52. IWK

Internationales Wissenschaftliches Kolloquium

International Scientific Colloquium

PROCCEDINGS | 10 - 13 September 2007

FACULTY OF COMPUTER SCIENCE AND AUTOMATION

COMPUTER SCIENCE MEETS AUTOMATION

VOLUME I

Session 1 - Systems Engineering and Intelligent Systems

Session 2 - Advances in Control Theory and Control Engineering

Session 3 - Optimisation and Management of Complex Systems and Networked Systems

Session 4 - Intelligent Vehicles and Mobile Systems

Session 5 - Robotics and Motion Systems

TECHNISCHE UNIVERSITÄT ILMENAU

Bibliografische Information der Deutschen Bibliothek
Die Deutsche Bibliothek verzeichnet diese Publikation in der deutschen Nationalbiografie; detaillierte bibliografische Daten sind im Internet über **http://dnb.ddb.de** abrufbar.

ISBN 978-3-939473-17-6

Impressum

Herausgeber:	Der Rektor der Technischen Universität Ilmenau Univ.-Prof. Dr. rer. nat. habil. Peter Scharff
Redaktion:	Referat Marketing und Studentische Angelegenheiten Kongressorganisation Andrea Schneider Tel.: +49 3677 69-2520 Fax: +49 3677 69-1743 e-mail: kongressorganisation@tu-ilmenau.de
Redaktionsschluss:	Juli 2007
Verlag:	Technische Universität Ilmenau/Universitätsbibliothek Universitätsverlag Ilmenau Postfach 10 05 65 98684 Ilmenau www.tu-ilmenau.de/universitaetsverlag
Herstellung und Auslieferung:	Verlagshaus Monsenstein und Vannerdat OHG Am Hawerkamp 31 48155 Münster www.mv-verlag.de
Layout Cover:	www.cey-x.de
Bezugsmöglichkeiten:	Universitätsbibliothek der TU Ilmenau Tel.: +49 3677 69-4615 Fax: +49 3677 69-4602

Preface

Dear Participants,

Confronted with the ever-increasing complexity of technical processes and the growing demands on their efficiency, security and flexibility, the scientific world needs to establish new methods of engineering design and new methods of systems operation. The factors likely to affect the design of the smart systems of the future will doubtless include the following:

- As computational costs decrease, it will be possible to apply more complex algorithms, even in real time. These algorithms will take into account system nonlinearities or provide online optimisation of the system's performance.
- New fields of application will be addressed. Interest is now being expressed, beyond that in "classical" technical systems and processes, in environmental systems or medical and bioengineering applications.
- The boundaries between software and hardware design are being eroded. New design methods will include co-design of software and hardware and even of sensor and actuator components.
- Automation will not only replace human operators but will assist, support and supervise humans so that their work is safe and even more effective.
- Networked systems or swarms will be crucial, requiring improvement of the communication within them and study of how their behaviour can be made globally consistent.
- The issues of security and safety, not only during the operation of systems but also in the course of their design, will continue to increase in importance.

The title "Computer Science meets Automation", borne by the 52nd International Scientific Colloquium (IWK) at the Technische Universität Ilmenau, Germany, expresses the desire of scientists and engineers to rise to these challenges, cooperating closely on innovative methods in the two disciplines of computer science and automation.

The IWK has a long tradition going back as far as 1953. In the years before 1989, a major function of the colloquium was to bring together scientists from both sides of the Iron Curtain. Naturally, bonds were also deepened between the countries from the East. Today, the objective of the colloquium is still to bring researchers together. They come from the eastern and western member states of the European Union, and, indeed, from all over the world. All who wish to share their ideas on the points where "Computer Science meets Automation" are addressed by this colloquium at the Technische Universität Ilmenau.

All the University's Faculties have joined forces to ensure that nothing is left out. Control engineering, information science, cybernetics, communication technology and systems engineering – for all of these and their applications (ranging from biological systems to heavy engineering), the issues are being covered.

