LEARNING THEORY ALLIANCE

Caltech

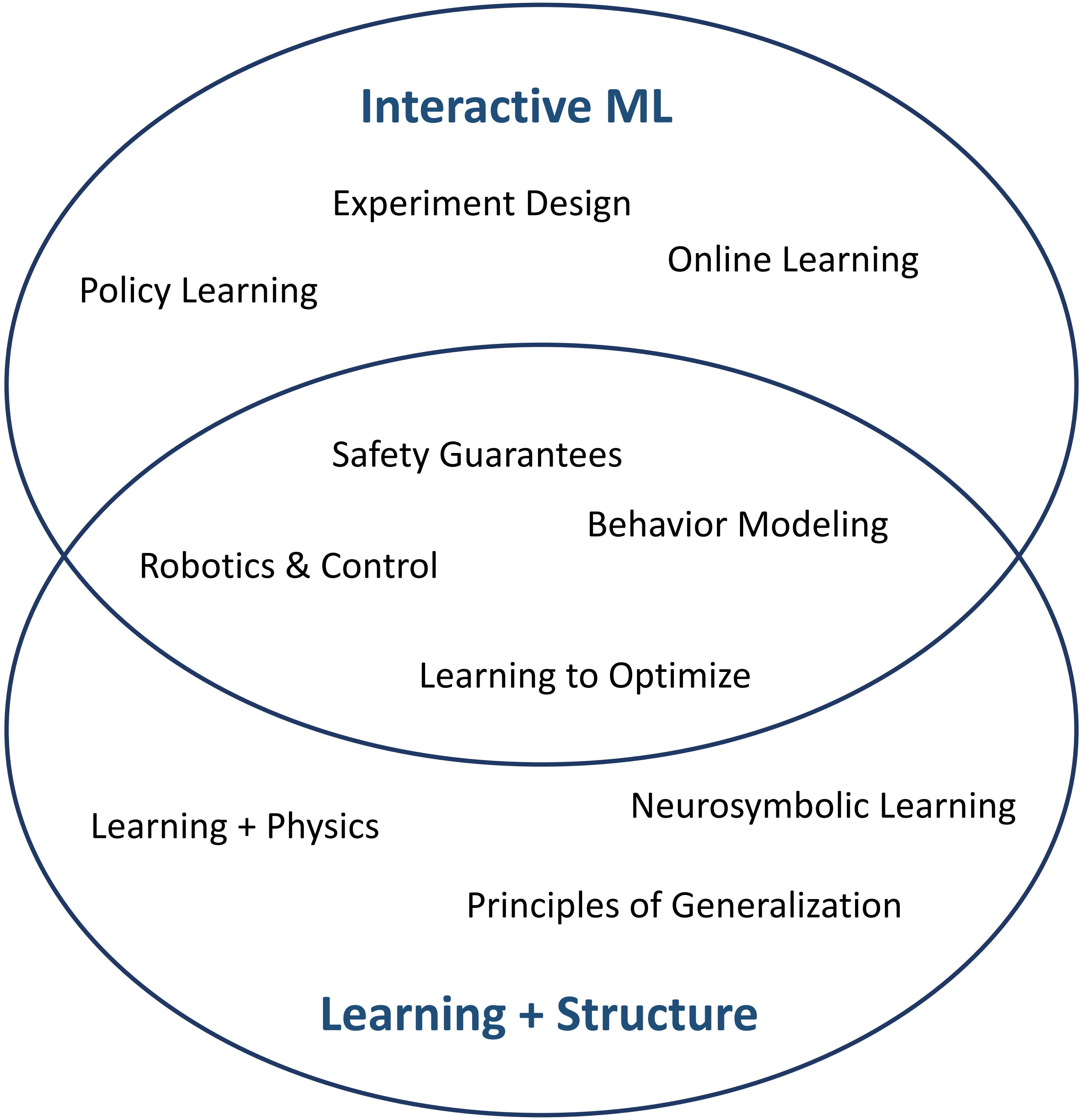
How to Write Statements of Purpose for PhD & Fellowship Applications

Yisong Yue

Yisong Yue

@yisongyue

Computing and Mathematical Sciences
Computation and Neural Systems
California Institute of Technology
http://www.yisongyue.com



