RESEARCH STATEMENT

Xinyu Sun

My focus has been on computer-generated and computer-assisted research since my Ph.D. years, and it has persisted long after my graduation. My research covers problems in combinatoris, with emphasis on algebraic combinatorics, WZ-algorithm and combinatorial games. The emergence of powerful mathematical computing environments, growing availability of correspondingly powerful computers and the pervasive presence of the Internet allow mathematicians to attack problems thought of being impossible just a decade ago. Symbolic and numeric computation are utilized to search results, and whenever possible, prove them without human intervention.

Problems of great intereste to my research are related to WZ-algorithm and its generalization into mutli-variant summation; algebraic-combinatorial approach to systems of polynomials and their applications in enumeration of lattice paths, specifically in parking functions and their generalizations; and a variety of combinatorial games.

Finding recursions over the summation of some binomial coefficients is a common problem in all disciplines of sciences, and finding an effective and efficient way to do so is crucial to solving a wide range of problems. Zeilberger answered the question once and for all for summations over a single variable, but the problem persists for summations over multi-variant summations. Applying symbolic computing, with Garoufalidis, we developed a new algorithm that could find such recursions in a computer-automated process that is faster than other known algorithms.

Parking functions was first introduced by Konheim and Weiss in the study of hashing function. The name comes from a picturesque description of parking n cars under certain preference rules. Parking functions have many interesting combinatorial properties, that is closely related to trees, labeled graphs, and lattice paths. It is this relationship that motivated early studies of parking functions. Other topics that are related to parking functions are hyperplane arrangement, non-crossing partitions, plane partitions, etc.

Mathematical game is a topic that is suitable for mathematicians in all different walks of their career. There are well-accomplished mathematians dedicated to mathematical games, and there are also graduate, undergraduate, and even high school students with reasonable, and sometime even astonishing, successes in games. Analyzing these games is difficult because a slight change in the rules of a game can result in a dramatic change in the winning strategy. Thus each game demands a new approach. At the mean time, it is still possible to use computers with targeted programming to find patterns in the solutions. Sometimes it is even possible prove human-generated conjectures with completely automated processes.

1. WZ-Algorithm, Multivariate Summation and Jones Polynomials

1.1. **WZ-Algorithm.** A *term* is $F(n,k)$ called *hypergeometric* if both $\frac{F(n+1,k)}{F(n,k)}$ and $\frac{F(n,k+1)}{F(n,k)}$ are rational functions over n and k. A key problem is to construct recursion relations for sums of the form

$$\sum_{i=0}^{N} a_i(n)S(n+i) = b(n), \text{ where } S(n) = \sum_k F(n,k). \tag{1.1}$$

As observed by Zeilberger, Sister Celine' constructive method showed such a recursion always exists. However her method is hard to compute and does not always give the optimal recursion. Applying Gosper's algorithm, Zeilberger invented an elegant algorithm, namely, Zeilberger's algorithm. Instead of looking for a hypergeometric term on the right-hand side of a recursion, the algorithm looks for a rational function, the *certificate*, $C(n,k)$ such that

$$\sum_{i=0}^{N} a_i(n)F(n+i,k) = C(n,k+1)F(n,k+1) - C(n,k)F(n,k). \tag{1.2}$$

While both algorithm converts the problem of finding recursions into solving system of linear equations, Zeilberger's is much faster, and almost always gives the optimal recursion.

However, it is a much different problem to find recursions over multi-variant summations (multisums), namely, recursions over $S(n)$, where

$$S(n) = \sum_{k_1,\ldots,k_\ell} F(n,k_1,\ldots,k_\ell)$$

for a hypergeometric function $F(n,k_1,\ldots,k_\ell)$. Due to the nature of Gosper's algorithm, Zeilberger's algorithm cannot be applied to multisums. Even though Sister Celine's algorithm still works, it is extremely time consuming. Let alone finding optimal recursions.

Many people have had success on developing new algorithms with different level of successes. Wegschaider improved Sister Celine's algorithm; Apagodu and Zeilberger developed an algorithm that extended Gosper's algorithm; and Schneider developed another program, Sigma, that tried to tackle the problem iteratively. All these programs also reduce the problem into solve systems of linear equations, and the performance inevitably depends on the numbers of variables and the numbers of equations in the systems.

