Dynamic Detection and Elimination of Local Symmetry in CSPs

Belaïd Benhamou and Mohamed Réda Saïdi

Université de Provence, CMI Laboratoire des Sciences de L'Information et des Systèmes **(Équipe INCA, LSIS)**

SymCon'07 - Providence, USA - Sunday, 23rd September 2007

Motivation

- *Local symmetries* are the symmetries of the resulting CSP at a node of the search tree corresponding to a partial instantiation.
- *Global symmetries* are the local symmetries of the CSP corresponding to the root node of the seach tree.
- Detection of local symmetry is, in general, a hard task.
- Many research works on symmetry in CSPs appeared recently, but, most of them deal only with the global symmetry.

Outline

- Background on CSPs.
- Semantic symmetry in CSPs
- Syntactic symmetry in CSPs.
- Syntactic symmetry and semantic symmetry
- The weakened syntactic symmetry conditions.
- Dynamic detection of local symmetry.
- Exploitation of local symmetry.
- Experiments.
- Conclusion.

CSP background: Definition

- **Definition:** A CSP is a quadruple (V, D, C, R) where :
  - $V = \{v_1, ..., v_n\}$ is a set of n variables.
  - $D = \{D_1, ..., D_n\}$ is the set of finite discrete domains associated to the CSP variables.
  - $C = \{C_1, ..., C_m\}$ is a set of m constraints. Each constraint involves a subset of the CSP variables.
  - $R = \{R_1, ..., R_m\}$ is a set of relations corresponding to the constraints of C, R_i represents the list of value tuples permitted by the constraint C_i.
- A binary constraint is a constraint which involves at most two variables.
- A binary CSP is a CSP involving only binary constraints.

