International Journal of Research in Advent Technology (E-ISSN: 2321-9637) Special Issue
National Conference "IAEISDISE 2014", 12-13 September 2014

An Autonomous Agent Is an Intelligent Agent Operating On an Owner's Behalf without Any Interference

Sompal[1], Ravinder Singh[2], Anshul[3]
CSE Department[1], CSE Department[2], CSE Department[3]
Manav Bharti University, Solan (H.P.) [1, 2], MDU Rohtak[3]
Email: sompal26@gmail.com[1], *ravimadhan@gmail.com*[2], *nshlhd@gmail.com*[3]

Abstract: Advances in computing hardware, software, and network technology have enabled a new class of interactive applications involving 3D animated characters to become increasingly feasible. Many such applications require algorithms that allow both autonomous and user-controlled animated human figures to move naturally and realistically in response to task-level commands. The framework design is inspired by research in motion planning, control, and sensing for autonomous mobile robots. In particular, the problem of synthesizing motions for animated characters is approached from the standpoint of modeling and controlling a "virtual robot".

Keywords: Autonomous agent, robot, animated agent, virtual sensors, Navigation, etc.

1. INTRODUCTION

Computer graphics involves the creation of images of things both real and imagined. Computer animation entails the generation and display of a series of such images. In either case, mathematical models are used to represent the geometry, appearance, and motion of the objects being rendered. Recent improvements in computer hardware and software have enabled mathematical models of increasing complexity to be used. Indeed, synthetic images of some objects have been created which bear remarkable similarity to actual photographs of real objects. As computing technology continues to improve, we can expect computer graphics and animation of increasing realism in the future. Ultimately, the dream of virtual reality may be fully realized, in which artificial scenes can be created that are indistinguishable from real life.

The fundamental challenges in computer graphics and animation primarily in having to deal with complex geometric, kinematic, and physical models. Researchers have developed 3D scanning, procedural modeling, and image-based rendering techniques in order to automate the modeling of the geometry and appearance of complex objects. For the purposes of animation, algorithms are needed to automate the generation of motions for such objects. Moreover, even objects whose geometry and appearance are very simple, may be difficult to animate due to the number of moving parts/joints which must be dealt with.

2. MANUSCRIPTS

A. Animation framework

The concept of an autonomous agent has received a fair amount of attention in the recent artificial intelligence literature. Ironically, even though several international conferences are held annually on the topic, the term agent lacks a universally accepted definition. The word autonomy is similarly difficult to define precisely. In the context of animated characters, we will consider autonomous to mean not requiring the continuous intervention of a user, and adopt the following general definition as a starting point:

"*An autonomous agent is a system situated within and a part of an environment that senses that environment and acts on it, over time, in pursuit of its own agenda and so as to effect what it senses in the future."*

Franklin & Graesser, 1996
"Is it an Agent, or just a program?"

B. Physical agents:

A physical agent (what we commonly conceive of as a "robot") is essentially a collection of hardware and software that operates in the physical world. It has control inputs (motors) that apply forces and torques on its joints, which in turn cause the robot to move itself or other objects in it environment. The robot also comes equipped with various kinds of sensors that provide feedback on its motion, along with information about its surroundings.

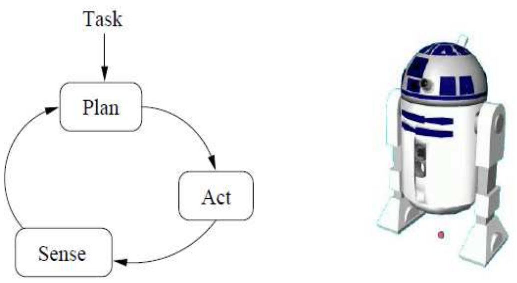

Figure: The control loop of a robot (physical agent).

C. **Animated agents**

An animated agent is a software entity that exists and moves within a graphical virtual world. We consider the problem of motion generation for an animated agent as essentially equivalent to that of designing and controlling a virtual robot. Instead of operating in the physical world, an animated agent operates in a simulated virtual world, and employs a "virtual control loop".

1. **Control Inputs:** The animated agent has a general set of control inputs which define its motion. These may be values for joint variables that are specified explicitly, or a set of forces and torques that are given as input to a physically-based simulation of the character's body.

2. **Virtual Sensors:** The animated agent also has a set of "virtual sensors" from which it obtains information about the virtual environment.

3. **Planning:** Based on information provided by the virtual sensors and the agent's current tasks and internal state, appropriate values for its control inputs are computed.

