Annals of DAAAM for 2010 & Proceedings of the 21st International DAAAM Symposium, Volume 21, No. 1, ISSN 1726-9679
ISBN 978-3-901509-73-5, Editor B. Katalinic, Published by DAAAM International, Vienna, Austria, EU, 2010
Make Harmony Between Technology and Nature, and Your Mind will Fly Free as a Bird
DAAAM Symposium

HEURISTIC ACTION PLANNING ALGORITHM FOR DISTRIBUTED MULTIAGENT SYSTEMS

SVACO, M[arko]; SEKORANJA, B[ojan] & JERBIC, B[ojan]

Abstract: *In this paper an approach of modelling multiagent action planning and task execution problems is discussed. The distributed multiagent system taken in consideration is a system of m autonomous agents. Agents are constantly communicating and collaborating in order to participate in coordinating and deciding upon their actions. A heuristic action planning algorithm (HAPA) is presented for solving multiagent action planning and task allocation. Agents are given tasks in a simplified blocks world to recreate represented structures. By utilizing the planning algorithm and through interaction agents are allocated tasks which they execute to reassemble the given form.*
Key words: *Distributed multiagent systems, autonomous planning*

1. INTRODUCTION

In this paper a heuristic action planning algorithm (HAPA) for application in multi agent systems is presented. One of the main goals is constructing a universal framework which can be implemented on various types of industrial robots. Distributed multiagent robotics is somewhat a system of repeating and recreating human behaviour patterns. Humans are much more efficient when working in groups; they exhibit more axis of freedom, more data can be handled, complex tasks can be accomplished. Robotics and primarily industrial robotics has always been a part of a central planning system where all agents (robots, machines) are under control of a PLC or some other central unit, and exhibit very low level of autonomy. The presented approach suggests that some handling tasks may be accomplished by interaction between agents in the system, and therefore some level of autonomy must be introduced. Nowadays the most flexible industrial robots have 7 (6) degrees of freedom (DOF) without the end effector (gripper). The human arm has 27 DOF (Agur and Lee), thus comparing the flexibility one robotic arm is quite limited and cannot perform all required tasks. Implementing communication protocols, two or more robotic controllers can exchange information and act together in performing more demanding operations. Each controller with its actuator unit is an agent with defined level of autonomy.

2. THE MULTIAGENT SYSTEM

Related works (Ephrati & Roscensnhein; Sycara et. al.) are mostly virtual multiagent applications and cannot be easily implemented on real industrial systems. The approaches are primarily intended for autonomous planning done by multiple agents who cannot collide and are of infinite small dimension. The developed HAPA and the distributed multiagent system (MAS) presented in this paper operate in a real world environment bounded by rules and limitations. Agents a_l ($l = 1...m$) workspace is defined as a simplified blocks world with respect to a global Cartesian coordinate system *c*. In the simplified blocks world b_j represents a building block by which agents assemble structures. Each building block has certain properties which agents perceive from their workspace: *size* (type) of a building block T (b_j) = {1, 2, 3…} and *Cartesian position and orientation* in workspace: P (b_j) = {x, y, r}. All blocks have the same width and height (single unit) but their length can vary and can be one, two, three, etc. unit lengths. That results with flexibility so building blocks can be supplemented with each other i.e. block with two unit lengths can be replaced with two blocks of single unit length and vice versa. Each agent is defined as an autonomous, self-aware entity with limited knowledge of the global workspace (Schumacher) and with some cognition of other agents. Each agent is a complete system with a separate processing unit, actuators, vision system for acquiring information from its environment, force and torque sensors for tactile feedback and other interfaces. A space function F (a_l) must be defined to determine the consumed space by an agent a_l, $F=\{x_1, y_1, x_2, y_2, r, t\}$ in time ***t***. The first pair of Cartesian coordinates depicts the first vertex of a rectangle and the second pair depicts the second vertex respectively, where *r* defines the rotation angle of the rectangle with respect to the origin point of the coordinate system *c*. New agents can be introduced to the system and some agents can be delegated tasks not concerning the problem of reassembling structures. The MAS is insensitive to dynamic changes in number of agents, whereas the impact is lower system flexibility and prolonged times for achieving final goals. Agents tasks are recreating shown structures which are defined as a final form put together from various objects with determined patterns and relationships. A structure is determined by interrelations and arrangement of objects b_j into a complex entity. Structure $S = \{R_i b_j\}$ is a set of relations R_i (i=1…m-1) between objects (b_j, j=1…n). The MAS has properties of a market organization type (Sandholm; Shoham & Leyton-Brown) where agents bid for given resources (blocks) in their workspace. Agents need to communicate and negotiate time schedules when areas of interest in the global workspace are not occupied. Agents goal are assembling the structure with available elements following the given set *S*. An example of a structure is illustrated in Fig. 1 a). Building blocks are then randomly scattered and some new objects can also be introduced (Fig. 1 b). Agents global goal is to recreate the initial structure by following a set of rules and propositions given in a knowledge base (KB):

