Hsin-Hao Su  
Research Statement

My research interests lie between theoretical computer science and distributed computing. They can be divided into three different categories:

1. **Distributed Graph Algorithms (a.k.a. Network Algorithms)**. The area studies how to solve problems efficiently in a network. Traditional studies of algorithms usually consider aggregating the data into a single machine and then computing the solution. Yet, in a network consisting of a vast amount of information such as the Internet, the cost of aggregating all the information can be very expensive. The area of distributed graph algorithms studies how a network solves problems by having each node gather a limited amount of information (or equivalently, having as few communication rounds as possible). In reality, systems such as Pregel by Google and Giraph by Apache are built to process large-scale network data for such a purpose.

2. **Biologically-Inspired Distributed Algorithms**. Most biological systems are similar to the distributed model of computation in the sense that they operate without a centralized coordinator. In fact, the nature itself is distributed by design. I am interested in modelling and understanding how local behaviors affect the global outcomes in biological systems.

3. **Algorithms for Classic Combinatorial Optimization Problems**. I am also interested in developing more efficient algorithms for the classic problems such as matchings, minimum cuts, and maximum flow.

In the following, I will describe my past and present research of each area in separate sections, followed by a future research section in the end.

## 1 Local Distributed Graph Algorithms

The area of local distributed algorithms studies how a network solves problems by having each node using only the information in its vicinity. Linial [Lin87, Lin92] formalized the LOCAL model, which then became the standard model in the locality literature. Briefly speaking, the model operates in synchronous rounds. In each round, each node can send a message to the neighbors, receive messages from the neighbors and then do some local computation. In the end of the algorithm, each node produces its own output. Thus, in a $k$-rounds algorithm, each node uses only the information in its $k$-neighborhood.

The coloring problem and its variants are among the most well-studied problems in the area. They have various applications such as job scheduling and contention resolution. For example, consider that in a wireless network where two adjacent nodes cannot broadcast at the same time. Protocols such as TDMA resolve this problem by assigning adjacent nodes with different time slots. Finding out the time slots corresponds to the coloring problem.

My thesis has been awarded the 2016 Principles of Distributed Computing Doctoral Dissertation Award for contributions to the fundamental problems in the area. In my opinion, one of the most important contribution is that it has paved the way for the separation between two of the most important problems of the area, the MIS problem and the $(\Delta + 1)$-coloring problem. In the followings, I will highlight the key contributions made by me and my co-authors to the area.

- **The first sub-logarithmic rounds algorithm for $(\Delta + 1)$-coloring**. The $(\Delta + 1)$-coloring problem is the most well-studied coloring problems, where $\Delta$ is maximum degree, since $(\Delta + 1)$ is the least amount of colors such that the coloring problem can be solved locally. In 1986, Luby [Lub86] as well as Alon, Babai, and Itai [ABI86] gave $O(\log n)$ rounds randomized algorithms for
the maximal independent set (MIS) problem and the $(\Delta + 1)$-coloring problem. Since then, there has been an extensive amount of studies on the $(\Delta + 1)$-coloring problem [CV86, GP87, GPS88, Lin92, SV93, PS96, Joh99, PR01, KW06, BEK14, Bar15, FHK16, BEPS16]. Interestingly, for general graphs, none of the algorithms above perform better than the long-standing $O(\log n)$ bound (although their algorithms are faster when the maximum degree is bounded). This year, we gave the first sub-logarithmic rounds algorithm [HSS16]. Our algorithm runs in $O(\sqrt{\log \Delta}) + 2^{O(\sqrt{\log \log n})}$ rounds, which is sub-logarithmic in $n$ for any value of $\Delta$.

