Available online at www.sciencedirect.com

SciVerse ScienceDirect

Procedia Engineering 29 (2012) 2832 – 2837

Procedia Engineering

www.elsevier.com/locate/procedia

2012 International Workshop on Information and Electronics Engineering (IWIEE)

An Improved Algorithm of CSP

Caihong Zhao[a*], Yanyan Cui[b]

[a] *Jiyuan Vocational and Technical College, Xueyuan Road No2,Henan Jiyuan 459000, China*
[b] *Department of Electrical Engineering, Jiyuan Vocational and Technical College, Xueyuan Road No2,Henan Jiyuan 459000, China*

Abstract

According to the constraint satisfaction problems, their algorithm are discussed and analyzed in detail, an improved efficiency backtracking algorithm of minimum residual heuristic value and forward test（MRV + FC）is proposed. The algorithm efficiency is superior to backtracking search (BT) algorithm and forward checking (FC) algorithm. The experiment results show that bigger the scale of the problem is, the more obvious the effect of the algorithm is..

Keywords：Constraint satisfaction program; Improved MRV+FC backtracking algorithm

1. Introduction

Since 1974, the constraint satisfaction problem (CSP) has first been proposed in image processing. As an important solution approach, CSP has been widely used in other areas of artificial intelligence and computer science. From the n-queens, graph colouring and other classic problems to scheduling, planning, resource allocation and other large application problems can be formalized as a CSP to solved [1]. After the 1990s, with the general programming language to replace the logical language of constraint satisfaction problem, the CSP application has greatly improved for the solution problem.

2. Basic overview of CSP

2.1. Basic concept

In a certain range, CSP is looking to meet the each other constraint relation′ s assignment problem for all the variables, which is composed of variables, domain and constraints [2].

* * Corresponding author.
E-mail address: zhaocaihong1974@126.com.

doi:10.1016/j.proeng.2012.01.399

CSP can be formalized as a constraint network, respectively by the set of variables, the set of each variable range and the set of constraint relationship among the variables to define, they may be expressed as triple (D, X, C), of which:

D is a set of range, D={D1，D2，…，Dn};

X is a set of variables, X={X1，X2，…，Xn}, each variable Xi corresponds to a range Di, which is all possible value′ s finite field of the variable Xi. The value of range is not necessarily the numerical type, it may be text or symbols and other types;

C is a set of constraints, C={C1,C2,…，Cm}，each constraint Cj consists of two parts: One is a set of variables V(Cj)={Xj1,Xj2,…,Xjp }，another is a set of relations R(Cj)=R(Xj1,Xj2，…，Xjp)$\subseteq$Dj1×Dj2×…×Djn，p <n. Constraint Cj is used to describe the relationship between a subset of variables.

In the CSP, feasible solution that in the case of satisfy all constraints, each variable Xi is from their belongs to range Di to a selected value di. Each of variable value constitutes the set that is a group of solutions of CSP.

2.2. Classical algorithm of CSP

Solving algorithms for CSP has four, which is respectively backtracking (BT), iterative broadening (IB), back jumping (BJ) [3]. These algorithms are used tree structure to represent the current search state, each nodes of tree can be regarded as the partial solution. At the same time, some variable values of each node have been identified, which are by a layer of parent node decision, these variables are called past variables. Relative to the past variable is the future variable, which have not been selected variable values of the variable in the node. And values of these variables may be determined at a later time. Furthermore, it has current variable that is currently considering variable. Tree structure branch is that the other possible values of variable. After selecting a branch in the tree, algorithm will assign a value for the variable, and will delete with the current part of the solution inconsistent value from the future variable value rang. When the value of future variable range is deleted into an empty, this case is called deadened, and algorithms will detect the deadened situation. Above several algorithms, the difference is that they deal with the way of future variable.

(1) BT algorithm

Each node may specify a value for the variable, which is compared with the current partial solution. If violating any constraint or conflict the situation with the pas variable value, then giving the variable value, to continue searching for the next variable value. After all values have been searched in the current range, still not found to meet the constraints or consistent values with the past variable, then the algorithm will backtracking to the previous variable and to find other value of from its range. If each variable finds a value from a range, and meeting all constraints that are not conflict with the past variables, the algorithm can stop. If expecting to find other solutions, the algorithm may proceed.

(2) IB algorithm

The algorithm is essentially depth-first search algorithm of involving a threshold b. If B is the current threshold set, and a node of searching tree or a variable is visited b times, Including the first accessed and be backtracking, then the following are not accessed the sub-nodes that may be ignored. If having not found the solution under the current limits, then the threshold may be gradually increasing. But if finding the solution or the value of threshold b is greater than or equal to the largest number of branches of searching tree, the algorithm may be end.