With Garoufalidis, we developed another algorithm, iSum, that eliminated one variable in each step, so that the systems involved in the steps were manageable. At the same time, in each step, we employed another algorithm that improved Abramov's algorithm, which found the denominators for rational solutions to the equations

$$\frac{a_m(n)}{b_m(n)}x(n+m) + \cdots + \frac{a_0(n)}{b_0(n)}x(n) = \frac{c(n)}{d(n)},$$

where $a_i(n), b_i(n), c(n), d(n)$ are polynomials. The new algorithm dramatically reduced the sizes of the systems of linear equations in each step, and as a result, the performance was better than those listed above.

Even with the new algorithm, there are still a lot of questions to be answered. It happens because they are summations too complicated to handle, or because they are not exactly hypergeometric, but still closely resemble those. In either case, improving the performance is constantly the goal of developing new software, and at the same time, we also need to develop new theories to tackle those summations that are closely related to hypergeometric functions.

1.2. **Twist knows and A-polynomials.** As an application to our algorithm, we applied an earlier version of the idea to the q-analog.

The problem started in quantum topology. Consider the family of *twist knots* K_p for integer p. Let $\hat{J}_p(n)$ denote the cyclotomic function of K_p, and we have:

$$\hat{J}_p(n) = \sum_{k=0}^{\infty} q^{n(n+3)/2+pk(k+1)+k(k-1)/2}(-1)^{n+k+1}\frac{(q^{2k+1}-1)(q;q)_n}{(q;q)_{n+k+1}(q;q)_{n-k}}, \tag{1.3}$$

Together with S. Garoufalidis, we gave the explicit formulas of the non-commutative *characteristic polynomial* of the twist knot K_p, in other words, the recursions of $\hat{J}_p(n)$. The degrees of the polynomials are the absolution values of p, for all p.

The non-commutative A-polynomial of twisted knots in 3-space, defined by S. Garoufalidis, is the characteristic polynomial of

$$A_p(n) = \sum_{k=0}^{\infty} (-1)^k q^{-k(k+1)/2}(q^{1-n};q)_k(q^{1+n};q)_k \hat{J}_p(k),$$

which is effectively a double-sum. We were able to find the optimal recursions by taking advantage of knowledge the recursion of the inner sum, $\hat{J}_p(k)$. The results partially confirmed his earlier conjecture that says the polynomial specializes at $q = 1$ to the better known A-polynomial of a knot.

2. Goncarov Type Polynomials, Parking Functions and Lattice Paths

2.1. **Two-boundary lattice paths and parking functions.** We can define $\underline{r} = (r_0, r_1, \cdots)$ and $\underline{s} = (s_0, s_1, \cdots)$ as two sequences of non-negative integers, thought of as *left and right boundaries*. An $(\underline{r}, \underline{s})$–lattice path of length n is defined to be non-decreasing sequence $(x_0, x_1, \cdots)$ such that $r_i \leq x_i < s_i$. We denote by $\mathrm{LP}_n(\underline{r}, \underline{s})$ the number of all $(\underline{r}, \underline{s})$–lattice paths of length n. Together with Kung and Yan, by inspecting the Goncarov polynomials, we showed that

$$\mathrm{LP}_n(\underline{r}, \underline{s}) = \det\left[\binom{(s_i - r_j)_+}{j-i+1}\right]_{0\le i,j\le n-1}. \tag{2.1}$$

Parking functions are rearrangements of lattice paths. An $(\underline{r}, \underline{s})$–parking function of length n is a sequence of positive integers $(x_0, x_1, \cdots)$ such that its rearrangement $(x_{(0)}, x_{(1)}, \cdots)$ into a non-decreasing sequence satisfies the inequality $r_i \le x_{(i)} < s_i$. We shall denote the number all $(\underline{r}, \underline{s})$–parking function of length n by $\mathrm{PF}_n(\underline{r}, \underline{s})$.