Moreover, our algorithm implies a separation between two related fundamental problems, the $(\Delta + 1)$-coloring problem and the MIS problem. They are related in the sense that they are reducible to each other. In particular, a MIS algorithm can be used to solve the $(\Delta + 1)$-coloring problem with no additional costs. Conversely, after obtaining the $(\Delta + 1)$-coloring, the MIS can be computed in $O(\Delta)$ rounds by iterating through the color classes. Before our work, it was not clear whether the MIS problem is strictly harder than the $(\Delta + 1)$-coloring problem. With the $\Omega(\sqrt{\log n / \log \log n})$ lower bound of MIS for graphs with $\Delta = 2^{O(\sqrt{\log n \log \log n})}$ by Kuhn et al. [KMW16], they directly imply that the $(\Delta + 1)$-coloring problem is provably easier than the MIS problem.

Our result involves a new network decomposition method and a novel concentration analysis. Moreover, the algorithm for sparse subgraphs in the decomposition are based on our earlier result [EPS15], where we showed the separation between the $(2\Delta - 1)$-edge coloring and the maximal matching.

**The first deterministic poly-logarithmic rounds algorithm for $O(\Delta)$-edge coloring.** It is often thought that the polynomial time algorithms in the centralized world are analogous to the poly-logarithmic rounds algorithms in the distributed world. One of the biggest challenge in the area is to get a poly-logarithmic rounds deterministic algorithm for the MIS problem and the coloring problems. In a very recent paper [GS17], we developed the first poly-logarithmic deterministic algorithm for $O(\Delta)$-edge coloring —more precisely, $(2 + \epsilon)\Delta$—, partially settled a long-standing open problem since 1990’s. The result also yields an improvement on the best randomized algorithm for obtaining an $O(\Delta)$-edge coloring.

**New tools for efficient constructions of non-trivial colorings.** In many variants of the coloring problems, such as the frugal coloring problem, the defective coloring problem, the list coloring problem and the $k$-coloring problems with $k \leq \Delta$ for specific graph classes, the proofs showing the existence of the solution often employ one or more applications of Lovász Local Lemma (LLL). In [CPS14], we developed new tools for converting the LLL proofs into efficient distributed algorithms for obtaining the colorings. Using such a tool, we developed efficient distributed algorithms for obtaining $(4 + \epsilon)\Delta / \log \Delta$-colorings for triangle-free graphs and $(1 + \epsilon)\Delta$-edge-coloring for general graphs in two separate papers [PS15, EPS15]. Previously, efficient distributed algorithms were only known for both problems when the degrees are large, say $\Delta = \log^{\Omega(1)} n$. In the former, we also improved the best known chromatic number of triangle-free graphs, which was $(160 + \epsilon)\Delta / \log \Delta$. The distributed construction of LLL not only led to more efficient algorithms for the coloring problems, but also influenced the community to look at related problems in distributed computing, which turned out to be fruitful. For example, the new lower bound technique for LLL and related problems [BFH+16], as well as new insights in the role of randomness in distributed computing [CKP16].

The area has become very active. In fact, over the last decade, especially in the last two years, there has been an outbreak of significant results among the local distributed algorithms community. To name a few for example, the graph shattering technique [BEPS16], the new deterministic $(\Delta + 1)$-coloring algorithms [Bar15, FHK16], the new MIS algorithm [Gha16].
2 Distributed Algorithms Beyond Locality

The LOCAL model studies how each node computes the solution using the information in the neighborhood without limiting the amount of information. The CONGEST model is a more practical model which puts restrictions on the bandwidths of the links between the nodes. In the model, in each round, a node can only send a message consisting of $O(\log n)$ bits to each neighbor.

Traditional graph problems such as the minimum spanning tree problem, the shortest path problem, the minimum cut problem, and the maximum flow problem are considered to be the “global problems”, as it is not hard to see that any algorithm has to take at least diameter time, $D$, to compute the solution. The seminal paper of Das Sarma et al. [DSHK+11] shows that, in CONGEST model, $\tilde{\Omega}(D + \sqrt{n})$ rounds are necessary for these problems via reductions from communication complexity results.

Minimum cut approximation [NS14]. Danupon Nanongkai and I developed a $(1 + \epsilon)$-approximation algorithm for the minimum cut problem whose round complexity is $\tilde{O}(D + \sqrt{n})$, which matches the lower bound of Das Sarma et al. up to a poly-logarithmic factor. This allows the network to efficiently estimate the bottleneck of the congestion to a very precise order. The result uses the semi-duality relation between tree packings and cuts of Nash-Williams [NW61], which was introduced to the algorithmic domain for computing the minimum cuts by Karger [Kar00] and Thorup [Tho07].

