

Dual Stewart Platform Mobility Simulator

R.F. Boian, M. Bouzit, G.C. Burdea, J. Lewis and J.E. Deutsch

Abstract— The paper presents the simulation and modeling of gait on a system using two prototype compact Stewart platforms. Control issues raised by the reduced size of the robots and hence their lower power output are discussed. The flexible haptic material concept is introduced and used to realistically model heterogeneous haptic surfaces simulating the virtual ground. An algorithm for virtual foot/virtual surface contact modeling and transforming the platform motion into walking in VEs is also presented.

I. INTRODUCTION

GAIT simulators have long been a field of interest for researchers. The integration of such a system with a virtual environment may, in theory, expand the range of applications to entertainment and real-life task training of patients with walking dysfunction. The appealing reasons for using such systems are the flexibility and transparent data collection offered by virtual environments over real environments [9][14]. In addition, the ability to train for a task in a controlled environment, away from the potential hazards of a real environment makes such simulators viable choices in medical or military applications.

A gait simulator integrated with a VR simulation has to accomplish two main tasks: map the user's motion into virtual environment navigation and calculate the haptic feedback to be applied to the user's feet or legs as a result of the interactions in the virtual world. The extent to which these tasks are implemented depends mainly on the limitations imposed by the simulator system.

In most situations, the applicability and success of a virtual reality simulation is conditioned by the degree of video and audio feedback. For physical-based simulations, the realism is also defined by how close the haptic feedback feels compared to the real life experience. In the case of gait simulators, the haptic feedback is calculated from the interaction between a virtual avatar and a virtual surface. It is then applied primarily to the feet and through them to the whole body. The ability to render various surface shapes is

essential for enhancing the user experience [11].

The treadmill is the most widespread device used for gait training. It is used for training at various speeds on a straight flat surface or small incline, but cannot be used to simulate real terrain characteristics such as curves, uneven surfaces, or properties such as stiffness or friction. Several research groups have focused on developing advanced and flexible simulators that would provide a more life-like experience to the user. Such examples are the Sarcos Biport and the Sarcos Treadport developed by Hollerbach et al [6], the ATR-GSS treadmill developed by Miyasato [10], Iwata's GaitMaster and Torus Treadmill [7][8], the CAREN System (E-Motek Inc.), or the HapticWalker [13].

Most of the projects mentioned above focus on simulating stairs, slopes or uneven surfaces. Besides its shape, a real surface is always characterized by physical properties, which sometimes are more challenging to reproduce than the shape of the surface. For instance, the flat smooth surface of a sidewalk is not easy to negotiate when it is covered with ice. In general, foot-platform systems [2][6][8] have the ability to apply additional forces to the user's feet besides those necessary to simulate the surface shape. Depending on the degrees of freedom offered by the system, and the available control/mechanical bandwidth, such forces could be used to simulate the physical properties of a surface.

The research presented here is based on the Mobility Simulator co-developed by the Human-Machine Interface Laboratory (Rutgers University) and the RIVERS Laboratory (University of Medicine and Dentistry of New Jersey). The simulator uses two prototype Stewart platforms that can control positions and forces to each foot in 6DOF. Section 2 gives a brief overview of the system hardware and software. Section 3 discusses the robot servo control and associated stability, power and accuracy issues. Section 4 details the haptic rendering process. Section 5 presents the results of an initial validation study. Concluding remarks and directions for future research are given in Section 6.

II. THE MOBILITY SIMULATOR

The main hardware components of the Mobility Simulator are two 6DOF prototype Rutgers Mega-Ankle (RMA) pneumatic robots (see Figure 1). Each robot has a Stewart platform configuration with six dual-acting pneumatic pistons. Each piston is mounted in parallel with a linear potentiometer, which provides information of the smaller mobile platform position vs. the larger fixed one. The mobile platform has a foot attachment plate and a 6DOF

This work was supported by grant BES-0201687 and a Research Experience for Undergraduates supplement from the National Science Foundation.

R.F. Boian, M. Bouzit, and G.C. Burdea are with the CAIP Center, RUTGERS University, Piscataway, NJ, USA (e-mail: {boian,bouzit,burdea}@caip.rutgers.edu).

J. Lewis and J.E. Deutsch are with the RIVERS Lab, School of Health Related Professions, UMDNJ, Newark, USA (e-mail: {deutsch, lewisje}@umdnj.edu).

Author for correspondence Greg Burdea (phone: 732-445-5309; fax: 732-445-4775)

force sensor. Force data are used when the Rutgers Mega-Ankle robot is in force control mode. The Mobility Simulator also incorporates an electro-pneumatic control interface, an un-weighting frame (Biodex Co.), a graphical workstation and a large screen custom display. The design of the simulator fits in the "foot platforms" category defined by Hollerbach [6]. The user stands with each foot secured with straps to the top of a platform, while the VR simulation [2] is displayed on the large screen facing the user. To improve performance and safety, the user's body is secured in a harness and connected to the un-weighting frame placed above the two platforms. Handlebars mounted on the frame's posts are available for upper extremity support. In this setup, the user can walk on top of the RMA platforms with small steps limited by the platform's workspace [3].



Figure 1. The Rutgers Mobility Simulator. © Rutgers University and UMDNJ. Reprinted by permission.