We have

$$\sum_{i=0}^{n}(-1)^i\binom{n}{i}(s_i - r_i)_+^{n-i}\mathrm{PF}_i(\underline{r}, \underline{s}) \quad = \quad \delta_{n,0}, \tag{2.2}$$

which also gives us

$$\mathrm{PF}_n(\underline{r}, \underline{s}) = \det\left[\frac{(s_i - r_j)^{j-i+1}}{(j-i+1)!}\right]_{0\le i,j\le n-1}. \tag{2.3}$$

A version of equation 2.1 and 2.3 were obtained by Steck earlier. We gave two proofs of equation 2.2 by creating a bijection and using inclusion-exclusion.

The combinatorial involution also yields weighted enumeration formulas. The *area or sum enumerator* $\mathrm{Area}(q; H)$ of H is the polynomial in the variable q defined by

$$\mathrm{Area}(q; H) = \sum_{(x_0, x_1, \ldots, x_{n-1})\in H} q^{x_0+x_1+\ldots+x_{n-1}}.$$

we define $\mathrm{Area}(q; \underline{r}; \underline{s})$ to be the area enumerator of $\mathrm{Path}_n(\underline{r}; \underline{s})$, then it equals the determinant obtained from determinant 2.1 by the substitutions

$$\binom{(s_i - r_j)_+}{j-i+1} \longrightarrow q^{r_j(j-i+1)+\binom{j-i+1}{2}}\binom{(s_i - r_j)_+}{j-i+1}_q.$$

Similar substitution also works for the $(\underline{r}; \underline{s})$–parking functions.

With Kung, de Mier and Yan, we later proved that the generating functions with periodic boundaries are algebraic.

With Kostic, we generalized the definition of (p, q)-parking functions, created by Cori and Poulalhon. We showed there is an equivalent mapping from (p, q)-parking functions to pairs of non-crossing paths. Using Goncarov-type polynomials, we showed that the enumeration of the generalized (p, q)-parking functions are polynomials too.

It will be interesting to find the combinatorial correspondence of the (p, q)–parking functions in each case, and constructive bijection between them. We can also investigate the algebraic properties and combinatorial consequences of other polynomials. Examples can be seen that Faber polynomials occurred in counting lattice paths, Poisson–Charlier polynomials in counting partitions and various q–polynomials in counting in vector spaces over a finite field.

3. Mathematical Games and Computer Proofs

Mathematical games, or combinatorial games, are two-player games with complete information (the players knows all the information about the game), no chance moves (no dice), a number of, usually finite many, positions, and the outputs are strictly win/lose or draw/draw.

Such games are interesting to us because they are quite unlike the traditional existential decision and optimization problems. Whereas in the existential decision problems area, there are only few problems whose complexity has not yet been determined, the complexity of the majority of combinatorial game is still unknown. A study of the precise nature of the complexity of those games enables us to attain a deeper understanding of the difficulties involved in certain new and old open game problems, which is a key to their solution. Research of these games can also lead us to new and interesting algorithmic challenges, in addition to the fun of playing games.

3.1. **Chomp.** *Chomp* is a two-player game that starts out with an $M \times N$ chocolate bar, in which the square on the top left corner is poisonous. A player must name a remaining square, and eat it together with all the squares below and/or to the right of it. Whoever eats the poisonous one (top-left) loses. The game can also be interpreted as two players alternately name a devisor of a given number N, which may not be multiples of previously named numbers. Whoever names 1 loses.

One of my earlier results proved a formula for P-position of certain types, which extended the one given by Berlekamp, Conway and Guy.

Another aspect of the research is to use computers to prove theorems. Computers have been used extensively to assist researchers to find a massive number of discrete results quickly and easily. However, using computers to prove theorems is still at its infancy. This is because computers in general do not understand logic and cannot calculate up to infinity. Using symbolic computing, Zeilberger proved that with three-rowed Chomp, when the last rows have up to 115 squares, there are either finitely many P-positions, or the differences between the first two rows will eventually be constants. It was later proved that for certain types of positions with more than three rows, there exist periodic patterns between positions and their nim values, in one of my papers. Similar patterns also exist among the P-positions.