Generalizing the Congested Clique [GKS]. In the last few years, there has been a rising number of studies on the problems in the Congested Clique model, which is a model that emphasizes on the amount of communication while putting aside the role of locality. The model is the same as the CONGEST model except that each pair of node can exchange a message of size $O(\log n)$ in each round (The input graph is therefore different from the underlying communication graph, where the latter is a clique). In an ongoing work with Mohsen Ghaffari and Fabian Kuhn, we show how to emulate the Congested Clique in the CONGEST model efficiently. That is, if one developed an algorithm in the Congested Clique model, it can then be converted to an algorithm in the CONGEST model with overheads that depend on the expansion of the graph.

3 Classic Combinatorial Optimization Problems

The matching problems are one of the classic problems in combinatorial optimization. As with many other classic problems, it is a challenge to make any progress in them, since they have been studied for a very long time. For example, the first weighted bipartite matching algorithms were given by Jacobi written in Latin over 150 years ago [Jac65]! However, any improvements on these problems could be substantial, since they are fundamental and they have many practical applications. For example, the matching problems are known to have applications in the scheduling of communication switches, task assignment, finding the tracking paths in a radar system etc. Moreover, many algorithms in combinatorial optimization employ matching as a subroutine, including Christofides’ algorithm for the travelling salesman problem, algorithms for Chinese Postman Problem, and algorithms for undirected single-source shortest paths. Improvements in matching algorithms directly lead to faster algorithms for these problems. The followings are my contributions with my PhD advisor Seth Pettie and his former student Ran Duan to the matching problems in the centralized setting:

- **A new algorithm for maximum weight perfect matching (MWPM) problem in general graphs [DPS17].** In 1991, Gabow and Tarjan [GT91] gave an algorithm for MWPM in general graphs that runs in $O(m\sqrt{\alpha(m,n)\log n \log(nW)})$ time, where $W$ is the largest edge weight. The notorious analysis and the cumbersome bound stood there until very recently, when we gave a new scaling algorithm that runs in $O(m\sqrt{\log(nW)})$ time. Previously, the bound was only known for MWPM in bipartite graphs [GT89, OA92].

- **A new algorithm for maximum weight matching (MWM) problem in bipartite graphs [DS12].** In the MWM problem, the matchings are not required to be perfect. Before our work,
the best known bounds for the MWM and MWPM along the scaling approach are both known to be $O(m\sqrt{n}\log(nW))$ for bipartite graphs \cite{GT89, OA92}. However, we show that the MWM problem in bipartite graphs can be solved in $O(m\sqrt{n}\log W)$ time, which improves the previous bounds over 20 years ago. The algorithm draws an intriguing connection between the primal method \cite{BG64} and the Dilworth’s Lemma.

4 Biologically-Inspired Distributed Algorithms

Besides the computing world, distributed processes happen everywhere in the nature. For example, the propagation of a neural network, the collective behavior of social insects, the flocking of birds, and the communication between cells. The area of biological distributed algorithm is a new interdisciplinary area that studies the distributed processes in the nature from a distributed computing perspective. There are two main purposes of the study.

- In biological systems, the underlying mechanism is usually very simple (e.g. simple communication and limited amount of memory). Yet, the question is how such a simple mechanism achieves complicated tasks. Understanding that could help us build new, efficient distributed systems with reduced costs.
- In biology research, many hypotheses are supported by the data from the experiments. By analyzing and understanding the underlying mechanism, we may justify these hypotheses in a mathematically rigorous way.

During my postdoc at MIT, I worked on the modelling of collective behaviors in insect colonies. We closely collaborate with an insect biologist, Anna Dornhaus, at University of Arizona.

