

# Distributed CSPs: Why it is assumed a Variable per Agent?\*

Miguel A. Salido

DSIC, Technical University of Valencia  
Camino de Vera s/n, 46071, Valencia, Spain  
msalido@dsic.upv.es

**Abstract.** Nowadays, many real problems can be formalized as Distributed CSPs. A distributed constraint satisfaction problem (DisCSP) is a CSP in which variables and constraints are distributed among multiple automated agents. Many researchers assume for simplicity that each agent has exactly one variable. For real distributed problem these techniques require a large amount of messages passed among agents, so these problems are very difficult to solve. In this research summary, we question why the lack of works to manage large-scale problems.

**keywords:** Distributed Constraint Satisfaction Problems.

## 1 Introduction

In recent years we have seen an increasing interest in Distributed Constraint Satisfaction Problem (DisCSP) formulations to model combinatorial problems (see the special issue on Distributed Constraint Satisfaction in Artificial Intelligence journal, vol 161, 2005). There is a rich set of real-world distributed applications, such as networked systems, planning, scheduling, etc, for which the DisCSP paradigm is particularly useful. In such distributed applications, privacy issues, knowledge transfer costs, robustness against failure, etc preclude the adoption of a centralized approach [3].

A distributed CSP is a CSP in which the variables and constraints are distributed among automated agents. Finding a value assignment to variables that satisfies inter-agent constraints can be viewed as achieving coherence or consistency among agents.

The more cited papers related to DisCSP make the following assumptions for simplicity in describing the algorithms:

1. Each agent has exactly one variable.
2. All constraints are binary.
3. Each agent knows all constraint predicates relevant to its variable.

---

\* This work has been partially supported by the research projects TIN2004-06354-C02-01 (Min. de Educacion y Ciencia, Spain-FEDER) and GV/2007/274 (G. Valenciana).

Although the great majority of real problems are naturally modelled as non-binary CSPs, the second assumption is comprehensible due to there exist some techniques that translate any non-binary CSP into a equivalent binary one [1].

However, the first assumption is too restrictive and the main basic research is focused to small instances and little work has been done to solve real-life problems.

## 2 From Basic Research Toward Applied Research

One of the pioneer researchers in DisCSP said "So far, we assume that each agent has only one local variable. Although the developed algorithms can be applied to the situation where one agent has multiple local variables by the following methods, both methods are neither efficient nor scalable to large problems" [5].

- Method 1: each agent finds all solutions to its local problem first. By finding all solutions, the given problem can be re-formalized as a distributed CSP, in which each agent has one local variable whose domain is a set of obtained local solutions. Then, agents can apply algorithms for the case of a single local variable. The drawback of this method is that when a local problem becomes large and complex, finding all the solutions of a local problem becomes virtually impossible.
- Method 2: an agent creates multiple virtual agents, each of which corresponds to one local variable, and simulates the activities of these virtual agents. However, since communicating with other agents is usually more expensive than performing local computations, it is wasteful to simulate the activities of multiple virtual agents without distinguishing the communications between virtual agents within a single real agent, and the communications between real agents.

In spite of significant progress in distributed CSP, the following question is straightforward: *Why it is assumed a Variable per Agent?*

Only some works include a set of variables into an agent [4],[2]. Therefore, more research must be done to solve more realistic problems.

## References

1. F. Bacchus and P. van Beek, 'On the conversion between non-binary and binary constraint satisfaction problems', *In proceeding of AAAI-98*, 311–318, (1998).
2. Bessire C. Belaïssaoui M. Ezzahir, R. and El-H. Bouyakhf, 'Dischoco: A platform for distributed constraint programming', *In Proceedings of IJCAI-07 Workshop on Distributed Constraint Reasoning*, (2007).
3. B. Faltings and M. Yokoo, 'Introduction: Special issue on distributed constraint satisfaction', *Artificial Intelligence*, **161**, 1–5, (2005).
4. M.A. Salido and F. Barber, 'Distributed CSPs by graph partitioning', *Applied Mathematics and Computation*, **183**, 491–498, (2006).
5. M. Yokoo and K. Hirayama, 'Algorithms for distributed constraint satisfaction: A review', *Autonomous Agents and Multi-Agent Systems*, **3**, 185–207, (2000).