(3) FC algorithm

FC algorithm process is basic same with BT algorithm, each assigned variable range at least has a value, also called these values is compatible with the variables of all the assigned values. To ensure this

point, when assigning to give a variable every time, FC algorithm may delete some incompatible value with the current variable from the variable range that is never assigned. If having not assigned variable range become empty, then of the current variable value will be rejected; Otherwise, FC algorithm will continue to assignment that is not assigned a variable, until all variables are assigned. If the current variable all of values are rejected, then backtracking to the previous variable; If no variable can be backtracking, the problem is unsolvable.

(4) BJ algorithm

Processing back has difference between BJ algorithm process and BT algorithms, the rest are the same. When needed back, BJ algorithm need to find the variables that cause the failure. If the each value of current variable has conflict with the value of previous variable, then backtracking to the nearest criminal node, and not just backtracking to the previous node. If the current variable has been assigned, only back to the previous node when occurring backtracking.

3. Improved algorithms

3.1. Comparison between BT algorithm and FC algorithm

CSP is a NP complete problem, solving the key is efficiency. In the state space tree, in terms of the backtracking algorithm of the depth first iteration (Scale of the problem is q), the best case time complexity is T(q) =0(q), and the worst case time complexity is generally index order [4] . Therefore, the common backtracking of adopting the non-information search to solve large-scale constraint satisfaction problems, obtaining the results often can not meet the needs.

The difference between FC algorithm and BT algorithm considers the constraints among the variables, backtracking algorithm only considers among the variables of having been assigned whether or not the constraints, but forward check also need consider the variables that will be assigned, whether they comply with constraints. In other words, the constraint requires of forward check is higher. Table 1 is the BT algorithm and FC Algorithms.

Table 1. The efficiency comparison between Backtracking search and Forward check(P=6.5)

N	BT algorithm		FC algorithm	
	The number of constraints	The number of constraints sum	The number of constraints	The number of constraints
1	0	0	3.839999e+03	3.839999e+03
2	2.560000e+02	2.560000e+02	2.420374e+04	2.804374e+04
3	4.392960e+03	4.648960e+03	9.526736e+04	1.233111e+05
4	3.730049e+04	4.194945e+04	1.940041e+05	3.173152e+05
5	1.856164e+05	2.275658e+05	1.508427e+05	4.681578e+05
6	5.704576e+05	7.980235e+05	4.281824e+04	5.109761e+05
7	1.107687e+06	1.905711e+06	5.785941e+03	5.167620e+05
8	1.374810e+06	3.280521e+06	5.134135e+02	5.172754e+05
9	1.097797e+06	4.378317e+06	3.939228e+01	5.173148e+05
10	5.661390e+05	4.944456e+06	3.219156e+00	5.173180e+05
11	1.890020e+05	5.133458e+06	3.233508e-01	5.173184e+05
12	4.090621e+04	5.174365e+06	4.377080e-02	5.173184e+05

13	5.745078e+03	5.180110e+06	8.449645e-03	5.173184e+05
14	5.238958e+02	5.180634e+06	2.403699e-03	5.173184e+05
15	3.103146e+01	5.180665e+06	1.006026e-03	5.173184e+05
16	1.194195e+00	5.180666e+06	0	5.173184e+05

In Table 1, P satisfies the constraint probability for the variable value. It can be seen that forward check search is significantly more than backtracking search in the number of constraint expected check of the former steps. With the deepening of searching process, the number of constraint expected check of forward check search rapidly decline, and the total number of constraint check is also far less than the backtracking search in the search of entire constraint graph. Most of the work of forward check is early in the search.

It is not to say that FC algorithm is superior to BT algorithm at any time. From the Table 2, If $p < 3.0$, the total number of constraint expected checks in the BT algorithm is less than FC algorithm checks. This indicates that the smaller p, the less constraint value, the backtracking search makes the wrong choice opportunities to reduce, that is the error occurred earlier, avoiding the backtracking search to the wrong branch searching too deep. It should be noted, FC algorithm is superior to BT algorithm in most cases.

Table 2. The expected number of constraint judgment is with the trend of P

p	Backtracking search algorithm	Forward check search algorithm
1.0	124	146
1.5	168	212
2.0	230	287
2.5	318	365
3.0	453	449
3.5	667	553
4.0	1026	698
4.5	1673	918
5.0	2941	1278
5.5	5702	1930
6.0	12515	3270
6.5	31682	6502
7.0	91996	15637
8.0	1009571	143173
9.0	12043540	1730012

From the above analysis is not difficult to see that the efficiency of FC algorithm is better than BT algorithm in most cases.

3.2. The search order of affecting algorithm efficiency

Processed variable order and the value of variable range is assigned to the different order, they are also greatly affect the efficiency of the search strategy [2].

(1)The heuristic minimum width ordering (MWO)

In the problem, some variables are more variable constraints than the other variable. For the heuristic minimum width ordering strategy, all the variables will be the total ordering that have minimum width, then based on this order to deal with variable. This strategy is less variable constraint variables to be placed behind handling, which may make the request backtracking to reduce.

