Distributed Constraint Optimisation

A Distributed Constraint Optimisation Problem (DCOP) consists of a set of agents, where each agent has a set of variables. Each variable has a corresponding domain of values. Every set of assignments of values to variables is a valid solution – for each agent with m variables, with average domain size d, we require $d^m$ space for each agent and we have a total problem solution space of size $d^{n m}$. We test using random binary graph colouring problems. IMP2 is the improved compilation and SNI sets. Interchangeability in DCOP

Consider an agent in a distributed 2-colouring problem with multiple local variables, three of which are external. The variables a, b are linked to agents A and B, who have a constraint between them. Therefore an assignment to variable b indirectly affects variable a. So, we calculate SNI sets with respect to both agents. Solutions {1,2,3,4} and {5,6,8} are sub-neighbourhood interchangeable (SNI) w.r.t. agent C.

Improve Compilation

1. Find one optimal solution for each assignment to the external variables – if we assume n agents, each with m variables, with average domain size d, for which p variables are private, we require $d^{m-p}$ space for each agent and we have a total problem solution space of size $d^{n m-p}$, which is a reduction by a factor of $d^p$.

Interchangeability in DCOP

Sub-neighbourhood interchangeability: Two values x and y for a variable V are sub-neighbourhood interchangeable (SNI) with respect to a subset S of the neighbours of V, if and only if for every constraint C between V and variables in S, x and y satisfy C for identical sets of assignments to the other variables in C.

Consider variable c, which is an external variable linked to agent C. The original values 0 and 1 are represented 4 times each. Solutions {1,3,5,7} and {2,4,6,8} are sub-neighbourhood interchangeable (SNI) w.r.t. agent C.

Experiments

We compare the original compilation method BASIC, with our improved method IMP1, and also IMP2, which uses both the improved compilation and SNI sets. We test using random binary graph colouring problems, with parameter settings $<a,e,v,i,d>$. We then compare the algorithms varying v, i, d and in turn, averaging over 20 test instances.

For many parameter settings (with complex local problems), the improved compilation offers orders of magnitude savings over the basic compilation. Furthermore, using the SNI sets that occur in the compilation reduces the computation required in the distributed search by on average 45%.

Time taken to compute agents.

Non-Concurrent Constraint Checks (cutoff = 10,000,000).