically implemented by combining together a number of simpler circuits known as gates. The circuit minimization problem is to find another circuit that produces the same output as some given circuit—but instead uses a minimal number of gates. The problem is important to the design of computers, since they consist of a huge number of electronic circuits and would be produced more efficiently if those circuits were minimized. Indeed, it has been estimated that improved methods of minimizing circuits would be worth millions to the world's economy.[16] Circuit minimization also applies to quantum circuits, which often need to be specified so that gates only take adjacent signals as input.[25] This can be achieved by using "swap" gates—which then also have to be minimized.

Opportunity. This problem is very difficult in general and some traditional methods involve algebraic simplifications. Progress has recently been made using SAT solvers to minimize certain special types of circuits. For example, using off-the-shelf SAT solvers new minimal modular arithmetic (MOD) circuits have been discovered[18] as well as new minimal quantum circuits.[25] These SAT solvers do not take advantage of the algebraic simplifications and isomorphism removal used by previous methods—thus for the first time we have the opportunity to apply the algebraic simplifications which can be done by a CAS in tandem with the search power of a SAT solver. Such an approach has not yet been used though recently a number of small circuits for 3×3 matrix multiplication were discovered by using a SAT solver to search for partial solutions which were then postprocessed to full solutions using a CAS.[12]

Plan. This project will be worked on by two MSc students (one of whom, Abinaya Venkatesan, has already agreed to work on the project) and later by a PhD student. The MSc students will be directed by myself and my collaborators Vijay Ganesh and Supratik Chakraborty. The students will work on finding minimal circuits for MOD circuits and matrix multiplication circuits. This project will result in searches for small minimal circuits of both types and thereby improve our knowledge of such circuits—either by exhibiting new unknown minimal circuits or proofs that circuits using a certain number of gates cannot exist. As the system becomes more tuned to such searches this project will also incorporate new kinds of circuit types—in the third year of this project a PhD student will begin working on minimizing quantum circuits using our SAT+CAS system.

Application to Recreational Mathematics and Discrete Tomography

Discrete tomography is the problem of reconstructing a two-dimensional image given a number of its one-dimensional projections. It is of use in image processing and electron microscopy but also forms the basis of some puzzles in recreational mathematics such as nonograms.

Opportunity. SAT solvers have been previously used to solve and generate puzzles like Sudoku[C4] and nonograms.[2] However, the algebraic aspects of discrete tomography have yet to be employed in an automated reasoning solver.[10]

Plan. The types of problems that arise in this domain—such as developing more efficient puzzle solvers and generators—will be worked on by one undergraduate student per summer over the next five years. These problems are an ideal fit for undergraduates as they are simple to understand, do not require extensive background knowledge, and are small enough to make progress on over a couple of months. Additionally, they have been designed to pique the interest of students to get started in research.

Impact. These problems just scratch the surface of the of the SAT+CAS method's potential and speak to its wide applicability. This research will be impactful in three major ways: First, it will improve our understanding of a variety of problems—either by finding new solutions of the problems that were out-of-reach of previous search algorithms or by showing that such solutions do not exist. Second, it will allow us to develop more rigorous and less ad-hoc searches—for example, by eliminating the reliance on ad-hoc code and providing nonexistence certificates when searches come back negative. Third, it will improve our understanding of how best to structure mathematical searches for problems from a wide variety of domains. With a more through understanding of the issues involved we can start to implement general-purpose search tools that automatically structure the search in ways that lead to optimal performance. Such a general-purpose search tool would be a dream come true for many scientists and engineers—and the SAT+CAS method has potential to make this dream a reality.

Budget Justification

There are three main expenses necessary in order for this research proposal to be successful: salary expenses for the highly qualified personnel who will complete the research, travel expenses for presenting the research at conferences, and equipment expenses for the computers necessary in the course of the research.

Salary Expenses

Salary of the highly qualified personnel is the biggest and most important expense that will be required in order to ensure the success of this research program. The salary that will be paid to each highly qualified personnel will be based on their qualifications and breaks down into undergraduate (UG), Master's (MSc), PhD, and postdoctoral fellows (PDF). The timeline of hiring for these personnel is depicted in Table 1.

The undergraduates will be paid $3,000 for four months of work on a single project. I will work with one undergraduate a year for the next five years—a total expense of $15,000.

The Master's students will each be paid $6,000/year from this grant (with their remaining salary covered by teaching and research assistantships offered through the school). I will work with two Master's students over the next five years and expect to work with them for approximately 2.5 years each—a total expense of $30,000.

