An Architecture for Animated Agent Simulation

FERNANDO WAGNER DA SILVA   LUIZ GONÇALVES   MARA RIOS   ANTONIO OLIVEIRA

Laboratório de Computação Gráfica - COPPE
Universidade Federal do Rio de Janeiro (UFRJ), CP 68511 Rio de Janeiro RJ 21945-970
{nando,lmarcos,mara,oliveira}@lcg.ufrj.br

Abstract. This work describes ongoing research towards finding a methodology for simulation of agents which are alive creatures or robots (live in real environments). In general, modeling such "animated" agents involves dealing with several physics and biology constraints (gravity, sensing). The proposed architecture deals with dynamics and kinematics issues and provides a well structured way to control the simulation step. Two different applications – "Roger the Crab" and "The Worm" – show the versatility of the model.

Agent Model

To animate general articulated objects, a computational model is employed to correctly represent its main structures allowing a robust control of the animation parameters over time. The Agent Model consists of a skeleton formed by links, joints and sensors, organized hierarchically. Links are geometric segments of the object allowed to move, and may be represented by rigid or deformable bodies depending on the application. Joints are geometric constraints used to connect the links, allowing relative movement (also known as Degree of Freedom, DOF) between the segments of the structure. Joints are classified as revolute, spherical and prismatic. The first two are the most used for representing complex structures such as robots and the human body. To simulate the Agent's sensors, the environment (which may be pre-defined or dynamic) is mapped into proprioceptive and visual data (the Agent's pose).

Control Cycle (Simulation Step)

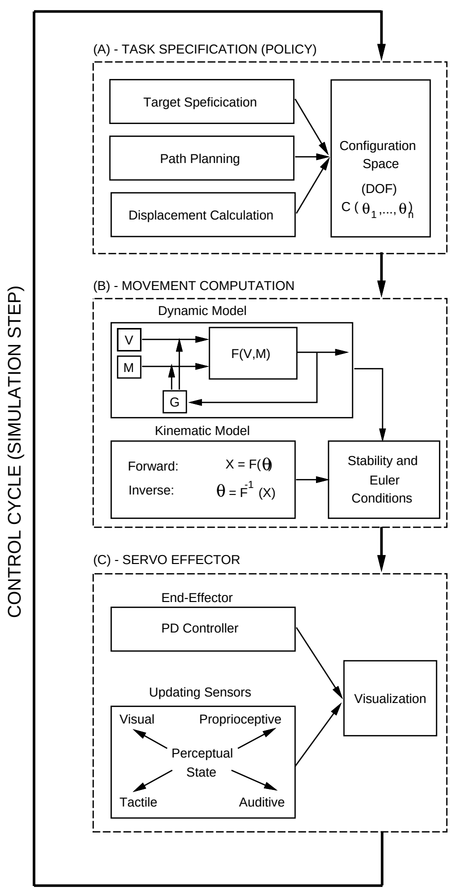

Diagram: a high-level description of the architecture. *(A)* - Based on a task dependent policy, the Agent defines the target and computes the best path over the configuration space (C-space), i.e. solving a harmonic function. Differential displacements in C-space coordinates are also computed here. *(B)* - Movement computation involves using dynamic and kinematic models. *(C)* - Based on the motion parameters computed in *(B)*, the effectors simulate the physical movements and the Agent's sensors are updated. The Agent's pose and perceptual state is then reported to the visualizing interface.

Applications: (top) "Roger the Crab", a (visual and haptics) Robot simulator, performing an inspection task; (bottom) "The Worm", an agent that supports a generic structure (links and joints in any configuration), performing a ground progression.