- **The density estimation problem** It is well-known among the ant biologists that the ants adjust their behaviors according to the population density. In \cite{MSL16}, we provided a theoretical guarantee on how well the ants estimate the density on a 2D torus based on the encounter rate during their random walks. The results can also be generalized to other classes of graph as well. Interestingly, we discovered a parameter that determines how well such an approach could perform. Even in non-expanders graphs such as a 2D torus, the random walk-based sampling approach can perform nearly as well as if they were in a clique! We also extend our analysis to estimating the size of a social network, where the information can only collected by the agents moving on the network, since there is no master list of nodes to access. On some classes of graphs, our result is provably better on the number of probes than the previous known algorithms.

- **The task allocation problem** In task allocation problems, each task is associated with a demand. The question is how do the ants allocate themselves in a distributed way such that the demands are satisfied for all the tasks. Since it involves symmetry breaking, the problem is unsolvable unless either randomization is used or there are individual variations among the ants. In an on-going work \cite{RDL+}, we assume randomization is used and all ants are identical. We study how efficient can the ants (re)-allocate properly if the demands have been changed under different assumptions on the sensing capabilities of the ants. We also found that with extra number of ants, it is possible to facilitate the task allocation significantly, which might provide a possible explanation for the existence of idle ants. On the other hand, biologists proposed that each individual has different thresholds for the tasks. Recently, we started to investigate how individual variation facilitates task allocation without the power of randomization.

5 Future Research

The problems that I have examined so far is just the tip of an iceberg. There are many important open problems left in the areas. For example, there is still a large gap between the upper bound and
the lower bound for the MIS problem. Also, for the \((\Delta + 1)\)-coloring problem, there are very fast deterministic algorithms for bounded degree graphs, while randomized algorithms are faster when the degrees are large. Is it possible to bridge the gap between these two? Moreover, the deterministic poly-logarithmic rounds algorithm for MIS remains the ultimate open problem in the area. On the other hand, for problems that seem to be non-local such as the \((\Delta + 1)\)-edge-coloring problem, certainly there should be a lower bound for it. The list can go on and on. Instead of listing all of them, I would like to describe my general research paradigm in the following.

**Distributed and centralized–two birds with one stone:** I believe that studying the related problems in different models together is a very efficient way to understand the problems. For example, I think understanding the techniques in the centralized literature is a key to develop efficient distributed algorithms. In the distributed minimum cut paper [NS14], our algorithm borrowed the ideas from the centralized model and the dynamic graph model. In the end, it resulted in not only an almost-tight algorithm for the distributed setting, but also a much simpler centralized algorithm that nearly matches the best known bound. Another example is our deterministic edge coloring paper [GS17] borrows ideas of augmenting paths and blocking flows from matchings/flows in the centralized literatures. Therefore, studying in different models are certainly reciprocal.

Having understood the essential techniques for matching in the centralized model, next I plan to investigate the problem in the distributed model. Getting a \((1 - \epsilon)\)-approximate poly-logarithmic rounds maximum weight matching distributed algorithm for general graphs in the **CONGEST** model is one of the my next ambitious goals.

**Social Insect Behaviors and Mobile Agents Computing–two sides of the same coin:** The social insects and the mobile agents only differ slightly in the way that the latter may have a stronger computational power. Our density estimation paper is a perfect example of how these two areas stitch together. Our collaborator Anna Dornhaus and I have formulated a list of problems that are of common interest to both social insect biologists and theoretical computer scientists, including the soldier placement problem, the communication problems in foraging, and the transportation problem. In the future, I will explore the problems in both areas together.

**Expanding the research scopes:** The algorithms in the dynamic graph model and the local computation model usually share similar techniques with those in the distributed model. In the future, my goal is to also include these areas into my research scope via collaboration with the experts in the areas.

**References**


Mohsen Ghaffari, Fabian Kuhn, and Hsin-Hao Su. Generalizing the congested clique. manuscript.


Mira Radeva, Anna Dornhaus, Nancy Lynch, Radhika Nagpal, and Hsin-Hao Su. Cost of task allocation with local feedback: Effects of colony size and extra workers. manuscript.