The PhD students will each be paid $12,000/year from this grant (with their remaining salary covered by teaching and research assistantships offered through the school). I will work with two PhD students at a time for the next two years and have a third PhD student join my group after two years. One PhD student will be funded through my startup grant—a total expense of $96,000 to be paid by NSERC.

A postdoctoral fellow will be paid $40,000/year from this grant in the first two years during which time they will be able to work on research nearly full-time in order to help establish the research program. After the first two years a postdoctoral fellow will be paid $30,000/year from this grant with their remaining salary provided through alternative means

Topic	Year 1	Year 2	Year 3	Year 4	Year 5
Physics	PhD1	PhD1	PhD1	PhD1	PhD1
Pooled Testing	PhD2	PhD2	PhD2	PhD2	PhD2
Graphs	PDF1	PDF1			
			PDF2	PDF2	PDF2
Circuits	MSc1	MSc1			
			MSc2	MSc2	MSc2
			PhD3	PhD3	PhD3
Tomography	UG1	UG2	UG3	UG4	UG5

Table 1: Timeline of the highly qualified personnel who will be trained as a part of this research program. The personnel are organized by the five overarching topics presented in the detailed proposal.

such as sessional teaching. I will work with a single postdoctoral fellow at a time for the next five years—leaving a total expense of $170,000 to be paid by NSERC.

This salaries were chosen to be competitive with the salaries offered by other professors for personnel of similar qualifications. They are necessary in order to attract the kind of highly qualified personnel necessary in order to produce the high-quality research that this grant will enable.

Travel Expenses

Travel is essential in order disseminate the research that will be produced into the research community at large. In particular, attending conferences will be essential in order for the graduate students and postdoctoral fellows to promote their work in the research community as well as essential to their development as scholars who can effectively communicate their work. Due to current travel restrictions some of this dissemination may occur online—however, there is still no virtual equivalent of face-to-face contact and thus some form of travel will still be necessary. On average, I have budgeted for each graduate student to travel to one major conference every year, and for each postdoctoral fellow (as well as myself) to travel to two major conferences every year. This works out to seven trips a year, with the trips of one PhD student being covered by my startup grant. The remaining six trips are budgeted at $2,000 each (including conference fees as well as food, flight, and hotel costs). Altogether, these expenses total $12,000/year for the first two years (and $14,000/year for years three to five) to be paid by NSERC.

Equipment Expenses

Although my research program does not require significant amount of specialized equipment, each personnel involved will need a laptop or desktop machine in order to complete their research. Each machine has been budgeted at a cost of $1,500 and it is assumed that the that machines may need to be replaced once over the course of five years. With up to six personnel working at any one time, this requires a total of expense of $16,500 over the course of the next five years. An additional $2,500/year has been budgeted for the costs associated with completing the computational component of the research. This will cover the specialized software licenses (e.g., Maple), cloud computing fees (e.g., Amazon EC2) and storage to save the data collected (e.g., Amazon Drive).

Total Expenses

In total, I have budgeted: $311,000 in salary costs to be paid by NSERC, $66,000 in travel costs to be paid by NSERC, and $29,000 in equipment costs to be paid by NSERC.

References

J*x* and C*x* footnotes reference journal and conference publications that appear in my CCV.

[1] E. Ábrahám. Building bridges between symbolic computation and satisfiability checking. In *Proceedings of the 2015 ACM on International Symposium on Symbolic and Algebraic Computation*, pages 1–6. ACM, 2015.

[2] D. Berend, D. Pomeranz, R. Rabani, and B. Raziel. Nonograms: Combinatorial questions and algorithms. *Discrete Applied Mathematics*, 169:30–42, 2014.

[3] A. Cabello. Kochen–Specker theorem and experimental test on hidden variables. *International Journal of Modern Physics A*, 15(18):2813–2820, 2000.

[4] D.-J. Cho, Y.-S. Han, and S.-K. Ko. Decidability of involution hypercodes. *Theoretical Computer Science*, 550:90–99, 2014.

[5] J. Conway and S. Kochen. The free will theorem. *Foundations of Physics*, 36(10):1441–1473, 2006.

[6] K. Coolsaet, J. Degraer, and E. Spence. The strongly regular $(45, 12, 3, 3)$ graphs. *The Electronic Journal of Combinatorics*, pages R32–R32, 2006.

[7] J. H. Davenport, M. England, A. Griggio, T. Sturm, and C. Tinelli. Symbolic computation and satisfiability checking. *Journal of Symbolic Computation*, 100:1–10, 2020.