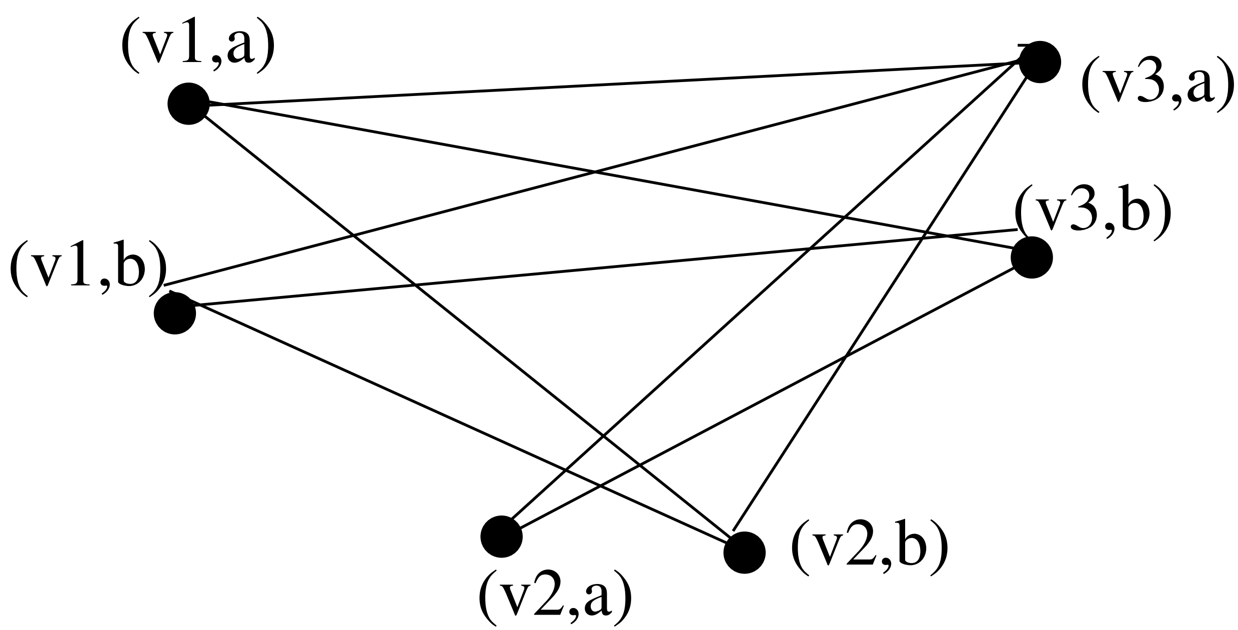
CSP Background: the Micro-structure

- **Definition:**

  The microstructure ([FREUDER 91], [JEGOU 93]) of a CSP $\mathcal{P}$ is a graph $\mathcal{M}_{\mathcal{P}}(V \times \cup_{i \in [1,n]} D_i, \acute{E})$, where each edge of $\acute{E}$ corresponds either to a tuple allowed by a specific constraint or to an allowed tuple because there is no constraints between the associated variables.

Example

Consider the CSP $P = (V, D, C, R)$ where V={v_1, v_2, v_3},
D={D_1, D_2, D_3} with ($D_i = \{a, b\}$ for $i \in \{1, 2, 3\}$) , C={C_{12}, C_{23}} and
R={R_{12}, R_{23}} such that $R_{12} = \{(a, b), (b, b)\}$
and $R_{23} = \{(a, a), (a, b), (b, a)\}$
The microstructure of the CSP is the following:



CSP background: Instantiations

- An instantiation $\mathcal{I} = (\langle v_1, a_1 \rangle, \langle v_2, a_2 \rangle, \ldots, \langle v_n, a_n \rangle)$ is the variable assignment $\{v_1 = a_1, v_2 = a_2, \ldots, v_n = a_n\}$ where a value a_i of the domain D_i is assigned to the variable v_i .
- The instantiation $\mathcal{I}$ is consistent if it satisfies all the constraints of C, thus $\mathcal{I}$ is a solution of the CSP.
- An instantiation of a subset of the CSP variables V is called a partial instantiation, it defines a nogood when it is inconsistent.
- Each partial instantiation $\mathcal{I}$ defines a node $n_{\mathcal{I}}$ in the search tree which corresponds to the local CSP $\mathcal{P}_{\mathcal{I}}$ resulting from $\mathcal{P}$ by considering $\mathcal{I}$ and its induced propagations.

Semantic Symmetry

- **Definition:**[Semantic symmetry for consistency]
  Two variable-value pairs $\langle v_i, b_i \rangle \in V \times D_i$ and $\langle v_j, c_j \rangle \in V \times D_j$ are symmetrical for consistency iff the following assertions are equivalent:
  1. There is a solution of the CSP which assigns the value b_i to the variable v_i;
  2. There is a solution of the CSP which assigns the value c_j to the variable v_j.
- **Definition:**[Semantic symmetry for all solutions]
  Two variable-value pairs $\langle v_i, b_i \rangle \in V \times D_i$ and $\langle v_j, c_j \rangle \in V \times D_j$ are symmetrical for $sol(\mathcal{P})$ if and only if each solution of the CSP assigning the value b_i to v_i can be mapped into a solution assigning the value c_j to v_j and vice-versa.

Syntactic Symmetry

- **Definition:** A syntactical symmetry of a CSP $\mathcal{P} = (V, D, C, R)$ having the microstructure $\mathcal{M}_{\mathcal{P}}$, is a mapping $\sigma : V \times \cup_{i \in [1,n]} D_i \longrightarrow V \times \cup_{i \in [1,n]} D_i$, that preserves the edges and the non-edges of $\mathcal{M}_{\mathcal{P}}$.
- **Definition:** Let $\mathcal{I}$ be a partial instantiation of the CSP $\mathcal{P}$ corresponding to the node $n_{\mathcal{I}}$. A syntactic symmetry of $\mathcal{P}_{\mathcal{I}}$ is a local symmetry of the CSP $\mathcal{P}$.
- **Definition:** A syntactic symmetry of the CSP $\mathcal{P}$ is a global symmetry.
- **Remark:** A global symmetry is a local symmetry of the CSP corresponding to the root node.