Together with all the organizers I should like to thank you for your contributions to the conference, ensuring, as they do, a most interesting colloquium programme of an interdisciplinary nature.
I am looking forward to an inspiring colloquium. It promises to be a fine platform for you to present your research, to address new concepts and to meet colleagues in Ilmenau.

Professor Peter Scharff
Rector, TU Ilmenau

Professor Christoph Ament
Head of Organisation

Table of Contents

C O N T E N T S

2 Advances in Control Theory and Control Engineering

Paper submitted after copy deadline

2 Advances in Control Theory and Control Engineering

A. Yu. Nedelina / W. Fengler

DIPLAN: Distributed Planner for Decision Support Systems

INTRODUCTION

An analysis of the problems in the efficiency management reveals the necessity to introduce new intelligent technologies to the management process. The management of complex objects and processes of various natures in the context of rigid real-time requirements is a complicated problem. This is in particular the case in applications like energy and power management, water resource management, nuclear waste management, reactor dynamics, and fuel management etc.

Artificial Intelligence (AI) planning technologies form the basis of the intelligent management, because questions, which are related to the planning theory as a component of the task solution theory, occupy one of the leading positions in the field of intelligent systems research.

Planners are Artificial Intelligence applications for plans generating – optimal or admissible sequences of actions allowing to achieve the desired goal or to solve the problem situations. These programs are in many cases embedded into real time decision support systems (RTDSS) or analytical information systems that must show real-time behavior. Since this request usually can't be met by a complete systematic search within a search space of all possible solutions, such programs are usually equipped with heuristic functions and other types of guidance, in order to respond promptly.

A planning problem solution in complicated dynamic environments with real-time constraints requires the use of an integrated approach with parallel computing that allows to reduce the time of the problem solution sufficiently.

In the paper there are presented the concept and prototype of DIPLAN planner based on a distributed approach for solving AI planning problem.

PLANNING AS STATE-SPACE SEARCH

Most planning problems can be considered as searches through the space of world states.

Planning domains are characterized by a current world state or current state of the intellectual system environment along with regulations for state transitions within a state space. In DIPLAN, world states are described by boolean state variables (fluents).

A planning domain *D* is defined by the collection *D* = <*F, S, A, R*>, where

- F is a finite set of fluents;
- $S \subseteq 2^F$ is a finite set of states;
- A is a finite set of actions, $A \subseteq M$ = {M1,…, Mn} of RTDSS systems;
- $R \subseteq S \times A \times S$ is a transition relation.

Action $a \in A$ is executable in $s \in S$ if $\exists s' \in S : R(s,a,s')$.

A state space problem *P* for a planning domain *D* is defined by the collection

P = <D, I, G>, where

- $I \subseteq S$ is an initial state;
- $G \subseteq S$ is the set of goal states.

A plan π for a planning problem *P* in a planning domain *D* is a sequence of state-action pairs that transforms the initial state $s_0 \in I$ into one of the goal states $\in$ G, where

- {(s, a) : s $\in$ S, a $\in$ A, a executable in s};
- $\exists\ (s_0, a) \in \pi : s_0 \in I, a \in A$;
- $\exists\ R(s,a,s') : s \in S, a \in A, s' \in G$.

PLANNING APPROACH

The proposed distributed planner (DIPLAN) introduced here, concerns state-space planning systems.

The state space in DIPLAN is represented as a tree *T* = (*V*, *E*), where *V* is a set of nodes and *E* is a set of arcs (Fig. 1). Each node in the tree is a state of environment characterized by a number of parameters. Root of the tree is an initial state of environment. Arcs are labeled by environment activities that transform one state to another.