Statements of Purpose

PhD Applications

Fellowships

Photo source: https://www.shutterstock.com/image-vector/education-vector-icon-student-cap-black-1804543951

PhD Applications

- Convey a sense of who you are as a researcher
  - How you think
  - What you’re passionate about
  - How you arrived at your career & research goals

- Complementary to other information:
  - Letters of Recommendation
  - Resume/Website
  - Transcript
  - Publications/Projects/Github

What Grad Admission Committees Look For

- Your intellectual motor
- What you care about
- A sense of continued growth
  - Your potential => you as a researcher 5 years from now
- (Everything else is found in other parts of your application.)

Do's

- Explain your intellectual journey in a substantive way.
- Use your previous experience as supporting evidence to explain this journey.
- Be honest with your level of commitment to specific areas.

Don't Do's

- "Ever since I was five, after watching the movie E.T., I wanted to…"
- Simply list your prior work without tying it together – redundant with resume.
- "I definitely want to do exactly quantum learning theory, and become a professor…"

Example Statement #1

Naama Ben-David

I am an undergraduate student at the University of Toronto, about to graduate with a B.Sc. in Computer Science and a minor in Linguistics. I have been involved in research in both computational linguistics and software engineering, and each of my projects has produced a result for publication. However, I find that my interests lie in theoretical computer science; I am currently working as a research assistant in the theory of distributed computing, and will be continuing this work in the next few months.

Unlike many of my peers, my interest in the field did not begin at a young age. I started my undergraduate studies intending to pursue a degree in linguistics. To my surprise, I enjoyed math courses more than language ones, and I found myself attracted to the field of computer science. By the third year of my studies, after an exchange at the Hebrew University of Jerusalem, I had reduced my linguistics program to a minor, and had decided to specialize in computer science.

⋮

⋮

In the spring of 2014, I worked with Professor Hirst and his PhD student in computational linguistics. Their project aims to use automatic speech recognition to diagnose Primary Progressive Aphasia. We developed a system that detects sentence boundaries in a stream of words from impaired speech samples. This allows analyzing sentence structure, which is important for calculating metrics used to distinguish between healthy and impaired speech. We have submitted our work for publication.

During the summer of 2014, I worked with Professor Chechik and her team in the field of software engineering. For the past few years, Professor Chechik's team has been working on introducing a concept of uncertainty to early stages of software design. This allows for developers to postpone some design decisions while continuing to work. Over the course of the summer, I was involved in creating a tool that handles uncertainty in software models; it allows for uncertainty to be expressed more easily, and has the ability to visualize and refine uncertain models based on given properties. We have submitted our developments on the tool for publication.

⋮

⋮

Earlier this month, I started working as a research assistant for Professor Hadzilacos and Professor Toueg in the theory of distributed computing. My interest in the field began in an independent study course that I took in the fall semester. This experience offered me a peek into the fascinating world of research in computer science theory. In the upcoming semester, we will be examining a class of shared objects, called Deterministic Abortable objects, to see how they compare to their wait-free counterparts. Linearizable wait-free objects are heavily studied in distributed computing, as they display powerful and attractive behavior. However, they are often expensive to implement. Deterministic Abortable objects exhibit slightly weaker, though still useful properties, and recent work has shown that they might offer a cheaper alternative.

⋮

⋮

During my undergraduate studies, I held teaching assistant positions in two different courses: Data Structures and Software Design, and I worked as a private tutor in an introductory Theory of Computing course. I find that I enjoy teaching – both the interaction with other students, and the opportunity to deepen my understanding of the material. In addition, I gained software development experience during an internship at BlackBerry, where I received an "Outstanding" ranking in my final worker evaluation – the highest possible ranking, only given out in special circumstances.

My growing passion for computer science, as well as my positive experiences doing research in the field, have motivated me to pursue graduate studies. I have been involved in research related to automatic speech recognition, software models and distributed computing, and I find that I am most excited by theoretical problems. I know that [redacted] has a very strong theory group, and I would be honored to become a graduate student at your department.