Syntactic Symmetry / Semantic symmetry

- **Definition:**

  Two variable-value pairs $\langle v_i, b_i \rangle \in V \times D_i$ and $\langle v_j, c_j \rangle \in V \times D_j$ are syntactically symmetrical in $\mathcal{P}$ iff there exists a symmetry σ of $\mathcal{P}$ such that $\sigma(\langle v_i, b_i \rangle) = \langle v_j, c_j \rangle$

- **Theorem:**

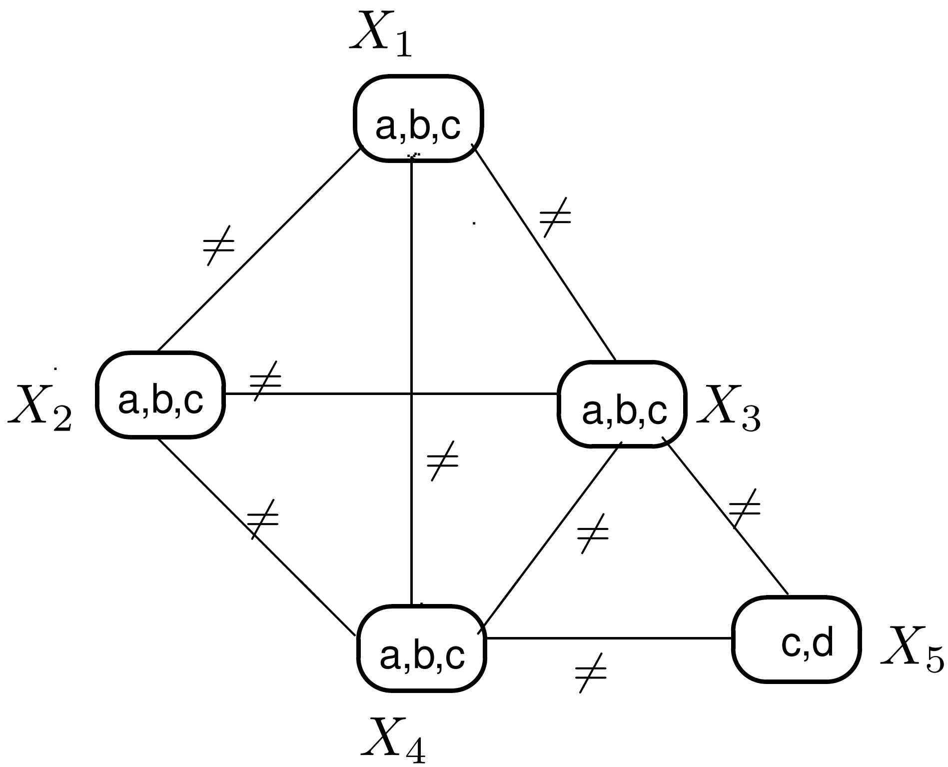
  If two variable-value pairs $\langle v_i, b_i \rangle \in V \times D_i$ and $\langle v_j, c_j \rangle \in V \times D_j$ are syntactically symmetrical, then they are semantically symmetrical for all solutions of the CSP.

