9-1-1995

# Simulation and analysis of complex human tasks

Norman I. Badler
*University of Pennsylvania*, badler@seas.upenn.edu

Welton Becket
*University of Pennsylvania*

Bonnie L. Webber
*University of Pennsylvania*, bonnie@inf.ed.ac.uk



Postprint version. Presented at *SPIE Workshop on Modeling, Simulation, and Control Technologies for Manufacturing*, September 1995, 9 pages.

# Simulation and analysis of complex human tasks

**Abstract**
We discuss how the combination of a realistic human figure with a high-level behavioral control interface allow the construction of detailed simulations of humans performing manual tasks from which inferences about human performance requirements can be made. The Jack human modeling environment facilitates the real-time simulation of humans performing sequences of tasks such as walking, lifting, reaching, and grasping in a complex simulated environment. Analysis capabilities include strength, reachability, and visibility; moreover results from these tests can affect an unfolding simulation.

**Keywords**
human factors, computer graphics, virtual environments, virtual reality, simulation, human modeling, task modeling, human performance

**Comments**
Postprint version. Presented at *SPIE Workshop on Modeling, Simulation, and Control Technologies for Manufacturing*, September 1995, 9 pages.

# Simulation and analysis of complex human tasks

Norman I. Badler

Welton M. Becket

Bonnie L. Webber

University of Pennsylvania
Philadelphia, PA 19104-6389

## ABSTRACT

We discuss how the combination of a realistic human figure with a high-level behavioral control interface allow the construction of detailed simulations of humans performing manual tasks from which inferences about human performance requirements can be made. The *Jack* human modeling environment facilitates the real-time simulation of humans performing sequences of tasks such as walking, lifting, reaching, and grasping in a complex simulated environment. Analysis capabilities include strength, reachability, and visibility; moreover results from these tests can affect an unfolding simulation.

**Keywords:** Human factors, computer graphics, virtual environments, virtual reality, simulation, human modeling, task modeling, human performance

## 1 INTRODUCTION

Present technology lets us approach human appearance and motion through computer graphics modeling and three-dimensional animation. We seek not to replace humans, but to substitute adequate computational surrogates in various situations in virtual prototyping or simulation-based design that would be otherwise unsafe, impossible, or too expensive for testing with live subjects.

Our goals are to build computational models of human-like figures which manifest animacy and believable behavior. Towards this end, we

- Create an interactive computer graphics human model.
- Endow it with reasonable biomechanical properties.
- Provide it with "human-like" behaviors.
- Use this simulated figure as an agent to perceive and effect changes in its world.
- Describe and guide its tasks through instructions, commands, warnings, policy, etc., in a natural way.

- Permit interaction, both physical and communicative, with other agents.

In addition, a computational agent can be measured to provide quantitative data on the execution of a given task, for example, what was or was not reachable or seen, what steps required excessive exertion, and so on.

The graphical generation of animation requires some control activities on the part of a user or animator. We distinguish three sorts of user-oriented control schemes:

1. Manipulate the human agent directly to position and move it as desired.
2. Show the human agent what to do, so it it moves by imitation.
3. Tell the human agent what to do, so that it moves by understanding.

The first technique involves more-or-less a direct manipulation paradigm, where a user can "grab" (on-screen) hands, feet, center of mass, eye gaze direction, etc., and move it directly via interactive cursor. This is the principal mode of interaction with many human modeling systems, such as our *Jack*®system,[4] which embodies the direct manipulation paradigm. The transformation of such inputs often require considerable computational work, such as inverse kinematics, in order to establish joint angles not directly specified, maintain agent balance, or take steps if moved in the ground plane.[4]

The second technique falls into the "Virtual Environment" category. Here a synthetic human body is associated with a live participant through appropriate sensors.[2] the corresponding parts of the synthetic agent are made to move in synchrony with the live subject. This approach is generally called "performance animation" since it involves a "performer" and a virtual surrogate or "avatar."

The third technique addresses the delivery of instructions to a human agent. It was mentioned above as one of our major goals. As people primarily engage in this mode of motion specification via language, it has become important to understand how people understand instructions[6,4,26] and how those instructions are carried out. In particular, instructions fail to specify some of the details of movement, so the agent's own "intelligence" must somehow fill in the missing actions in the given context. The agent must have sufficient "skills" to transform sensory inputs, desires, and intention into embodied action.