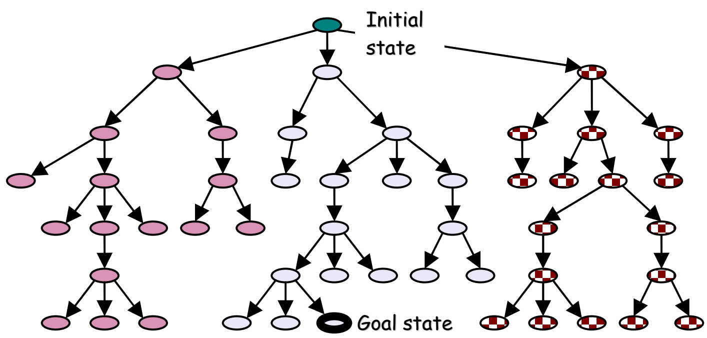

Figure 1. The DIPLAN state-space search tree

In that case, the state-space planning problem in DIPLAN is a search through the state-space for an optimal (shortest) path from initial state to some desired goal state of the goal state set.

The DIPLAN planning process flow diagram is presented on the Fig.2.

Basic methods and algorithms of state-space search for a planning system have been analyzed in the context of this research work [1,2]. The heuristic search algorithms allow to improve the efficiency of search by means of problem reduction. Hence, the current research work focuses on the heuristic search algorithms.

Unfortunately, the time calculating and memory requirements decrease the applicability of the considered heuristic methods in software tools such as RTDSS.

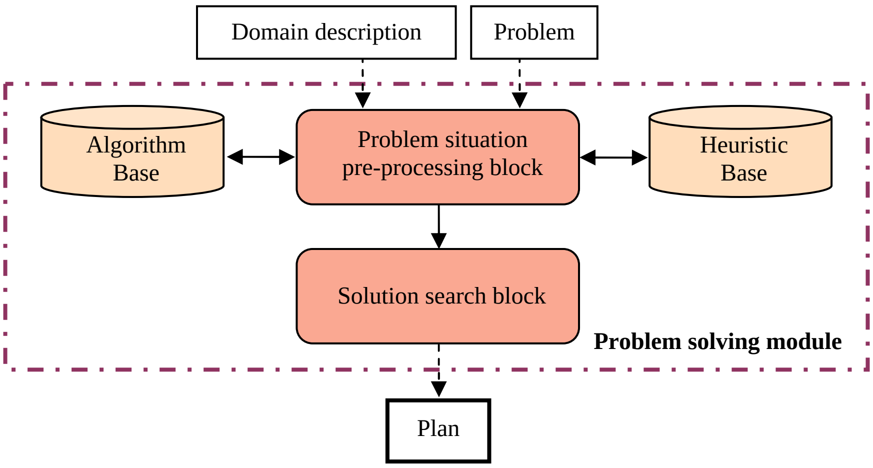

Figure 2. The scheme of DIPLAN blocks interaction

The following solution was proposed in the paper: parallel modifications of the heuristic search algorithms, optimized for the parallel hardware architecture [3]. Distributed computation approach allows to solve the planning problem by decomposition of a search problem into independent search spaces (OR- subtrees). Each search subtree may be allocated to separate processor (Fig. 1).

A challenging feature of the modern Artificial Intelligence applications is the ability to distribute the workload among several processors, in order to increase the execution speed. Although the technology of parallel architectures is quite mature and a large number of parallel systems are available at a reasonable cost, there are not many software products that can exploit these capabilities.

The classification of parallel hardware architectures is given on Fig. 3.

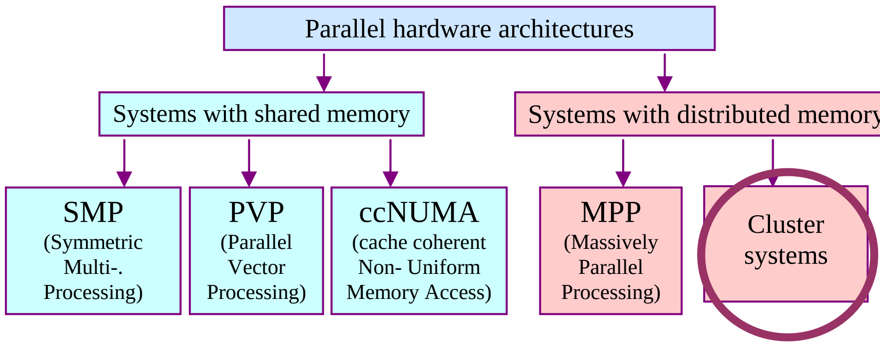

Figure 3. The classification of parallel systems by memory architecture

The cluster systems are chosen for parallel implementation of DIPLAN planner due to best correlation between cost / performance [4].