The weakened syntactic symmetry 1/6

- **Definition: (Assignment tree)**

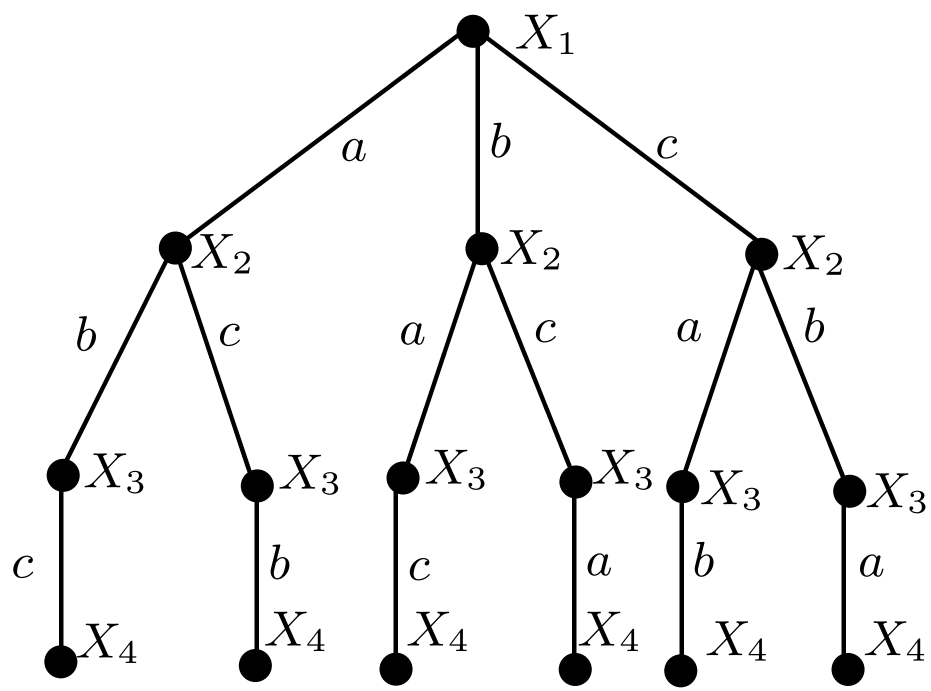
  We call an assignment tree of a CSP $\mathcal{P}$ corresponding to a given search method and a fixed variable ordering, a tree which gathers the history of all the variable assignments made during its consistency proof, where the nodes represent the variables of the CSP and where the edges out coming from a node X_i are labeled by the different values used to instantiate the corresponding CSP variable X_i.

The weakened syntactic symmetry 2/6



Example of graph coloring problem.

The weakened syntactic symmetry 3/6



- Assignment tree of Forward Checking process applied to the problem w.r.t the variable ordering $\{X_1, X_2, X_3, X_4, X_5\}$.

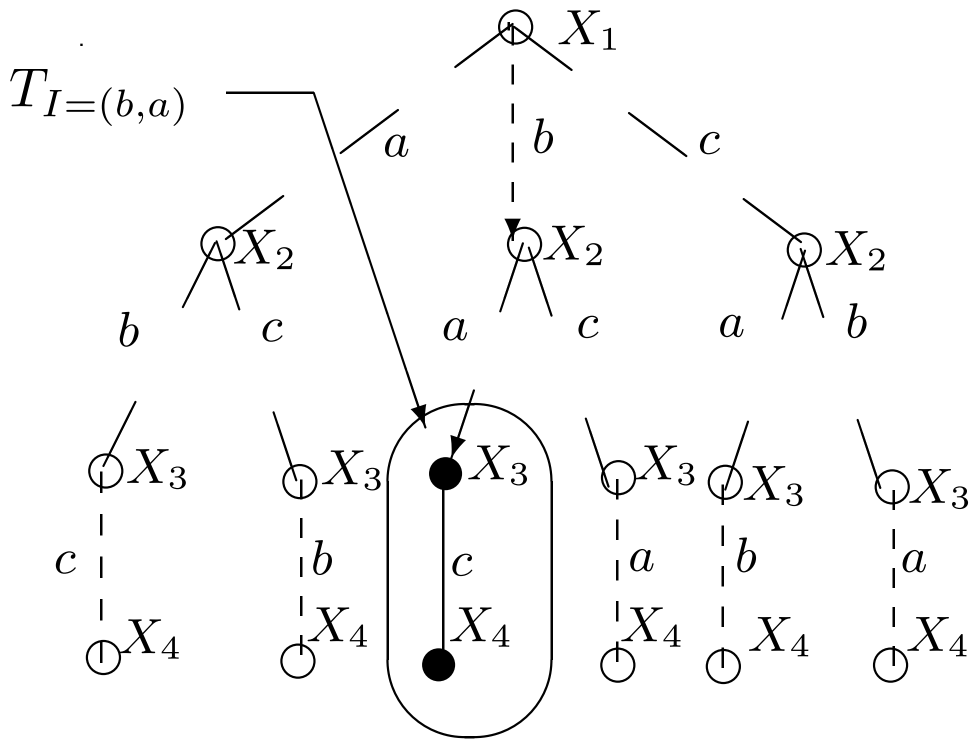
The weakened syntactic symmetry 4/6

- **Definition: (Failure tree)**

  Let T be an assignment tree corresponding to a consistency proof of a CSP $\mathcal{P}$, $I = (a_1, a_2, ..., a_i)$ an inconsistent partial instantiation of the variables X_1,X_2,...,X_i corresponding to the path $\{X_1, X_2, ..., X_i\}$ in T. We call a failure tree of the instantiation I, the sub-tree of T noted by $T_{I=(a_1,a_2,...,a_i)}$ such that:

  1. The root of the tree T and the root of the sub-tree $T_{I=(a_1,a_2,...,a_i)}$ are joined by the path corresponding to the instantiation I;
  2. All the CSP variables corresponding to the leaf nodes of $T_{I=(a_1,a_2,...,a_i)}$ have empty domains.