## 1.1 The Animation Spectrum

The animation "spectrum" has great bearing on the user interface to an animated human. At one end of the spectrum is *crafted animation*, which produces high quality imagery, requires highly skilled animators, attempts to entertain, excite, and perform the impossible, aims for a large (and paying) viewer market (e.g. movie-going public), and is characterized by long production time and high production cost. At the other end is *expedient animation*, which produces medium quality imagery, is created by an engineer or computer user, aims for a small viewer market (such as an individual, the management, the customer), attempts to answer specific "what if" questions to show possibilities, and is characterized by short (and preferably real-time) production time and low production cost. At this end are also a variety of real-time virtual or simulation environments which are not even "animated" in the conventional sense, but are created by the direct actions of the participant. These real-time control environments are the future of systems as diverse as battlefield simulation, driving simulators, real-time games, and live virtual environments of any sort. Our systems are decidedly in the *expedient* category.

## 1.2 The Human Factors Domain

One particularly well-motivated application for real-time human models is human factors analysis. Visualizing the appearance, capabilities and performance of humans is an important and demanding application. Building software for human factors applications serves a widespread, non-animator user population. Our software design has tried to take into account a wide variety of engineering design-oriented tasks, rather than just offer a computer graphics and animation tool for the already skilled or computer-sophisticated animator.

The embodiment of our choices for human modeling is a software system called *Jack*. The *Jack* human model supports anthropometric scaling, mass, and joint-limit information for given populations. The default figure has 68 joints and 134 degrees of freedom. *Jack* also provides whole-body, real-time, inverse-kinematics for goal-oriented specification of body movements.[4] A reactive walking model allows realistic portrayal of human navigation between workspaces, and a human grasping model combined with a fast strength-considerate motion planning facility allows automatic complex reaching and grasping in constrained spaces.[11] These automated tasks can be developed interactively in the *Jack* environment or using a task-network language, Parallel Transition Networks (PaT-Nets) which allows hierarchical, reusable, network-based descriptions of tasks. For a Virtual Reality experience, *Jack* can be configured with a digitizing glove and a 3D magnetic body tracking device permitting the user to visualize and move his or her entire body in the virtual environment.

One can think of *Jack* as an experimental environment in which a number of useful general variables may be readily created, adjusted, or controlled: the workplace, the task, the human agent(s) and some responses of the workplace to internally or externally controlled actions. The results of specific instances of these input parameters are reported through computer graphics displays, textual information, and animations. Thus the field of view of a $95^{th}$ percentile male while leaning over backwards in a tractor seat as far a possible may be directly visualized through the graphic display. If the figure is supposed to watch the corner of the bulldozer blade as it moves through its allowed motion, the human figure's gaze will follow in direct animation of the view.

The *Jack* software is built on Silicon Graphics workstations because those systems have the 3-D graphics features that greatly aid the process of interacting with highly articulated figures such as the human body. Our program design has tried to take into account a wide variety of physical problem-oriented tasks, rather than just offer a computer graphics and animation tool for the already computer-sophisticated or skilled animator.

# 2 TASK ANALYSES

*Jack* features computation of some of the most commonly performed task analyses:

- Multiple position and orientation (reach) goals.
- Static (postural) balance.
- Viewing cones and one or both eye views.
- Static or dynamic strength, torques, or force.
- NIOSH lifting index.
- Collision detection and avoidance.
- Showing the effect of population anthropometry on a particular goal configuration.
- Portraying the results of an external task simulation.

The last category admits customized analyses through an Application Program Interface (API); we have experimented with computing fatigue, heat, and radiation exposure.

## 2.1 Eye view

*Jack* can show the view from any object, in particular, a figure's eye. Translucent "view cones" may be displayed from the eyes of a human figure. With the apex at the eye lens center, the shape of the cones follows any desired polygonal path, e.g. foveal area. By aiming the eyes with an interactive goal, the view cones follow the point of interest, converging or diverging as needed (subject to eye "joint" limits). Since the cones are translucent, workplace objects show though, giving the user a good impression of what can and cannot be seen by the subject. When connected to an immersive VR imaging device, the user can experience the same view as the *Jack* figure, interactively and simultaneously linked with multiple goal satisfaction, anthropometric re-scaling, etc.