DIPLAN PROTOTYPE

The DIPLAN architecture contains the following modules (Fig. 4):

- **Interface module** – uses graphical tools for data representation;
- **Information processing module** – responsible for input data gathering, goal and task specifications, and also for interaction with the external systems;
- **Solution search module-** provides a state-space search using a decomposition of the general task to local disjunctive subtasks, distributes

the subtasks between the processors, it also controls the lifecycle of the processors (i.e. it can generate and modify the task and initial data).

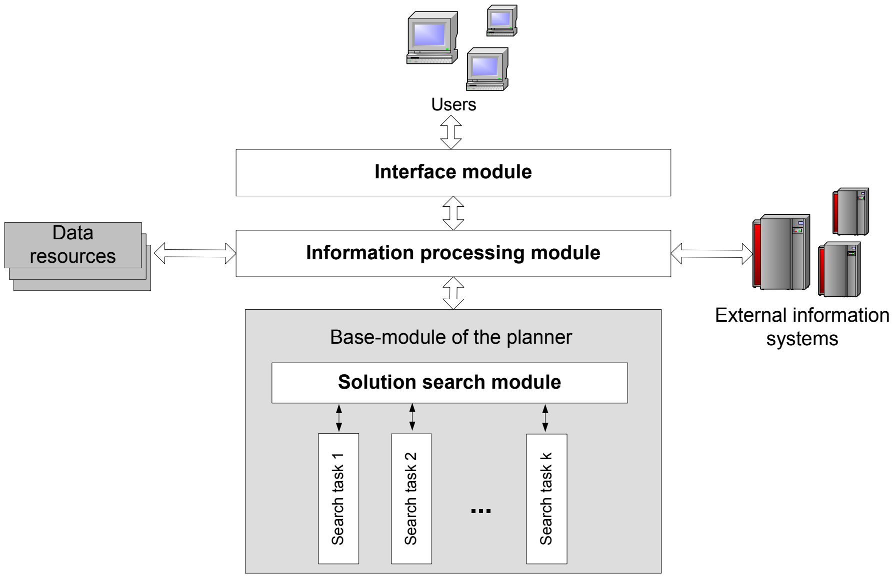

Figure 4. The generalized architecture of the DIPLAN planner

As shown in Fig. 4, the main functional component of DIPLAN is the problem solving module, which based on a distributed-memory Multiple Instruction Multiple DATA (MIMD) model.

The distributed environment of the Solution search module of DIPLAN is based on message-passing model with the usage of the communication package MPI (Message Passing Interface) for cluster systems [5]. During the parallel processing of solution search several individual processes work in a distributed system and communicate via data streams to perform a common task [6].

For the problem solving module was proposed and implemented parallel version of IDA* (Iterative-Deepening A*) algorithm. This algorithm is described in details in [3].

The DIPLAN was developed using C++ in the Microsoft Visual C++ 6.0 environment. The Solution search module was implemented in Visual C++ 6.0 with the usage of a parallel MPI technology and the package mpich.nt.1.2.5.[5], in a mode of parallel computation modeling on one processor.

EMPIRICAL RESULTS

For measuring the performance of solution search module we take a benchmark model – the 15-Puzzle [7,8]. It is simple, but has a combinatorial large problem space of 10^{13} states. Finding a solution to the 15-puzzle in terms of the state-space problem is a finding an optimal (shortest) solution path in a tree of possible puzzle configurations from an initial to goal state of the puzzle.

We used our parallel IDA* algorithm with the classic heuristical evaluation function for the sliding tile puzzles, the Manhattan distance [2]. This evaluation function estimates a sum of the minimum displacement of each tile from its goal position.