I am an undergraduate student at the University of Toronto, about to graduate with a B.Sc. in Computer Science and a minor in Linguistics. I have been involved in research in both computational linguistics and software engineering, and each of my projects has produced a result for publication. However, I find that my interests lie in theoretical computer science; I am currently working as a research assistant in the theory of distributed computing, and will be continuing this work in the next few months.

Unlike many of my peers, my interest in the field did not begin at a young age. I started my undergraduate studies intending to pursue a degree in linguistics. To my surprise, I enjoyed math courses more than language ones, and I found myself attracted to the field of computer science. By the third year of my studies, after an exchange at the Hebrew University of Jerusalem, I had reduced my linguistics program to a minor, and had decided to specialize in computer science.

In the spring of 2014, I worked with Professor Hirst and his PhD student in computational linguistics. Their project aims to use automatic speech recognition to diagnose Primary Progressive Aphasia. We developed a system that detects sentence boundaries in a stream of words from impaired speech samples. This allows analyzing sentence structure, which is important for calculating metrics used to distinguish between healthy and impaired speech. We have submitted our work for publication.

During the summer of 2014, I worked with Professor Chechik and her team in the field of software engineering. For the past few years, Professor Chechik's team has been working on introducing a concept of uncertainty to early stages of software design. This allows for developers to postpone some design decisions while continuing to work. Over the course of the summer, I was involved in creating a tool that handles uncertainty in software models; it allows for uncertainty to be expressed more easily, and has the ability to visualize and refine uncertain models based on given properties. We have submitted our developments on the tool for publication.

Earlier this month, I started working as a research assistant for Professor Hadzilacos and Professor Toueg in the theory of distributed computing. My interest in the field began in an independent study course that I took in the fall semester. This experience offered me a peek into the fascinating world of research in computer science theory. In the upcoming semester, we will be examining a class of shared objects, called Deterministic Abortable objects, to see how they compare to their wait-free counterparts Linearizable wait-free objects are heavily studied in distributed computing, as they display powerful and attractive behavior. However, they are often expensive to implement. Deterministic Abortable objects exhibit slightly weaker, though still useful properties, and recent work has shown that they might offer a cheaper alternative.

During my undergraduate studies, I held teaching assistant positions in two different courses: Data Structures and Software Design, and I worked as a private tutor in an introductory Theory of Computing course. I find that I enjoy teaching – both the interaction with other students, and the opportunity to deepen my understanding of the material. In addition, I gained software development experience during an internship at BlackBerry, where I received an "Outstanding" ranking in my final worker evaluation – the highest possible ranking, only given out in special circumstances.

My growing passion for computer science, as well as my positive experiences doing research in the field, have motivated me to pursue graduate studies. I have been involved in research related to automatic speech recognition, software models and distributed computing, and I find that I am most excited by theoretical problems. I know that [redacted] has a very strong theory group, and I would be honored to become a graduate student at your department.

Opening Summary

- Explains somewhat non-traditional background
- Conveys seriousness ("result for publication")
- Short and concise

General Research Experience

- Demonstrates broad understanding of what research is
- Demonstrates seriousness of wanting to do research
- Indirectly conveys a sense of wisdom => "I've explored many research projects and I know what I want."

Research Experience in TCS

- Conveys detailed understanding of the research direction
- Conveys substantive research experience in TCS

Other Relevant Experience

- Teaching experience is a nice plus

Conclusion

- Reiterate why you want to research in a certain area, and how your prior experiences support that

I am an undergraduate student at the University of Toronto, about to graduate with a B.Sc. in Computer Science and a minor in Linguistics. I have been involved in research in both computational linguistics and software engineering, and each of my projects has produced a result for publication. However, I find that my interests lie in theoretical computer science; I am currently working as a research assistant in the theory of distributed computing, and will be continuing this work in the next few months.

Unlike many of my peers, my interest in the field did not begin at a young age. I started my undergraduate studies intending to pursue a degree in linguistics. To my surprise, I enjoyed math courses more than language ones, and I found myself attracted to the field of computer science. By the third year of my studies, after an exchange at the Hebrew University of Jerusalem, I had reduced my linguistics program to a minor, and had decided to specialize in computer science.