(2) The heuristic minimum residual value ordering (MRV)

MRV suggest that tasks of the most likely to cause failure are firstly executed; its purpose is to find the failure as soon as possible, which may save the cost of searching. Under this strategy, to be processed next the object should be the most variable on their constrains. A simple method is to measure the variables range, making the next processed variables range is the smallest.

In a simple backtracking algorithm, such as BT algorithm, variable Range is static. Therefore, using MRV means that before the start of searching, the variables according to their range size may be arranged in ascending order. When the MRV and the prediction algorithm are used together, variable order is dynamic. After giving each variable assignment, at the same time testing constraints, then to comparing the variables discourse of all not assigned, to choose the variable of having a minimum range.

(3) Maximum cardinality ordering (MCO)

Between the maximum cardinality ordering and the minimum width ordering can be regarded as roughly the same strategy.

Table 3. The compared efficiency of three algorithms(P=6.5)

N	BT algorithm the number of constraints	FC algorithm the number of constraints	MRV + FC improved backtracking algorithm the number of constraints
1	0	3.839999e+03	4.839696e+03
2	2.560000e+02	2.420374e+04	3.804374e+04
3	4.392960e+03	9.526736e+04	1.233111e+05
4	3.730049e+04	1.940041e+05	2.173152e+05
5	1.856164e+05	1.508427e+05	1.308421e+05
6	5.704576e+05	4.281824e+04	3.231877e+04
7	1.107687e+06	5.785941e+03	3.185925e+03
8	1.374810e+06	5.134135e+02	4.347367e+02
9	1.097797e+06	3.939228e+01	2.874955e+01
10	5.661390e+05	3.219156e+00	1.246322e+00
11	1.890020e+05	3.233508e-01	2.233508e-01
12	4.090621e+04	4.377080e-02	2.375621e-02
13	5.745078e+03	8.449645e-03	4.549743e-03
14	5.238958e+02	2.403699e-03	1.321765e-03
15	3.103146e+01	1.006026e-03	1.000021e-03
16	1.194195e+00	0	0

3.3. Improved efficiency backtracking algorithm of minimum residual heuristic value and forward test (MRV + FC)

The minimum residual heuristic value will give priority to handle with the minimum range space variables. When a variable range is empty, MRV will select the variable and be detected immediately that is failure; In fact, forward test may enable the relationship constraints early in the search. Once a variable is assigned, forward test process will consider all adopted constraints and the unassigned variables of the associated variables. At the same time, it will delete the range elements that are not satisfied constraints. Obviously, the starting point of two improved strategy lie to the search tree pruning as soon as possible, which is usually the key to improve the efficiency of search algorithm. Certainly, the minimum residual heuristic value can be combined with forward test, improved backtracking algorithm of MRV and FC combination are usually effective for large-scale constraint satisfaction problems.

MRV + FC Improved backtracking algorithm of CSP is described as follows:

(1) To obtain the variables and constraint information of CSP, establishing the CSP task and to determine the range of each variable according to constraint information, if having range is empty, then the problem is no solution;

(2) i = 1 (i for the current task number);

(3) If i is greater than the number of tasks to be solved, the solution of the problem has been terminated

(4) Using the minimum residual heuristic value, and according to range space to carry on the ascending order for remaining solved tasks. In other words, fewer resources are selected priority;

(5) Elements of the successive inspection task variables Xi Range, until using the forward test of relational constraints, all remaining tasks are optional resources, which will be detected;

(6) If the variable range Xi has been crossed, then i = i-1; Otherwise, the tasks Xi will be assigned and modify the range of related remaining tasks, i=i+1, and go to step 3;

(7) If i=0, then the problem will be no solution and termination; Otherwise, resuming by the Xi causes to modification of related tasks range, to cancel the assignment of Xi, and to turn 4;

The experiment is in the 1G memory, 2.4GHz Intel(R) Celeron(R) CPU，Microsoft Windows XP Professional to carry on, the algorithm designs use Visual C + + 6.0.

As can be seen from Table 3, the efficiency of improved backtracking algorithm is superior to backtracking search algorithm and forward checking algorithm. At the same time, the larger problem, the effect is more obvious.

4. Conclusions

In this paper, it is detailed discussion and analysis for constraint satisfaction problem algorithms discussion and analysis of detailed, and to propose an improved backtracking algorithm (MRV + FC), its efficiency is superior to backtracking search algorithm (BT) and forward checking algorithm (FC). The results show that the larger problem, the effect is more obvious.

References

[1] Stuart Russell．Artificial Intelligence(Zheyi Jiang Translation)．Bei jing: People Post Press;2004

[2] Yingxin Jiang, Jigui Sun. Solving Constraint Satisfaction Problem and Brief Introduction to ILOG SOLVER System. Journal of Jilin University;2002,(1)

[3] Ke Xu, Fu Li. An Average Time Analysis of Backtracking on Random Constraint Satisfaction Problems．Journal of Software;2000,(3)

[4] Qingyang Fu, Xiaodong Wang. Algorithms and Data Structures (2nd ed). Bei jing: Electronics Industry Press;2001