## 2.2 Static and dynamic torque, strength, and comfort

Interactive graphic displays of joint torque or end-effector forces may be shown in *Jack* as the user manipulates the figure.[27] Torques along a joint chain may be shown, too. With or without additional loading on the body segments or end-effectors, *Jack* can compute and graphically display the static or dynamic reaction forces generated anywhere else in the body.[4] These torques may be graphically presented as color coded regions on a contour body, instantly showing overloaded joints in red and safe loads in shades from white (no load) to blue (maximum load). The torque loads may be directly compared to joint strength limits measured by Abhilash Pandya at NASA Johnson Space Center in a small (14 subject) population sample.[19]

A more general test implements the NIOSH lifting index.[25] The user specifies the starting and ending posture on a selected human figure. Sliders allow interactive experimentation with the weight to be moved, the number of repetitions, and the work duration. The NIOSH formula is evaluated and a message regarding the suitability of this motion for the figure is printed.

## 2.3 Interactive body sizing under active constraints

By changing the body dimensions through the Spreadsheet Anthropometry Scaling System, a given task analysis can be repeated for different bodies with no additional setup overhead. The goals are merely re-evaluated for the new scaled body. For example, suppose the eye is constrained to the design eye point of a cockpit, the hands and feet are positioned to appropriate goals, and the shoulders and hips are restrained by point goals representing a suitable restraint system. Then running through the a variety of representative body dimensions with reach goals for the hands, feet, and hips will show how well or how poorly the population can carry out that task.

# 3 CONNECTING WITH EXTERNAL SIMULATIONS

The ultimate analysis tool is a simulation which executes some task and drives the human figure with a set of goals and timings. In order to drive *Jack* through a program rather than the interactive user interface, two Application Program Interfaces (API) have been developed: one in Lisp[7] and one in *C*. Any *Jack* command can be issued from a command in the Lisp API; also, almost any data item can be retrieved through the Lisp API

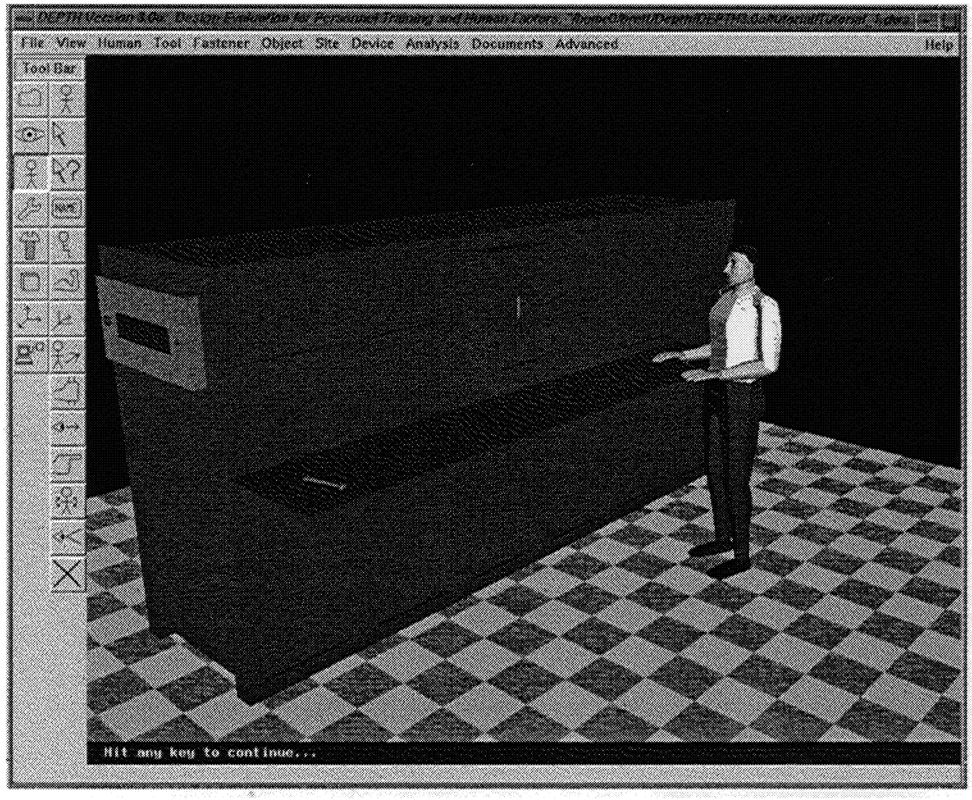


Figure 1: A DEPTH screen showing *Jack* next to an equipment bay.

providing a crucial information link for the external simulation. Lisp provides an interpreted, real-time, parallel instruction stream for the *Jack* simulation. A reduced capability API has recently been written in C to facilitate embedding *Jack* in other systems.