In the spring of 2014, I worked with Professor Hirst and his PhD student in computational linguistics.
Their project aims to use automatic speech recognition to diagnose Primary Progressive Aphasia. We
develop ed speech
sample metrics used to
disting blication.

Consider adding 1-2 sentences reflecting on the experience. What was learned, what was insightful?

During d of software
enginee roducing a
concep postpone some
design lved in creating a
tool that essed more easily,
and has the ability to visualize and refine uncertain models based on given properties. We have submitted our developments on the tool for publication.

Earlier t and Professor
Toueg pendent study
course nating world of
researc a class of shared
objects free counterparts
Lineari play powerful
and att c Abortable
objects wn that they
might o

Consider being more specific about your specific project, and a sense of what you enjoy about this research.

During my undergraduate studies, I held teaching assistant positions in two different courses: Data Structures and Software Design, and I worked as a private tutor in an introductory Theory of Computing course. I find that I enjoy teaching – both the interaction with other students, and the opportunity to deepen my understanding of the material. In addition, I gained software development experience during an internship at BlackBerry, where I received an "Outstanding" ranking in my final worker evaluation – the highest possible ranking, only given out in special circumstances.

My gro esearch in the
field, h related to
automa hat I am most
excited nd I would be
honore

Considering being more specific about the target school.

Opening Summary

- Explains somewhat non-traditional background
- Conveys seriousness ("result for publication")
- Short and concise

General Research Experience

- Demonstrates broad understanding of what research is
- Demonstrates seriousness of wanting to do research
- Indirectly conveys a sense of wisdom => "I've explored many research projects and I know what I want."

Research Experience in TCS

- Conveys detailed understanding of the research direction
- Conveys substantive research experience in TCS

Other Relevant Experience

- Teaching experience is a nice plus

Conclusion

- Reiterate why you want to research in a certain area, and how your prior experiences support that

General Recipe

- Open with a statement of your interests, and a summary of your background. Use a few supporting highlights as appropriate (e.g., published a paper).

- Use your prior experience to provide evidence of your intellectual goals, why you came to have these interests, and what lessons you learned about what you want to study.

- State other relevant interests as appropriate, e.g., teaching.

- Summarize, and, as appropriate, comment on the research & faculty at the school you're applying to.

Statements of Purpose

PhD Applications

Fellowships

Photo source: https://www.shutterstock.com/image-vector/education-vector-icon-student-cap-black-1804543951

Fellowship Statements

- Convey a research vision
  - Area of interest
  - Fundamental challenges in the field
  - Your approach to tackling those challenges
- Intellectual Merit
  - Why is this direction advancing the state-of-the-art in the field?
- Significance & Broader Impact
  - How are others outside your immediate field going to benefit?

Do's

- Have layers of specificity so that the reader can appreciate both the forest and (a few of) the trees.
- Intellectual merit => readers can appreciate the key technical details.
- Significance => argue that this challenge is fundamental and manifested in many problems.

Don't Do's

- Dive right into low-level technical detail without laying out the broader scope first
- Overload in unnecessary technical jargon.
- Only argue for a superficial connection – better to not include it at all.

Example Statement

Background Communication complexity seeks to understand complexity of computation by characterizing the amount of communication that must occur to compute the output of a function whose inputs are distributed among separate parties communicating on a broadcast channel. Aside from seeing only a portion of the input, players have unlimited computational resources. In the two-party variant, players Alice and Bob are given inputs x and y, respectively, each of size n. Their task is to jointly compute the output of some function $f(x,y)$ by following a protocol Π. At each step of the computation, the protocol dictates whose turn it is to send a message and what message they will send as a function of their input, previous messages sent, and perhaps shared and/or private randomness. In the multiparty ("number-on-forehead") variant, players 1 through t are assigned inputs $x_1,\dots,x_t$, again of size n, but now can see all inputs except their own. In both models, the cost of a protocol as a function of n is the worst-case number of bits that must be sent f. The communication cost of a function is the cost of the best protocol for f. See [5] for full definitions.