- Mathematical rules for structure sets
- Agents capabilities
- Grasping rules and limitations
- Object properties
- Agents workspace
- Vision system patterns database
- Force and torque sensor threshold values

If a simple structure with limited number of building blocks is presented to the agents (Fig. 1 a) there might be only one or few feasible solutions (sequence of steps) how to define that structure. If complex structures are presented there might be a variety of feasible solutions. Top down disassembling or

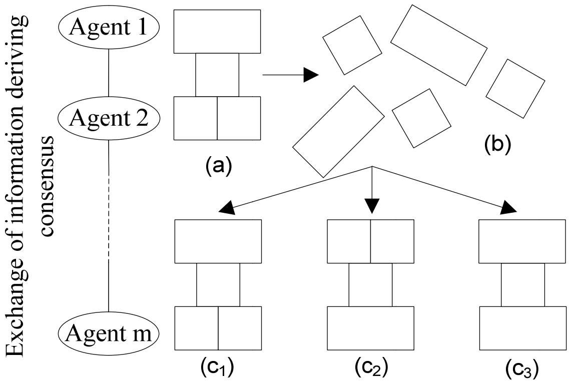

Fig.1. Simple task for the multiagent system

bottom up assembling the structure can be a means of defining a sequence of steps for the MAS. The HAPA utilizes a bottom up principle where from a provided set of objects {b_1…b_s}, can differ from the initial set {b_1…b_j}. Agents need to reach a solution in the given search space. Depending on the KB information agents can make a decision weather the desirable objectives can be preformed in accordance to proposed restrictions and limitations. Implementing an iterative algorithm a solution can be found. Each agent attains a unified set of sub goals g_t which fulfil the global goal *G*. All agents are of same relevance and their decisions are equally evaluated. When an agent bids for a task all the other members of the MAS gain that request. Task execution can be done synchronous or asynchronous giving the space functions *F* of the agents. A resource function *C* must be introduced as a measure of resource and time consumption. $C\ (a_l,\ b_j,\ e)$ is a function of agents' a_l position, specifications of a building block b_j (size and position in global workspace) and the position *e* where that block is planned to be embedded. Agents have on their disposal basic operators:

- *Pick* $(b_i\ ,gr)$ – agent picks up a block b_i with a grasping method *gr*
- *Move* $(p_1,\dots,p_r,\ t_1,\dots,t_r)$ – agents move in the global workspace from point p_1 to point p_r through *r*-2 interpolation points with motion specification t_r defined for each point.
- *Place* (b_j) - agent places a block b_j
- *Push (f, d, s)* – agent uses force sensor for auxiliary action of pushing an object with force/torque threshold *t* in vector direction *d* for *s* units
- *Vision* – the vision operator is used for identifying objects and their coordinates in *c*

Utilizing these operators agents construct sequence of actions for accomplishing each sub goal. By consecutively achieving all sub goals the global goal *G* is fulfilled and agents await further tasks.