The average of iterations, received on base of the empiric results on a 100 experiments, is about 53 iterations for the domain of the 15-Puzzle problem. The 15-Puzzle problem was solve using IDA* on one processor and using parallel IDA* version on 2 and 4 processors. The results of a 100 experiments for Intel Pentium IV 2,8 GHz divided to the 5 sets (20 experiments in each set) are given on Fig.5.

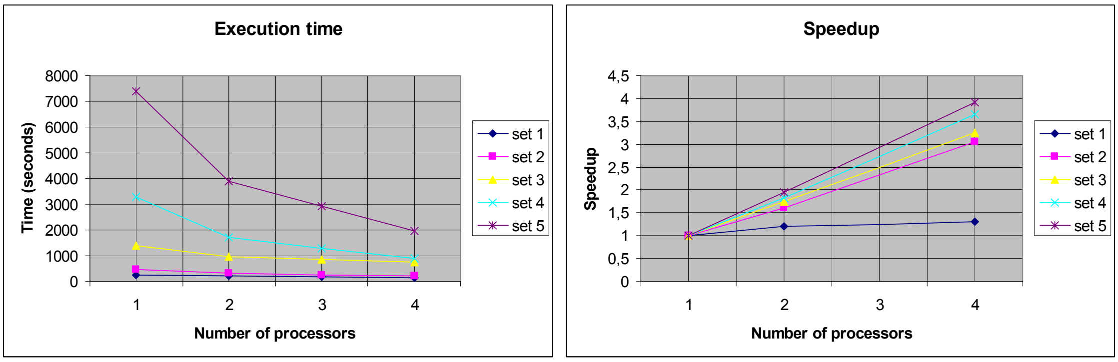

Figure 5. The diagrams of speedup and time execution for parallel IDA* version

CONCLUSION

The testing of the parallel IDA* algorithm for 15 Puzzle problem with the Manhattan distance heuristic has improved the speedup results in comparison with the consecutive version.

Later on, the developed distributed planner will be implemented on a cluster environment.

Furthermore, some real benchmark data will be tested for DIPLAN planner.

The possibilities of implementing the Solution search module of DIPLAN on symmetric multi-processing (SMP) systems will be analyzed.

References:
[
[1] N. Nilsson, “Principles of Artificial Intelligence”, Tioga Publishing Company, Palo Alto, CA, 1980.
[2] R.E. Korf, “Depth-first iterative-deepening: An optimal admissible tree search”, Artificial Intelligence 27, 1985, pp. 97-109.
[3] A.P. Eremeyev, A.Yu. Nedelina, “Methods and algorithms of strategic planning for decision support systems”, Proc. of the 6 Joint Conference on Knowledge-Based Software Engineering, IOS Press, 2004, pp. 247-252.
[4] R. Buyya, “High Performance Cluster Computing: Architectures and Systems”, Prentice Hall PTR, NJ, USA, 1999.
[5] W. Gropp, E. Lusk, R. Thakur, “Using MPI-2: advanced features of the message-passing Interface”, MIT Press, Cambridge, USA, 1994.
[6] M. Knapik, J. Johnson, “Developing intelligent Agents for distributed systems”, McGraw-Hill, New York, USA, 1998.
[7] A. Felner, R.E. Korf, S. Hanan, “Additive Pattern Database Heuristics”, J. of Artificial Intelligence Research 22, 2004, pp. 279- 318.
[8] A. Yu. Nedelina, A. P. Eremeev, W. Fengler, “Development of distributed planner for decision support systems“, Proc. of the 2nd International Conference on Dependability of Computer Systems (DepCoS - RELCOMEX 2007), IEEE, 2007, pp. 324-330.

Authors:
Dipl.-Ing. Anna Nedelina
Prof. Dr.-Ing. habil. Wolfgang Fengler
Moscow Power Engineering Institute (Technical University)
Russian Federation, Moscow
Phone: +7 916 3269342
Fax: +7 495 3174663
E-mail: anna@sytech.ru