3. A VIRTUAL ROBOT ARCHITECTURE

If every atomic task was very narrowly and explicitly defined, one could imagine simply maintaining a vast library of pre-computed, captured, or hand-animated motions to accomplish every possible task. The reality is that, in general, tasks cannot be so narrowly defined, lest the set of possible tasks become infinite. Instead, atomic tasks are typically specified at a high-level, and apply to a general class of situations, such as "move to a location" or "grasp an object". Thus, they are specific in one sense (the required end result), but also very vague (how the result is achieved).

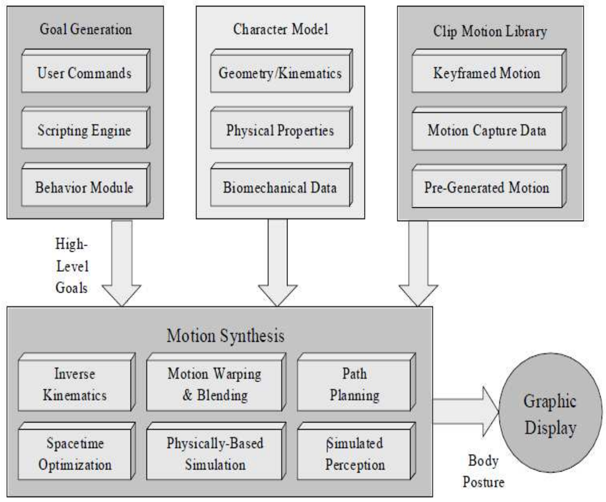

Figure: Resources available for motion synthesis.

A block diagram of the agent's available software resources is depicted in Figure 2.5. A set of high-level goals are generated and passed to the motion synthesis software. Utilizing some or all of the fundamental software components and data libraries, the resulting motion for the character is computed and passed to the graphic display device for rendering.

Goal-directed navigation

Navigation is an important task that frequently arises in the context of animated characters. Navigation task commands require that a character move from one location to another within a virtual environment filled with obstacles. Consider the case of an animated human character given the task of moving from a starting location to a goal location in a flat-terrain virtual environment. An acceptable navigation algorithm should produce a set of natural-looking motions to accomplish the task while avoiding obstacles and other characters in the environment.

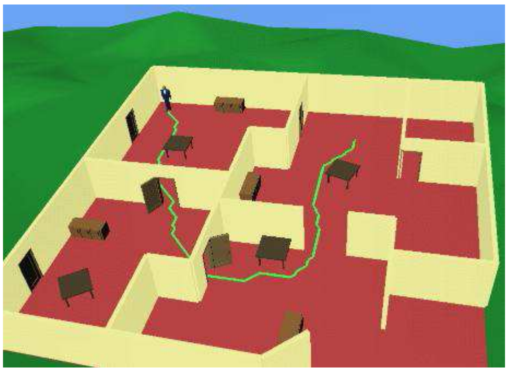

Figure: A human character navigating in a virtual office.

4. NAVIGATION USING PLANNING AND CONTROL

Our strategy will be to divide the computation into two phases: path planning and path following. The planning phase computes a collision-free path to the goal location, while the path following phase synthesizes the character's motion along the computed path.

Navigation goals are specified at the task level, and the locomotion animation is created using cyclic motion capture data driven by a path-following controller that tracks the computed path. Although the planner presented here is limited to finding navigation paths that lie in a plane, the algorithm can potentially be modified to compute paths that include vertical motions of the character

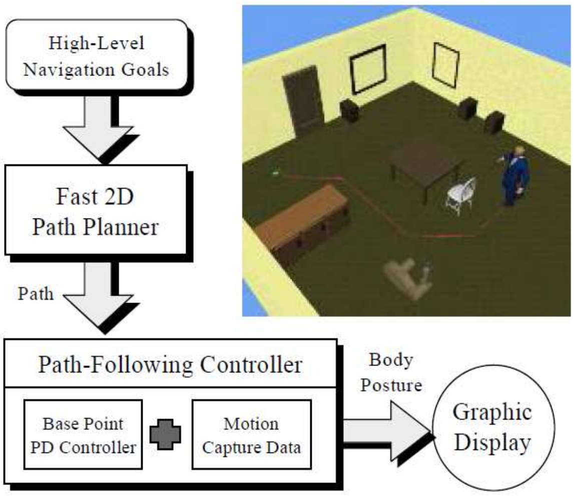

Figure: Data flow for the navigation strategy

5. SENSING THE ENVIRONMENT

Sensing for autonomous agents may be defined as "the process of gathering or receiving data about the environment and the agent itself". Architectures for autonomous agents usually provide some method of interfacing the agent to the environment through sensors. Sensory information can be encoded at both a low level and a high level and utilized by high-level decision-making processes of the agent. Sensing consists of two fundamental operations: gathering data about the environment, and interpreting the data. For a physical agent, gathering data involves devices such as cameras, laser range finders, and sonars, while interpreting data involves software algorithms (e.g. image segmentation, 3D model reconstruction, object recognition, motion estimation, etc). For an animated agent, sensing is performed via software, without the use of devices. However, the two fundamental operations of gathering and interpreting data remain.