We are involved in several projects that use various external simulations:

- The DEPTH system is an Air Force project in collaboration with Hughes Missile Systems, which seeks to combine a number of human factors tools into a aircraft maintenance evaluation workstation suite. (See Fig. 1.) The external simulation used in the prototype implementation is HOS from Micro Analysis and Design, Inc.[15] Tasks designed in the HOS network can be used to compose *Jack* commands which are sent through the API to drive basic *Jack* behaviors such as reach and locomote. Task completion signals from *Jack* are returned through the API to control task sequencing in HOS.
- Sandia National Lab is implementing real-time avatars based on *Jack* bodys and positioning algorithms.[24,3] Body positions are input from magnetic sensors (see Section 4 below), packaged, and shipped into *Jack* through the API. Joint angles computed by *Jack* are returned through the API to drive other image generation software.
- Our own testbed is based on an object-oriented, reactive planner that uses "societies of behaviors" to mediate the actions of multiple agents.[4,8] For example, the locomotion behavior may be controlled incrementally through the simulation so that the figure may pursue interactively changing targets while simultaneously avoiding obstacles in the direct path.[14] The simulator is reactive in the sense that the collision-free paths do not need to be precomputed; rather, the next step is based on the local configuration of objects to be approached or avoided.

# 4 VIRTUAL HUMANS

By using a number of 6 degree of freedom sensors from Ascension Technology Corp., motion of the entire human body may be controlled interactively.[2] The sensors are placed on the wrists (or palms), the small of the back, and the forehead. The two hand sensors provide position and orientation information for the principal end effectors. The back sensor approximates the location of the center of mass and provides locomotion and stepping information as well as pelvis orientation. The forehead sensor supplies gaze direction. These inputs directly control the nearly full range of *Jack* behaviors. Hand gestures are input through readily available hand pose sensing devices such as a Cyberglove. The result is a virtual human controlled by a minimally encumbered operator. Such an arrangement has been used to great advantage by Deere and Company: by instrumenting an actual operator they can test for operator reach, visibility, and comfort in a simulated vehicle before any hard prototype is constructed.[22]

# 5 HIGHER-LEVEL CONTROL

Recent trends in computer animation have been directed toward procedural techniques and performance capture. In procedural techniques, many of which are embedded in the *Jack* system as well as other commercial systems, algorithmic motion generation is used to aid an animator in interpreting human-scale specifications and parameters as frame-to-frame motion.[1,4,10,16,18,20,23,29,30] Typical examples are key-parameter motion, inverse kinematics, dynamic simulation, or parameterized walking. While these and other techniques vary in the amount of information that the animator must provide – e.g. key poses for key-framing, end effector goals for inverse kinematics, forces for dynamics, or stylistic parameters for walking – they share the characteristic that they only model a portion of the overall complex of real human motion. Context such as human capabilities, emotion, experience, comfort, perception, and decision-making are ignored or suppressed.

This dissatisfaction with procedural methods often results in a move to live motion capture by means of magnetic or optical sensors.[2,21,28] Once an individual and his or her motion is recorded, the joint angles and spatial paths can be used as a guide for an animator or can themselves be used as the input to other procedural techniques such as inverse kinematics. The advantage to such an approach is the preservation of the nuances of movement and kinetics that would be difficult for an animator to intuit and design. The disadvantage is that the motion is tied to an actual body and an actual performance, and does not provide in itself the parameterizations needed to assess human factors over variable bodies and environments. In particular, performance-based animation does not generally solve the problem of creating realistic and appropriate human movements in any of the following situations:

- Highly-variable situations, where one cannot *a priori* acquire the range of movements that people will make. Such situations can arise in real-time simulation activities, where the situation and actions of other agents cannot be predicted ahead of time.
- Dangerous situations, where one cannot involve a human actor.
- Time-critical situations, where the amount of time needed to program behaviors excludes programming as an option.

Such situations require immediate, situationally appropriate (reactive) behavior on the part of a synthetic agent.

Under such conditions, a synthetic agent must be made to sense its environment and respond to it in terms of reflexes, higher level intentions, expectations, and available skills. An effective way of doing this is through the integration of a rich collection of interacting techniques, organized in a principled, structured representation. These techniques include planners and parallel transition networks to aid in overall task control, and goal-based sensing, response, and (as necessary) physics-based, kinematic or inverse kinematic behaviors to achieve environmentally-appropriate movements. Together they simplify the expression of local environmental influences without complicating their expression at the higher levels and the expression of situational awareness and influences at a higher level without the added complexity of managing all potential lower level variability.