The weakened syntactic symmetry 5/6



- Failure tree $T_{I=(b,a)}$ of the partial instantiation $I = (b, a)$.

The weakened syntactic symmetry conditions 6/6

- **Theorem:** Let $\mathcal{P}(V, C, D, R)$ be a CSP, $\mathcal{I}_0 = (\langle v_1, a_1\rangle, \ldots, \langle v_{i-1}, a_{i-1}\rangle)$ a partial instantiation of $i-1$ variables instantiated before the current variable v_i such that the extension $\mathcal{I} = \mathcal{I}_0 \bigcup \{\langle v_i, a_i\rangle\}$ is inconsistent, $T_\mathcal{I}$ is the failure tree of $\mathcal{I}$ and $Var(T_\mathcal{I})$ the set of the variables corresponding to the nodes of $T_\mathcal{I}$. If $\langle v_i, a_i\rangle$ is syntactically symmetrical to $\langle v_j, b_j\rangle$ in the CSP $\acute{\mathcal{P}}_{\mathcal{I}_0}$ derived from $\mathcal{P}_{\mathcal{I}_0}$ by restricting its set of variables to $Var(T_\mathcal{I}) \cup \{v_i\}$, then the extension $\mathcal{J} = \mathcal{I}_0 \bigcup \{\langle v_j, b_j\rangle\}$ is inconsistent.
- **Remark:** here syntactic symmetry conditions are restricted to only the variables of the failure tree.

Detection 1/3

- A syntactical symmetry *Constraint Symmetries* [COHEN et Al 05]) of a CSP $\mathcal{P}$ is an automorphism of its microstructure $\mathcal{M}_{\mathcal{P}}$.
- The set of syntactic symmetries of a CSP $\mathcal{P}$ is identical to the automorphism group $Aut(\mathcal{M}_{\mathcal{P}})$ of its microstructure.
- We use tools like Saucy or Nauty to compute graph automorphismes.
- The idea is to maintain dynamically the microstructure $\mathcal{M}_{\mathcal{P}_{\mathcal{I}}}$ of the CSP $\mathcal{P}_{\mathcal{I}}$ corresponding to the local sub-problem defined at each current node $n_{\mathcal{I}}$.
- then color the microstructure $\mathcal{M}_{\mathcal{P}_{\mathcal{I}}}$ and compute its automorphism group $Aut(\mathcal{M}_{\mathcal{P}_{\mathcal{I}}})$.
- The CSP $\mathcal{P}_{\mathcal{I}}$ can be viewed as a new problem corresponding to the unsolved part of $\mathcal{P}$.

Detection 2/3

- **Full symmetry detection**
  - **The one-color-strategy**: Here, all the nodes of the microstructure $\mathcal{M}_{\mathcal{P}_{\mathcal{I}}}$ are assigned the same color, then when applying Saucy a set of generators Gen representing the full local symmetry group is returned. This approach allows to detect all the syntactical variable-value symmetries of the CSP $\mathcal{P}_{\mathcal{I}}$.

Detection 3/3

- **Limited symmetry detection**
  - **The multi-colors-strategy**: A first compromise is to limit permutations to only values of the same domains. A color is associated to each variable. Every node of the microstructure belonging to a variable is colored with the same color.
  - **The two-colors-strategy**: A second compromise is to associate to the current variable v_i (under instantiation) one color and all the other variables another color. All the nodes of the microstructure belonging to the current variable v_i have the first color and all the other nodes the second one.

