# Distributed Problem Solving in Geometrically-Structured Constraint Networks

# (Extended Abstract)

Roger Mailler
University of Tulsa
800 South Tucker Dr
Tulsa, Oklahoma
mailler@utulsa.edu

Huimin Zheng
University of Tulsa
800 South Tucker Dr
Tulsa, Oklahoma
huimin-zheng@utulsa.com

## ABSTRACT

Distributed Constraint Satisfaction (DisCSP) is a popular formalism that is used for developing a wide variety of general-purpose protocols. With very few exceptions, these protocol are tested using completely random instances with the understanding that this leads to better overall solutions. In many real-world situations, however, the variables in the problem represent objects that exist in $n$-dimensional space with constraints between them based on distance. In such instances, the constraint network forms a geometric graph and therefore is referred to as a Geometrically-Structured Constraint Satisfaction Problem (GS-CSP).

This paper introduces the GS-CSP and evaluates the performance of two complete DisCSP protocols to demonstrate how the introduction of structure affects these general problem solving approaches. Our findings show that GS-CSPs possess unique characteristics particularly in the phase transition regions and these characteristics can have a dramatic impact on the performance of current DisCSP algorithms.

## Categories and Subject Descriptors

I.2.11 [**Artificial Intelligence**]: Distributed Artificial Intelligence—*Multiagent systems*

## General Terms

Algorithms, Performance, Experimentation

## Keywords

Distributed, Constraints, Structured Graphs

## 1. INTRODUCTION

Since the early 1990s the Distributed Constraint Satisfaction Problem (DisCSP) [5] has been used to model a significant number of real-world problems. For example, DisCSP protocols have been used for controlling sensor networks [1] and robot routing [6]. Yet, neither of these domains has random structure.

**Appears in:** *Proceedings of the 12th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2013)*, Ito, Jonker, Gini, and Shehory (eds.), May, 6–10, 2013, Saint Paul, Minnesota, USA.


This paper introduces the geometrically-structure constraint satisfaction problem (GS-CSP), which is formed when variables are located in $n$-dimensional space and are connected together by random binary constraints when their distance is less than a given threshold. Section 2 gives a formal definition of the distributed GS-CSP. Section 3 then describes a set of experiments that were conducted using two complete DisCSP solvers, APO [2] and AFC-CBJ [3]. Finally, in section 4 we present our conclusions and future work.

## 2. THE DISTRIBUTED GEOMETRICALLY-STRUCTURED CSP

A geometric graph is a tuple $G = \langle X, r\rangle$ where $X \subset \mathbb{R}^n$ is a set of vertices located in $n$-dimensional space with undirected edges connecting all pairs $\{x \in X, y \in Y\}$ with $||y - x|| \leq r$ [4]. We can expand on the definition of the geometric graph to formally describe the Geometrically-Structured Constraint Satisfaction Problem (GS-CSP) $P = \langle V, D, f\rangle$, which consists of a set of $n$ variables $V = \{v_1, \ldots, v_n\}$ with each variable $v_i \in \mathbb{R}^n$, discrete, finite domains for each variable $D = \{D_1, \ldots, D_n\}$, and a set of binary constraints $c = \{c_1, \ldots, c_m\}$ where each $c_i(d_{i,x}, d_{i,y})$ is a function $c_i : D_{i,x} \times D_{i,y} \rightarrow \{true, false\}$ and $||y - x|| \leq r$. The task is to find an assignment $S = \{d_1, \ldots, d_n | d_i \in D_i\}$ such that all of the constraints are satisfied or to report that no such solution exists.

For a number of reasons, it is natural to think about objects that are located in an $n$-dimensional space as agents. So it is equally natural to extend the definition of the GS-CSP to the distributed GS-CSP. In the distributed version of the problem, each agent has a spacial location and manages one or more variables. The relationships between the variables are again dictated by a radius or range, however in the multi-variable per agent case, each variable may have a unique range associated with it. For example, imagine a robot that is coordinating its sensing activities to provide coverage of warehouse while at the same time must coordinate its use of a share radio frequency.