An intelligent agent must interleave sensing, planning, decision-making, and acting. Accordingly, it is desirable to create an architecture that permits specification and exploration of each of these processes.[17,5,9] Planning and decision-making can be accommodated through incremental, symbolic-level reasoning. When the agent decides to act, the symbolic actions must be instantiated in executable behaviors. Most behavioral systems use either state controllers or numerical feedback streams, but not both. By using both it is possible to obtain maximum flexibility and maintain appropriate levels of specification.[4,8]

We can characterize these two control levels as PaT-Nets and "Sense-Control-Act" (SCA) loops.

- PaT-Nets are parallel state-machines that are easy for humans and automatic planning systems to manipulate. They are also good at sequencing actions based on the current state of the environment or of the system itself. They characterize the tasks in progress, conditions to be monitored, resources used, and any temporal synchronization.
- The SCA loop performs low-level, highly reactive control involving sensor feedback and motor control.

In this paradigm, the agent can instantiate explicit PaT-Nets to accomplish certain goals (*e.g.*, go to the supply depot and pick up a new motor), while low-level control can be mediated through direct sensing and action couplings in the SCA loop (*e.g.*, controlling where the agent's feet step and making sure that s/he doesn't run into or trip over any obstacles). Since the sensors can establish what the agent can perceive, the agent is able to react through the SCA loop, and if desired, use this information to confirm, adopt, or select higher-level (cognitive) actions: for example, whether to step over or around an obstacle, initiate or receive communications, etc. Since PaT-Net state transitions are explicitly represented, statistical distributions of alternative behaviors may be easily embedded. PaT-Nets may instantiate themselves when certain events or conditions occur,[12,13] or they may be invoked or invented by high-level planning processes.[26] They depend on the required action and the agent's immediate intentions.

We believe that ongoing research into embodied human-like simulated agents will find, as we have, that this architecture of high level schemas and planners combined with low level SCA loops will achieve increasing success in producing intelligent and realistic behavior.

# 6 CONCLUSION

We view the *Jack* system as the basis of a virtual animated agent that can carry out tasks and instructions in a simulated 3D environment. A significant number of production installations of the *Jack* software are engaged in the direct human factors analysis of prototype designs. Even though *Jack* is under continual development, it has nonetheless already proved to be a substantial computational tool in analyzing human abilities in physical workplaces. It is being applied to actual problems involving space vehicle inhabitants, helicopter pilots, maintenance technicians, foot soldiers, and tractor drivers. This broad range of applications is precisely the target we intended to reach. The general capabilities embedded in *Jack* attempt to mirror certain aspects of human performance, rather than the specific requirements of the corresponding workplace.

# 7 ACKNOWLEDGMENTS

This research is partially supported by DMSO DAAH04-94-G-0402; ARPA DAMD17-94-J-4486; U.S. Air Force DEPTH through Hughes Missile Systems F33615-91-C-0001; ARO DURIP DAAH04-95-1-0023; Army AASERT DAAH04-94-G-0220; ARPA AASERT DAAH04-94-G-0362; NSF IRI91-17110 and IRI95-04372; and JustSystem Japan.

# 8 REFERENCES

[1] N. I. Badler, B. A. Barsky and D. Zeltzer (editors). *Making Them Move: Mechanics, Control, and Animation of Articulated Figures*, Morgan-Kaufmann, San Mateo, 1991.

[2] N. I. Badler, M. Hollick and J. Granieri. "Real-time control of a virtual human using minimal sensors," *Presence* 2(1), pp. 82-86, 1993.

[3] N. Badler, D. Metaxas, B. Webber, and M. Steedman. "The Center for Human Modeling and Simulation," *Presence, the Journal of Virtual Reality and Teleoperators* 4(1), pp. 81-96, 1995.

[4] N. I. Badler, C. B. Phillips and B. L. Webber. *Simulating Humans: Computer Graphics, Animation and Control*, Oxford University Press, New York, 1993.

[5] N. Badler, B. Webber, W. Becket, C. Geib, M. Moore, C. Pelachaud, B. Reich, and M. Stone. "Planning for animation," to appear in N. M-Thalmann and D. Thalmann (eds.), *Interactive Computer Animation*, Prentice-Hall, 1995.

[6] N. Badler, B. Webber, J. Kalita and J. Esakov. "Animation from instructions," in N. Badler, B. Barsky, & D. Zeltzer (eds.), *Making Them Move: Mechanics, Control, and Animation of Articulated Figures*, Morgan-Kaufmann, San Mateo, pp. 51-93, 1991.