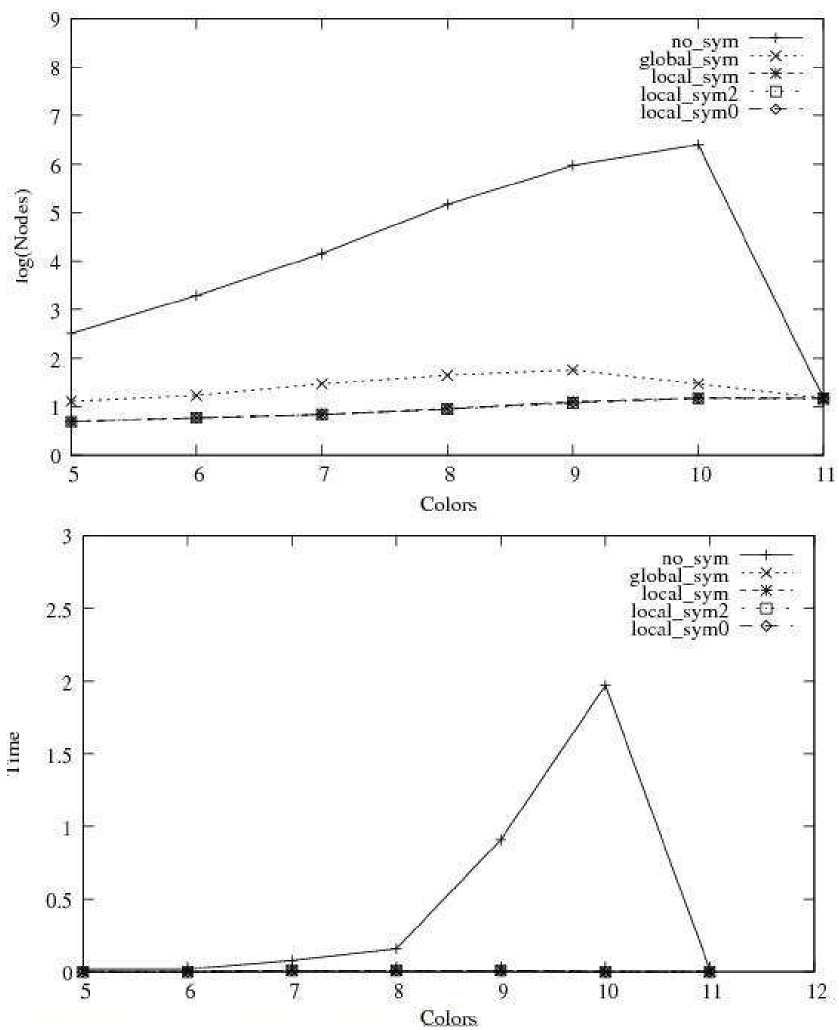
Exploitation

- During search, the domain D_i of the curent variable v_i is partitioned into value sub-sets that are locally symmetrical.
- To avoid generating locally symmetrical solutions, we consider one value from each subset of symmetrical values in D_i.
- If $\mathcal{I}$ is an inconsistent partial instantiation in which the assignment $\langle v_i, c_i \rangle$ of the current variable v_i is shown to participate in no solutions of the CSP $\mathcal{P}$, then all the pairs $\langle v_j, d_j \rangle$ which are symmetrical to $\langle v_i, c_i \rangle$ in $\mathcal{P}_{\mathcal{I}}$ do not. Thus we remove d_j from the domain of v_j, and prune the sub-space which corresponds to its assignment to v_j.