3. IMPLEMENTATION AND RESULTS

The HAPA has been tested to provide solutions for a structure as shown in Fig. 1. When multiple solutions are possible (Fig. 1 c) the MAS executes the one where $\sum C$ in the entire solution set is minimal. Currently only two dimensional structures are being solved but their solutions due to use of real world objects has to be three dimensional. Implementation has been done on a multiagent robotic system shown in Fig. 2. Collision avoidance between objects and agents is fully implemented and to a certain degree collision between moving agents. Currently there are no algorithms to solve real time agent collision or they exist but with limitations. Collision between two agents with kinematic chains of 3 DOF can be solved in a definite period of time (Curkovic & Jerbic). Implementation on MAS where each agent has multiple DOF would be time consuming; for that reason space function (*F*) is used. The MAS presented in this

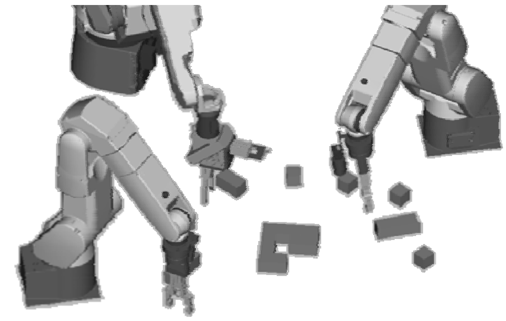

Fig.2. Virtual representation of the multiagent robotics system

research can handle only known objects in 2D scenes. Agents have physical limitations so their joint workspace is limited by their geometrical features. For successful grasping and manipulating with new objects learning object features and grasping points is needed.

4. CONCLUSION

This paper shows that a multiagent (robotic) system can be implemented to solve complex tasks more efficiently than a single agent: with more processing power (each agent's controller unit) calculations can be done faster. Using the developed HAPA agents delegate tasks and actions. In the future further generalization will be introduced where agents will be able to autonomously distinguish and solve new objectives and problems as depicted in Fig. 3. First step is implementation of reassembling 3D structures. For that purpose the KB will need to comprise rules regarding "laws of gravity" and etc. Further research will be concentrated on introducing new objects to the MAS. Taking into consideration the KB (known similar objects) agents will be able to find or construct grasping methods weather they can do it individually or assisted by other agents.

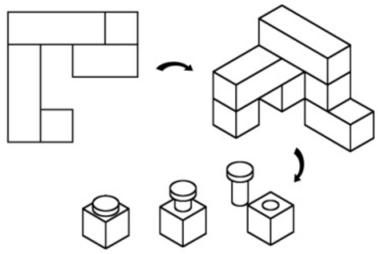

Fig. 3. Further tasks for the multiagent system

5. REFERENCES

Agur A. M. R., Lee M. J. (1999). *Grant's Atlas of Anatomy*, Lippincott Williams and Wilkins, ISBN 978-0683302646

Curkovic P., Jerbic B. (2010). Dual-Arm Robot Motion Planning Based on Cooperative Coevolution, In: *Emerging Trends in Technological Innovation,* Turner *et al.*, pp. 169-178, Springer Boston, ISBN 978-3-642-11627-8

Ephrati E., Roscensnhein J. (1994). Divide and conquer in multiagent planning, *Proc. of the 12th National Conference on AI,* pp. 375-380, Seattle, ISBN 0-262-61102-3, AAAI

Sandholm T. (1993). An Implementation of the Contract Net Protocol Based on Marginal Cost Calculations, *Proc. of the 11th Conference on AI*, pp. 256–262, ISBN 0-262-51071-5

Schumacher M. (2001). *Objective Coordination in Multi-Agent System Engineering*, Springer, ISBN 3-540-41982-9, NY

Shoham Y., Leyton-Brown K. (2009). *Multiagent Systems: algorithmic, game-theoretic and logical foundations*, Cambridge Uni. Press, ISBN 978-0-521-89943-7, NY

Sycara K., Roth S., Sadeh N., Fox M. (1991). Distributed Constrained Heuristic Search, *IEEE Trans. on System, Man and Cybernetics*, pp. 1446–1461, ISSN: 0018-9472