All these results are proved by computers without the need of human intervention, and each one demonstrates that with careful design, computers can be used to prove results involving infinity. Byrnes later proved my conjecture that both of the results are true for all the P-positions with the corresponding top rows fixed.

3.2. **Wythoff's Game.** *Wythoff's game* is an impartial game consisting of two piles of tokens. Players are to remove any number of tokens from a single pile, or the same number of tokens from both piles. The first player that cannot make a move loses. The game and its winning positions are well analyzed and explained in several papers.

Fraenkel gave a natural generalization of the game for any number of piles of tokens, with two conjectures stating that the P-positions of the new game satisfy similar properties of those of the Wythoff's game.

With Zeilberger, we proved that both conjectures are correct for three-heap Wythoff's game when the first heap has up to 10 tokens using computers. It was later shown that the first conjecture implies the second by me. Fraenkel and Krieger also proved the last result separately.

There are always interesting games being discovered and investigated. Even for existing games, different variations have generated surprising results with enormous interest. Such studies may lead us to find better understanding why a small change of the rules can yield totally different patterns in results, and thus completely different winning strategies for the games. The games can give us problems in complexity classes beyond P and NP, including Pspace, Exptime, and Expspace. There are also connections to other areas of interest, such as complexity, logic, graph, networks. Besides of the two player games we discussed above, one player or multi-player games can be of great interest too.

4. Future Research Plan

There are many interesting problems remaining in all different aspects of my research, and it is my plan to keep advancing my development of the recent years. To illustrate the potential of my research, problems of interest in the near future follow.

(1) This is an interesting problem for atomic chemistry, spin networks, and even quantum topology. Let

$$\begin{aligned} \mathbf{m} &= (a,b,c,d,e,f); \\ f(\mathbf{m},n,k) &= \frac{(-1)^k (k+1)!}{(na+nb+nc+nd-k)!(na+nd+ne+nf-k)!(nb+nc+ne+nf-k)!} \\ &\quad \cdot \frac{1}{(k-na-nb-ne)!(k-na-nc-nf)!(k-nc-ne-nd)!(k-nd-nb-nf)!} \end{aligned}$$

and try to find the recursion of the function $S(\mathbf{m},n) = \sum_k f(\mathbf{m},n)$. A close inspection shows that the function is *not* hypergeometric for a generic $\mathbf{m}$, therefore the algorithms developed before are not directly applicable. However there are some interesting features involving specific values of $\mathbf{m}$, and special care will be needed to see if those features can be developed into a general recursion for $S(\mathbf{m},n)$ with generic $\mathbf{m}$.

(2) The following questioins started from quantum topology too. Let

$$\begin{aligned} f(n,r,s) &= \frac{(-1)^s(s+1)!}{(4n-s)!(3n+r-s)!^2(-3n+s)!^2(-2n-r+s)!^2}, \\ S(n,r) &= \left(\sum_s f(n,r,s)\right)^4, \\ S(n) &= \sum_r (2r+1)S(n,r), \end{aligned}$$

and try to find the recursion for $S(n)$.

This is effectively a function with five levels of summations. Even with our new algorithm iSum, the recursions get quickly too complicated to compute. So further improving the performance of our algorithm, or developing new algorithms, is still the focus of my research in the foreseeable future.

(3) The binomial coefficients $\binom{n}{k}$ is one set of the most recognizable numbers in mathematics, and it is also well known that they are log-concave, i.e., given a function $f(k)$, define an operator

$$L : f(k) \longrightarrow g(k) = f(k)^2 - f(k-1)f(k+1).$$

Then $L\left(\binom{n}{k}\right) \geq 0$ for all n and k.

Nevertheless, it is still unknown to us if this is true:

$$L^m\left(\binom{n}{k}\right) \geq 0, \text{ for all } m, n, k.$$

(4) Mathematical games are always interesting objects to mathematician of all levels. It is my desire to introduce the ideas to undergraduate and graduate students alike. Well selected problems can stimulate students' interest in mathematics, motivate them to meet new challenges, and develop them into mathematicians in the future.