The experimented Methodes

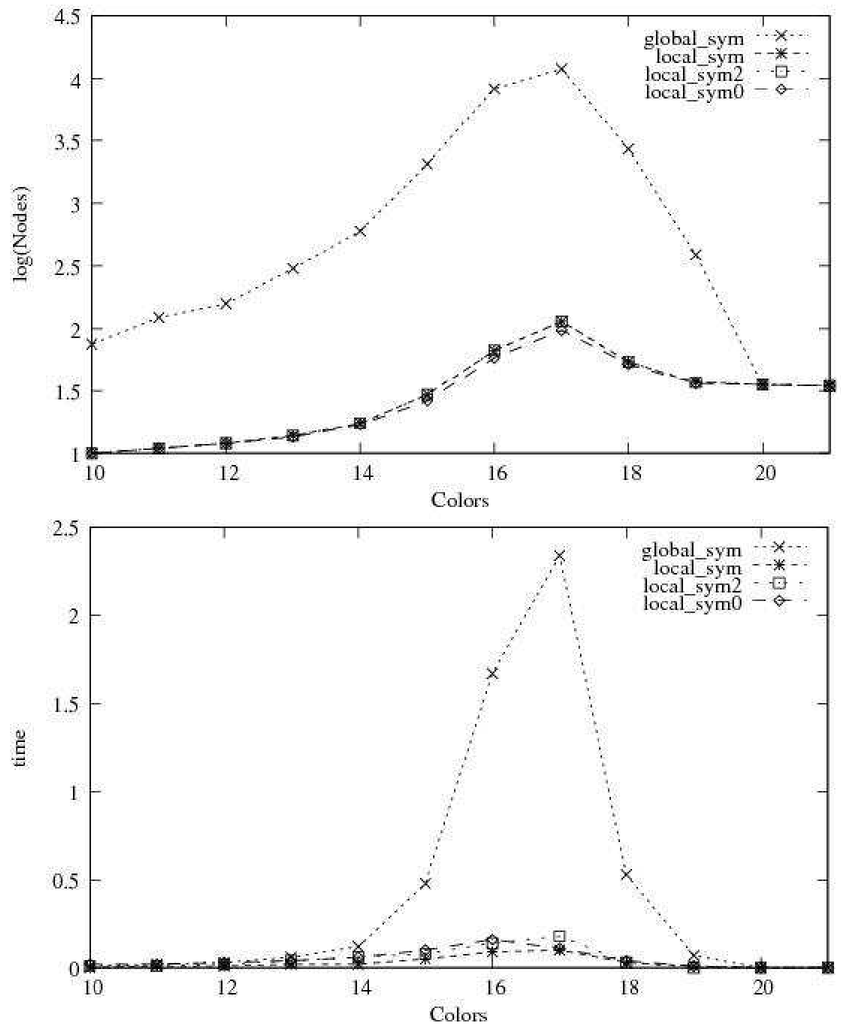
- **No-sym:** search without symmetry breaking;
- **Global-sym:** search with global symmetry breaking restricted to values of a same domain.
- **Local-sym0:** search with full local symmetry detection and elimination. This method implements the one-color strategy.
- **Local-sym1:** search with local value symmetry breaking (the weakened symmetry). This method implements the multi-colors strategy.
- **Local-sym2:** search with restricted local variable-value symmetry breaking (the weakened symmetry). This method implements the two-colors strategy.

Experiments: Random graph coloring problems (1/2)



Node and Time curves where $n = 15$ and $d = 0.9$.

Experiments: Random graph coloring problems (1/2)



Node and Time curves where $n = 35$ and $d = 0.9$.

Experiments: Dimacs graph coloring benchmarks 1/3

		No-sym		Global-sym		Local-sym0	
Instance	k	*Nodes*	*Time*	*Nodes*	*Time*	*Nodes*	*Times*
myciel5	6	-	-	8,040,259	59.84	1,516,699	16.87
anna	11	-	-	3,403	0.59	151	0.03
david	11	-	-	3,896	0.23	106	0.0
queen7_7	7	2,452	0.01	513	0.02	502	0.0
queen8_8	9	-	-	10,629,131	262.54	859,087	23.76

		Local-sym1		Local-sym2		Local-sym0	
Instance	k	*Nodes*	*Time*	*Nodes*	*Time*	*Nodes*	*Times*
myciel5	6	2,413,556	22.21	2,406,945	25.36	1,516,699	16.87
anna	11	168	0.05	168	0.08	151	0.03
david	11	124	0.03	124	0.03	106	0.0
queen7_7	7	502	0.01	502	0.0	502	0.0
queen8_8	9	1,399,436	29.7	1,396,774	30.16	859,087	23.76