Concise overview of the area

Communication complexity is of some inherent interest, but more important is that bounds in the communication model are often amenable both to tractable proof and ready transfer to other models. In addition to applications where communication is obviously relevant (e.g. distributed algorithms), results have been transfered to e.g. space bounds for data stream algorithms, circuit size and depth lower bounds, time-space trade-offs for Turing machines, area-time tradeoffs for VLSI, and communication lower bounds for combinatorial auction algorithms.

Highlight significance

As in much of complexity, reductions between problems are common, so research tends to focus on a few particular functions. In particular, we will here often be concerned with the set-disjointness function DISJ, wherein players treat their inputs as characteristic functions of subsets of $\{1,\dots,n\}$ and must output 1 if and only if their sets are disjoint. We also often express DISJ in composed form as $\text{DISJ} = \text{OR} \circ \text{AND}$. DISJ plays a role for communication similar to that of SAT in much of complexity. In addition to its importance within communication complexity, lower bounds for two-party DISJ directly imply lower bounds for communication in combinatorial auctions, and are used in [1] to prove striking new barrier results. Lower bounds for multiparty DISJ directly imply exponential size lower bounds for a wide class of proof systems not presently accessible by any other methods, and work on multiparty disjointness lower bounds has also resulted in techniques that hold promise for separating NP from ACC^0, an important circuit class.

Set the stage for intellectual merit

Sam Hopkins

Example Statement

Background Communication complexity seeks to understand complexity of computation by characterizing the amount of communication that must occur to compute the output of a function whose inputs are distributed among separate parties communicating on a broadcast channel. Aside from seeing only a portion of the input, players have unlimited computational resources. In the two-party variant, players Alice and Bob are given inputs x and y, respectively, each of size n. Their task is to jointly compute the output of some function $f(x,y)$ by following a protocol Π. At each step of the computation, the protocol dictates whose turn it is to send a message and what message they will send as a function of their input, previous messages sent, and perhaps shared and/or private randomness. In the multiparty ("number-on-forehead") variant, players 1 through t are assigned inputs $x_1,\ldots,x_t$, again of size n, but now can see all inputs except their own. In both models, the cost of a protocol as a function of n is the worst-case number of bits that must be sent f. The communication cost of a function is the cost of the best protocol for f. See [5] for full definitions.

Concise overview of the area

Communication complexity is of some inherent interest, but more important is that bounds in the communication model are often amenable both to tractable proof and ready transfer to other models. … tributed algorithm … , circuit size and depth lower bounds, time-space trade-offs for Turing machines, area-time tradeoffs for VLSI, and communication lower bounds for combinatorial auction algorithms.

Opening sentence could be more direct in its focus

Highlight significance

As in much of complexity, reductions between problems are common, so research tends to focus on a few particular functions. In particular, we will here often be concerned with the set-disjointness function DISJ, wherein players treat their inputs as characteristic functions of subsets of $\{1,\ldots,n\}$ and must output 1 if and only if their sets are disjoint. We also often express DISJ in compose… f SAT in much of … bounds for two-party DISJ directly imply lower bounds for communication in combinatorial auctions, and are used in [1] to prove striking new barrier results. Lower bounds for multiparty DISJ directly imply exponential size lower bounds for a wide class of proof systems not presently accessible by any other methods, and work on multiparty disjointness lower bounds has also resulted in techniques that hold promise for separating NP from ACC^0, an important circuit class.

This paragraph could use a sentence on why this is OK.

Set the stage for intellectual merit

Sam Hopkins

Proposed Work The two-party complexity of DISJ is well-understood, but a deep understanding of its multiparty complexity remains an open problem. I propose a program of attacks in multiparty communication complexity, with connections to DISJ.

Summarize research philosophy

Protocols for Composed Functions As hinted above, finding a function in NP that requires $(\log n)^{\Omega(1)}$ communication for $(\log n)^{\Omega(1)}$ players would separate NP and ACC^0. [2] rules out a large class of candidate functions, including DISJ, by supplying efficient protocols for $t > 1 + \log n$ players for functions of the form $f \circ g$, where f satisfies a strong symmetry condition.