[7] W. M. Becket. "The *Jack* Lisp API," *University of Pennsylvania Technical Report MS-CIS-94-01* 1994.

[8] W. M. Becket and N. I. Badler. "Integrated behavioral agent architecture," *Proc. Workshop on Computer Generated Forces and Behavior Representation*, Orlando, pp. 57-68, March 1993.

[9] J. Cassell, C. Pelachaud, N. Badler, M. Steedman, B. Achorn, W. Becket, B. Douville, S. Prevost and M. Stone. "Animated conversation: Rule-based generation of facial expression, gesture and spoken intonation for multiple conversational agents," *Computer Graphics*, Annual Conference Series, pp. 413-420, 1994.

[10] M. F. Cohen. "Interactive spacetime control for animation." *Computer Graphics* 26(2), pp. 293-302, 1992.

[11] B. J. Douville, X. Zhao and N. Badler. "Implementation of simulated human grasping for manufacturing tasks," *Proc. Intelligent Systems and Advanced Manufacturing Symposium*, SPIE 2596-30, 1995.

[12] R. J. Firby. "Building symbolic primitives with continuous control routines," In *Artificial Intelligence Planning Systems*, 1992.

[13] M. P. Georgeff and A. L. Lansky. "Reactive reasoning and planning," In *Readings in Planning*, J. Allen, J. Hendler, and A. Tate (eds.), Morgan Kaufmann, San Mateo, pp. 729-734, 1990.

[14] H. Ko, B. Reich, W. Becket and N. Badler. "Terrain navigation skills and reasoning," *Proc. Computer Animation*, Geneva, 1994.

[15] K. R. Laughery. "Micro SAINT – A tool for modeling human performance in systems." *Applications of Human Performance Models to System Design*, G. R. McMillan, D. Beevis, E. Salas, M. H. Strub, R. Sutton and L. Van Breda (eds.), Plenum Press, New York, 1989.

[16] D. Liu, S. J. Gortler and M. F. Cohen. "Hierarchical spacetime control," *Computer Graphics*, Annual Conference Series, pp. 35-42, 1994.

[17] M. B. Moore, C. W. Geib and B. D. Reich. "Planning and terrain reasoning," in *AAAI Spring Symposium on Integrated Planning Applications*, 1995. *University of Pennsylvania Technical Report MS-CIS-94-63.)*

[18] J. T. Ngo and J. Marks. "Spacetime constraints revisited," *Computer Graphics*, Annual Conference Series, pp. 343-350, 1993.

[19] A. Pandya, J. Maida, A. Aldridge, S. Hasson, and B. Woolford. "Development of an empirically based dynamic biomechanical strength model," in *Space Operations Applications and Research Conf. Proc.*, Vol. 2, pp. 438–444, 1991.

[20] M. van de Panne and E. Fiume. "Sensor-actuator networks," *Computer Graphics*, Annual Conference Series, pp. 335-342, 1993.

[21] B. Robertson. "Caught in the act," *Computer Graphics World* 17(9), pp. 23-28, 1994.

[22] SAE International. *Human modeling technology and applications.* Video tape SP-1100, Warrendale, PA, 1994.

[23] K. Sims. "Evolving virtual creatures," *Computer Graphics*, Annual Conference Series, pp. 15-22, 1994.

[24] S. A. Stansfield. "Distributed virtual reality simulation system for situational training," *Presence: Teleoperators and Virtual Environments* 3(4), pp. 360-366, 1994.

[25] T. R. Walters, V. Putz-Anderson and A. Garg. *Applications Manual for the Revised Lifting Equation*, US Department of Health and Human Services, January 1994.

[26] B. Webber, N. Badler, B. Di Eugenio, C. Geib, L. Levison and M. Moore. "Instructions, intentions and expectations," *Artificial Intelligence Journal* 73, pp. 253-269, 1995.

[27] S. Wei and N. Badler. "Graphical displays of human strength data," *Visualization and Computer Animation* 3(1), pp. 13-22, 1992.

[28] L. Williams. "Performance-driven animation," *Computer Graphics* 24(4), pp. 235-242, 1990.

[29] A. Witkin and M. Kass. "Spacetime constraints," *Computer Graphics* 22(4), pp. 159-168, 1988.

[30] J. Zhao and N. Badler. "Inverse kinematics positioning using nonlinear programming for highly articulated figures," *ACM Transactions on Graphics* 13(4), pp. 313-336, 1994.