Experiments: Dimacs graph coloring benchmarks 2/3

Instance	*k*	No-sym *Nodes*	No-sym *Time*	Global-sym *Nodes*	Global-sym *Time*	Local-sym0 *Nodes*	Local-sym0 *Times*
school1	14	-	-	-	-	45,183	11.82
school1_nsh	14	-	-	-	-	892,372	174.76
2-Insertion_3	4	832,150	1.02	277,408	0.73	19,646	0.07
2-FullIns_3	5	2,294,396	7.63	193,347	1.14	25,049	0.4
mugg88_25	4	-	-	-	-	440,676	5.62

Instance	*k*	Local-sym1 *Nodes*	Local-sym1 *Time*	Local-sym2 *Nodes*	Local-sym2 *Time*	Local-sym0 *Nodes*	Local-sym0 *Times*
school1	14	76,192	17.28	75,985	17.85	45,183	11.82
school1_nsh	14	1,487,287	257.57	1,486,523	270.4	892,372	174.76
2-Insertion_3	4	135,953	0.48	115,737	0.52	19,646	0.07
2-FullIns_3	5	49,202	0.59	48,076	0.65	25,049	0.4
mugg88_25	4	881,784	6.74	881,784	9.4	440,676	5.62

Experiments: Dimacs graph coloring benchmarks 3/3

		No-sym		Global-sym		Local-sym0	
Instance	k	*Nodes*	*Time*	*Nodes*	*Time*	*Nodes*	*Times*
zeroin.i.1	49	-	-	-	-	268	36.8
zeroin.i.2	30	-	-	-	-	262	7.22
mulsol.i.2	31	-	-	-	-	237	11.24
mulsol.i.3	31	-	-	-	-	237	11.1
le450_5a	5	178,753	13.88	170,123	13.75	165,169	35.2

		Local-sym1		Local-sym2		Local-sym0	
Instance	k	*Nodes*	*Time*	*Nodes*	*Time*	*Nodes*	*Times*
zeroin.i.1	49	268	7.0	268	35.49	268	36.8
zeroin.i.2	30	262	0.75	262	3,675	262	7.22
mulsol.i.2	31	237	0.85	237	10.14	237	11.24
mulsol.i.3	31	237	0.9	237	10.14	237	11.1
le450_5a	5	167,787	32.0	167,703	32.23	165,169	35.2

Experiments: The n-Queens Problems

	No-sym			Global-sym			Local-sym0		
n	$Sols$	$Nodes$	$Time$	$Sols$	$Nodes$	$Times$	$Sols$	$Nodes$	$Times$
10	724	19,744	0.03	362	9,872	0.03	353	9,592	0.12
11	2,680	85,939	0.1	1,382	43,958	0.07	1,305	41,972	0.6
12	14,200	416,828	0.28	7,100	208,414	0.25	6,839	203,120	2.67
13	73,712	2,154,845	2.69	37,361	1,093,606	1.99	35,310	1,050,401	12.88
14	365,596	11,799,746	46.95	51,726	5,899,873	20.65	43,245	5,750,997	78.85

	Local-sym1			Local-sym2			Local-sym0		
n	$Sols$	$Nodes$	$Time$	$Sols$	$Nodes$	$Times$	$Sols$	$Nodes$	$Times$
10	355	9,656	0.07	353	9,640	0.08	353	9,592	0.12
11	1,309	42,154	0.25	1,305	42,078	0.31	1,305	41,972	0.6
12	6,883	204,901	2.05	6,839	203,611	2.19	6,839	203,120	2.67
13	35,525	1,055,366	11.44	35,312	1,053,053	11.58	35,310	1,050,401	12.88
14	44,334	5,777,244	69.6	43,257	5,765,594	75.6	43,245	5,750,997	78.85

Conlusion

- **Work done :**
  - The conditions of local syntactic symmetry are weakened
  - Local symmetry is dynamically detected and exploited in a CSP tree search method
  - Local symmetry is profitable on several problems
- **Future work :**
  - Improving local symmetry detection
  - Implementing our approach in other look ahead tree search methods like MAC
  - Exporting our approach to SAT solvers
  - Adapting known global symmetry methods to deal with local symmetry