I intend to explore protocols for similar functions where we relax the symmetry condition on f. In [2], one class of such functions is proposed as a target for potential generalizations, providing a logical place to start an attack. Thus, we would begin by attempting to extend the protocols in [2]: I conjecture and intend to verify that there exist both nontrivial extensions of their protocols and perhaps also nontrivial reductions to them from less-symmetric functions. Along with improving our arsenal of communication protocols, a deeper understanding of the extent of the applicability of their techniques will aid in the search for hard functions to accomplish the separation of ACC^0 and NP.

Summarize key challenge.

Summarize key idea, which is relaxing symmetry condition

Relations Among Multiparty Lower-Bound Techniques Essentially all known lower-bound techniques in the two-player model are expressible as optima of particular linear programs. The authors of [4] introduce a new LP-based lower-bound technique and employ these LP characterizations to prove their technique optimal among almost all known techniques. Thus, (with the exception of new information-theoretic bounds), relationships among two-player bounds are well-understood.

Summarize prior work in this area.

On the other hand, recent work in [6] has vastly improved existing lower bounds on multiparty DISJ, but at the cost of significant technical complication. This and the success of the LP based project on the two-player side motivate my proposal to investigate LP formulations of and relationships among multiparty lower-bound techniques. The first question is whether existing multiparty lower bounds can be expressed as optima of LPs; an answer to this question (at least for simple techniques) seems well within reach. With such LPs available, we will be able to attempt both new proofs of relationships between existing methods and a search for new multiparty techniques by manipulating LP constraints. Additionally, LP formulations of lower bounds are likely deepen our understanding of the applications of such techniques to DISJ and related functions; indeed, some of the recent improvements in bounds for multiparty DISJ already rely on some LP and polynomial duality arguments (for example [3]), a hint that further work to formulate bounds in these terms may prove fruitful. A deepening of our understanding here may also result in progress towards finding candidate functions for the separation discussed above.

Describe new opportunities and angle of attack

Proposed Work The two-party complexity of DISJ is well-understood, but a deep understanding of its multiparty complexity remains an open problem. I propose a program of attacks in multiparty communication complexity, with connections to DISJ.

Opening sentence could be more direct in its focus, i.e., what is the high level challenge => why do we want to explore protocols for similar functions?

I intend to explore protocols for similar functions where we relax the symmetry condition on f. In [2], one class of such functions is proposed as a target for potential generalizations, providing a logical place to start an attack. Thus, we would begin by attempting to extend the protocols in [2]: I conjecture and intend to verify that there exist both nontrivial extensions of their protocols and perhaps also nontrivial reductions to them from less-symmetric functions. Along with improving our arsenal of communication protocols, a deeper understanding of the extent of the applicability of their techniques will aid in the search for hard functions to accomplish the separation of ACC^0 and NP.

This paragraph could more explicitly summarize the limitations of prior work.

new information-theoretic bounds), relationships among two-player bounds are well-understood.

On the other hand, recent work in [6] has vastly improved existing lower bounds on multiparty DISJ, but at the cost of significant technical complication. This and the success of the LP based project on the two-player side motivate my proposal to investigate LP formulations of and relationships among multiparty lower-bound techniques. The first question is whether existing multiparty lower bounds can be expressed as optima of LPs; an answer to this question (at least for simple techniques) seems well within reach. With such LPs available, we will be able to attempt both new proofs of relationships between existing methods and a search for new multiparty techniques by manipulating LP constraints. Additionally, LP formulations of lower bounds are likely deepen our understanding of the applications of such techniques to DISJ and related functions; indeed, some of the recent improvements in bounds for multiparty DISJ already rely on some LP and polynomial duality arguments (for example [3]), a hint that further work to formulate bounds in these terms may prove fruitful. A deepening of our understanding here may also result in progress towards finding candidate functions for the separation discussed above.

Summarize research philosophy

Summarize key challenge.

Summarize key idea, which is relaxing symmetry condition

Summarize prior work in this area.

Describe new opportunities and angle of attack

General Writing Tips

- Have first sentence of each paragraph summarize paragraph
  - One can just read the first sentence, and have a sense of what is being said
- Have layers of specificity that connects the forest to the trees
- Related to statements for faculty applications: https://yisongyue.medium.com/checklist-of-tips-for-computer-science-faculty-applications-9fd2480649cc