[8] A. de Grey. The chromatic number of the plane is at least 5. *Geombinatorics*, 28:18–31, 2018.

[9] A. G. D'yachkov and V. V. Rykov. Bounds on the length of disjunctive codes. *Problemy Peredachi Informatsii*, 18(3):7–13, 1982.

[10] L. Hajdu and R. Tijdeman. Algebraic aspects of discrete tomography. *Journal für die reine und angewandte Mathematik (Crelles Journal)*, 2001(534):119–128, 2001.

[11] M. J. H. Heule. Trimming graphs using clausal proof optimization. In T. Schiex and S. de Givry, editors, *Principles and Practice of Constraint Programming*, pages 251–267, Cham, 2019. Springer International Publishing.

[12] M. J. H. Heule, M. Kauers, and M. Seidl. Local search for fast matrix multiplication. In M. Janota and I. Lynce, editors, *Theory and Applications of Satisfiability Testing – SAT 2019*, pages 155–163, Cham, 2019. Springer International Publishing.

[13] J. Horáček. *Algebraic and Logic Solving Methods for Cryptanalysis*. PhD thesis, University of Passau, 2020.

[14] P. Huang, M. Liu, C. Ge, F. Ma, and J. Zhang. Investigating the existence of orthogonal golf designs via satisfiability testing. In J. H. Davenport, D. Wang, M. Kauers, and R. J. Bradford, editors, *Proceedings of the 2019 on International Symposium on Symbolic and Algebraic Computation*, pages 203–210. ACM, 2019.

[15] J. P. Inala, S. Gao, S. Kong, and A. Solar-Lezama. ReaS: combining numerical optimization with SAT solving. *arXiv preprint arXiv:1802.04408*, 2018.

[16] A. B. Kahng, J. Lienig, I. L. Markov, and J. Hu. *VLSI Physical Design: From Graph Partitioning to Timing Closure*. Springer, 2011.

[17] D. Kaufmann, A. Biere, and M. Kauers. Verifying large multipliers by combining SAT and computer algebra. In C. W. Barrett and J. Yang, editors, *2019 Formal Methods in Computer Aided Design (FMCAD)*, Formal Methods in Computer Aided Design, pages 28–36, 2019.

[18] A. Kojevnikov, A. S. Kulikov, and G. Yaroslavtsev. Finding efficient circuits using SAT-solvers. In O. Kullmann, editor, *Theory and Applications of Satisfiability Testing - SAT 2009*, pages 32–44, Berlin, Heidelberg, 2009. Springer Berlin Heidelberg.

[19] G. Kremer and E. Ábrahám. Fully incremental cylindrical algebraic decomposition. *Journal of Symbolic Computation*, 100:11–37, 2020.

[20] V. Liew, P. Beame, J. Devriendt, J. Elffers, and J. Nordström. Verifying properties of bit-vector multiplication using cutting planes reasoning. In A. Ivrii and O. Strichman, editors, *Proceedings of the 20th Conference on Formal Methods in Computer-Aided Design – FMCAD 2020*, Formal Methods in Computer-Aided Design, pages 194–204, 2020.

[21] M. Pavičić, J.-P. Merlet, B. McKay, and N. D. Megill. Kochen–Specker vectors. *Journal of Physics A: Mathematical and General*, 38(7):1577, 2005.

[22] A. Soifer. *The mathematical coloring book*. Springer Science & Business Media, 2008.

[23] M. Täufer. Rapid, large-scale, and effective detection of COVID-19 via non-adaptive testing. *BioRxiv*, 2020.

[24] S. Uijlen and B. Westerbaan. A Kochen-Specker system has at least 22 vectors. *New Generation Computing*, 34(1-2):3–23, 2016.

[25] R. Wille, A. Lye, and R. Drechsler. Optimal SWAP gate insertion for nearest neighbor quantum circuits. In *2014 19th Asia and South Pacific Design Automation Conference (ASP-DAC)*, pages 489–494. IEEE, 2014.

[26] O. Zaikin and S. Kochemazov. The search for systems of diagonal Latin squares using the SAT@home project. *International Journal of Open Information Technologies*, 3(11):4–9, 2015.

[27] E. Zulkoski, V. Ganesh, and K. Czarnecki. MathCheck: A math assistant via a combination of computer algebra systems and SAT solvers. In *International Conference on Automated Deduction*, pages 607–622. Springer, 2015.