## 3. EVALUATION

To evaluate the effect of structure on distributed protocols, we chose to use two very different DisCSP algorithms. Namely, the AFC-CBJ protocol, which is a tree-based backtracking protocol and APO, which is a partial centralizing mediation based protocol. Following standard practice, we used 20 variable problems with a domain size of 10. We used

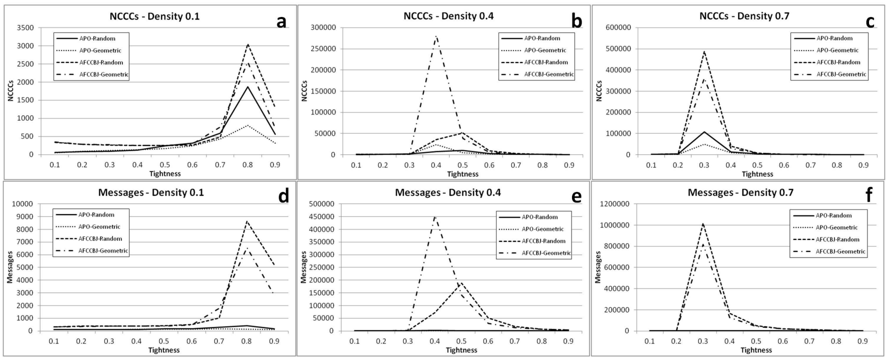


**Figure 1: A comparison of AFC-CBJ and APO on 20 variable random and geometrically-structured DisCSPs.**

values for $p_1$ of 0.1, 0.4, and 0.7 and values for $p_2$ that varied from 0.1 to 0.9. We conducted experiments on both random and geometrically-structured instances and collected 30 samples per data point. Both algorithms were given the exact same problem instances with the same initial variable assignments. We used a cycle based simulator where during each cycle, messages were delivered to the agents, they were allowed to process them, and then queue up messages for delivery at the beginning of the next cycle. During the runs, we counted the number of messages sent and the number of NCCCs used by each protocol.

The results of these experiments can be seen in Figure 1. We see that on low and high density problems, that distributed GS-CSPs on average are easier to solve than random DisCSP instances. This trend reverses for the medium density problems, where it is clear that for both APO and AFC-CBJ that a shifted phase transition has a meaningful effect. This has a particularly profound effect on AFC-CBJ.

Another interesting trend that is noteworthy is that the most recent implementation of APO outperforms AFC-CBJ on all instances for both metrics. In the best case, we found that it used 20X fewer NCCCs than AFC-CBJ. We should note that for time considerations, we were forced to stop some of the runs for AFC-CBJ at 250,000 cycles. This only affected the $p_1 = 0.4, p_2 = 0.4$ results by making them appear somewhat better than the actual values we would obtain if they were allowed to run to completion.

# 4. CONCLUSIONS AND FUTURE WORK

This paper introduces an important subset of the classical CSP formulation: the geometrically-structured CSP. The GS-CSP is based on the recognition that many real-world problems occur in $n$-dimensional space and that constraints in these domain are often based on distance. These problems can be represented as geometric graphs, which possess a unique set of properties. By exploring these problems, we have discovered that they are characteristically easier to solve when the density of their constraints are either fairly low or fairly high. However, for medium density problems, they become more difficult to solve than their random counterparts.

Many of these discoveries can be explained by examining the clustering properties of geometric graphs as the edge density increases. GS-CSPs form tightly coupled clusters at low densities, yet remain fairly disconnected overall when compared to random instances. There is considerable work that still remains to be done to fully understand the implications of geometric structure on constraint networks. For example, the experiments in this paper were done using a small number of variables with large domains. This choice was made in part to follow convention. However, we found that using a larger numbers of variables was impractical due to the run times of AFC-CBJ. We believe it is important to look at larger problems, potentially sacrificing the size of the variable's domains, in order to truly understand the consequences that structure has on problem solving complexity and solution optimality.

# 5. ACKNOWLEDGMENTS

This material is based on research sponsored by the Air Force Research Laboratory, under agreement number FA8750-11-2-0066. The U.S. Government is authorized to reproduce and distribute reprints for Governmental purposes notwithstanding any copyright notation thereon. The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies or endorsements, either expressed or implied, of the Air Force Research Laboratory or the U.S. Government.

# 6. REFERENCES

[1] J. Kho, L. Tran-Thanh, A. Rogers, and N. R. Jennings. An agent-based distributed coordination mechanism for wireless visual sensor nodes using dynamic programming. *The Computer Journal*, 53(8):1277–1290, 2010.

[2] R. Mailler. Improving asynchronous partial overlay. In *Proceeding of IAT 2012*, 2012.

[3] A. Meisels and R. Zivan. Asynchronous forward-checking for DisCSPs. *Constraints*, 12:131–150, 2007.

[4] M. Penrose. *Random Geometric Graphs*. Oxford Studies in Probability. Oxford University Press, New York, 2003.

[5] M. Yokoo, E. H. Durfee, T. Ishida, and K. Kuwabara. Distributed constraint satisfaction for formalizing distributed problem solving. In *International Conference on Distributed Computing Systems*, pages 614–621, 1992.

[6] X. Zheng and S. Koenig. Reaction functions for task allocation to cooperative agents. In *7th International Conference on Autonomous Agents and Multi-agent Systems*, 2008.