6. CONCLUSION

The fundamental challenges in computer graphics and animation lie primarily in having to deal with complex geometric, kinematic, and physical models. The development of better software tools has advanced the state of the art in the modeling and rendering of these models. However, software tools for the automatic generation of motion are still relatively scarce. In particular, techniques are needed to animate both autonomous and user-controlled human figures naturally and realistically in response to high-level task commands.

The preceding chapters of this paper have presented a research framework aimed at facilitating the high-level control of animated characters in interactive virtual environments.

The next section briefly summarizes the key ideas contained in this paper, and the following section contains a concluding discussion.

1. **Goal-directed Navigation:** We present a technique for quickly synthesizing collision- free motions for animated human figures given high-level navigation tasks amidst obstacles in changing virtual environments. The method combines a fast 2D path planner, a path-following controller, and uses cyclic motion capture data to generate the underlying animation. In order to provide a feedback loop to the overall navigation strategy, we incorporate an approximate synthetic vision module for simulating the visual perception of the character.

2. **Object Manipulation:** We introduce a new manipulation planner designed for efficiently generating collision-free motions for single-arm manipulation tasks given high-level commands. For moving an object, the planner automatically generates the motions necessary for a human arm to reach and grasp the object, reposition it, and return the arm to rest. The planner searches the configuration space of the arm, modeled as a kinematic chain with seven degrees of freedom. Goal configurations for the arm are computed using an inverse kinematics algorithm that attempts to select a natural posture.

7. DISCUSSION

Nearly all known motion generation techniques developed in the graphics and robotics literature can be unified under a common framework by viewing the motion synthesis problem as one whose solution involves two fundamental tools: model-building and search. Specifically, solving a motion synthesis problem almost always involves constructing a suitable model and searching an appropriate space of possibilities. At present, no single motion synthesis strategy can adequately handle the variety of tasks that can arise. Instead, different tools and techniques can be used in combination and tailored to meet the requirements of specific tasks. We believe that assembling a library of algorithms and motion data libraries under the unifying framework of an animated agent will help to facilitate the development of autonomous characters with realistic motions and behaviors.

REFERENCES

[1] W.W. Armstrong and M.W. Green. The dynamics of articulated rigid bodies for the purposes of animation. The Visual Computer, 1:231{240, 1985.
[2] C.G. Atkeson and J.M. Hollerbach. Kinematic features of unrestrained arm movements. MIT AI Memo 790, 1984.
[3] S. Arya, D. M. Mount, N. S. Netanyahu, R. Silverman, and A. Y. Wu. An optimal algorithm for approximate nearest neighbor searching. To appear in the Journal of the ACM.
[4] Y. Aydin and M. Nakajima. Realistic articulated character positioning and balance control in interactive environments. In In Proceedings of CA '99 : IEEE International Conference on Computer Animation., pages 160{168, Geneva, Switzerland, May 1999.
[5] R. Alami, T. Simeon, and J.P. Laumond. A geometrical approach to planning manipulation tasks: The case of discrete placements and grasps. In H. Miura and S. Arimoto, editors, Robotics Research 5, pages 453{459. MIT Press, 1990.
[6] N. Amato and Y. Wu. A randomized roadmap method for path and manipulation planning. In Proc. of IEEE Int. Conf. Robotics and Automation, pages 113{120, Minneapolis, MN, 1996.
[7] A. Bruderlin and T. Calvert. Goal-directed dynamic animation of human walking. In Proc. SIGGRAPH '89, pages 233{242, 1989.
[8] J. Bobrow, S. Dubowsky, and J. Gibson. Time-optimal control of robotic manipulators. Int. Journal of Robotics Research, 4(3), 1985.
[9] B. M. Blumberg and T. A. Galyean. Multi-level direction of autonomous creatures for real-time virtual environments. In Robert Cook, editor, Proc. SIGGRAPH '95, Annual Conference Series, pages 47{54, August 1995.
[10] J. Barraquand, L.E. Kavraki, J.C. Latombe, T.Y. Li, R. Motwani, and P. Raghavan. A random sampling scheme for path planning. Int. J. of Robotics Research, 16(6):759{774, 1997.
[11] J. Bates, A. B. Loyall, and W. S. Reilly. An architecture for action, emotion, and social behavior. In Artificial Social Systems: Proc of 4th European Wkshpon Modeling Autonomous Agents in a Multi-Agent World. Springer-Verlag, 1994.