During walking, each foot either supports the weight of the body or swings freely while taking a new step. Accordingly, each RMA platform can function in load compensation mode or in free motion mode. In free motion mode the RMA platform follows the swinging foot and compensates for its own mobile platform weight. During this mode, the platform applies very low to zero forces to the foot. In load compensation mode, the platform holds the supporting foot weight and slides backward simulating the behavior of a treadmill. In both modes, the robots can apply additional 6DOF forces or vibrations to the user's foot as commanded by the simulation running on the workstation. The two functioning modes mentioned above are a subset of the actual implementation, sufficient for the purpose of this paper. A more detailed description of the RMA platform's functioning modes can be found in [3].

The simulator software is distributed on two computers: the graphics workstation and the electro-pneumatic control

interface (an embedded PC) as illustrated in Figure 2. The graphics workstation handles the graphic and haptic rendering. Based on the information received from the control interface, it calculates the interaction between the virtual feet and the virtual environment and sends back to the control interface commands regarding the forces to be applied or the functioning mode to be used.

The control interface handles the low-level servo control of the two robots, and provides the simulator with the position of each foot calculated through forward kinematics. For performance purposes, the control interface also takes care of switching between certain functioning modes based on the forces applied by the foot as measured by the Rutgers Mega-Ankle 6DOF force sensor. The current functioning mode of each RMA platform is also sent to the graphics workstation. A more detailed presentation of the tasks executed by the control interface can be found in [3].

III. THE RUTGERS MEGA-ANKLE SERVO CONTROL

The servo controller design is split over three loops represented as shaded areas in Figure 2. From top to bottom, the loops are: task control loop, dynamics loop, and pressure loop.

The task control loop is responsible for processing the commands coming from the VR simulation and converting them into Stewart platform desired position, velocity, acceleration and forces. The commands coming from the simulation specify the functioning mode to be used (i.e. weight support mode or free motion mode), and haptic effects and forces to be applied to the users feet. This loop is executed alternatively for each platform. The task controller converts the simulation command using as additional input the measured positions and forces, read by the linear potentiometers and the force sensor of each robot.

The dynamics loop transforms the desired Cartesian positions and forces into actuator level forces. The inverse dynamics implementation takes into account the current state of the robot, including position, velocity, acceleration and external forces. The measure external forces T_{mes} are added with the desired forces T_{des} calculated by the task control loop, hence the final force being applied will be the force necessary to counteract the user's action added with the desired force to be applied to the user's foot.

The actuators used by the RMA robots are dual-acting pneumatic pistons, hence the force applied by the cylinder can be controlled by adjusting the air pressure in the upper and lower chambers. The actuator position control transforms the desired change in position into a desired force. The calculated cylinder lengths L_{des} are added with the measured length L_{mes} , and the resulting length difference L_{err} is transformed into a force through the $L \rightarrow F$ control block. This control uses a proportional derivative (PD) strategy.

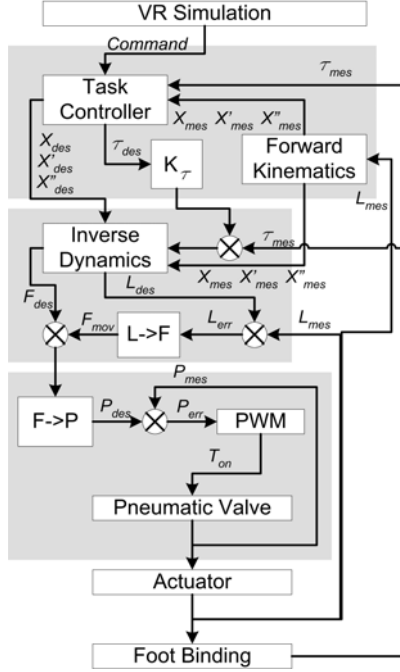


Figure 2. Controller architecture. © Rutgers University and UMDNJ. Reprinted by permission.

Finally, the pressure loop takes the desired actuator force, transforms it into upper and lower chamber pressures. The desired pressures are controlled in Pulse Width Modulation (PWM). The PWM strategy transforms the desired change in pressure P_{err} into the time interval T_{on} during which the valve should be kept open.

$F \rightarrow P$ is a direct transformation of actuator force into upper and lower chamber air pressures designed to minimize their change. Minimizing the amount the air pressure change improves the system response time. In cases when the desired force cannot be achieved through minimization, the pressures are determined so that they are balanced around the middle of the controllable pressure range. The PWM is implemented using an adaptive strategy. The change in pressure in a cylinder chamber is not linear in time, and the shape of the curve depends on the volume of the actuated cylinder chamber. Another factor taken into consideration is the intake airflow, which is affected by the number of actuator chambers accepting or exhausting air at a given moment in time. The PWM duty cycle is computed taking into consideration all these factors using the equation:

$$T_{on} = kP_{err} (1 + k_v V_{air}) (1 + k_N N_{chamber}) (1 + k_F F_{flow})$$

where V_{air} is the volume of uncompressed air necessary to achieve the change in pressure, $N_{chamber}$ is the number of chambers accepting or exhausting air simultaneously F_{flow} is the airflow curve over time.

A. Simulator Mechanical Bandwidth

The mechanical bandwidth of the mobility simulator was measured for translations in the horizontal plane and

rotations around the front/back axis. The motions are the most commonly used by our system to render haptic effects as discussed in the last section of the paper. The measurements were done with both RMA robots active simultaneously. While the bandwidth of each individual robot is higher when measured separately, when both robots are active the intake airflow is reduced, thus the bandwidth is lowered. The results are presented in Table 1.

Table 1. RMA robot mechanical bandwidth (both platforms active).

X translation 0.1m amplitude	Y translation 0.18m amplitude	Y rotation. 5 deg amplitude
1.56 Hz	1.79 Hz	1.5 Hz

B. Robot Stability in Foot Support Mode

One of the first problems encountered during the development of the system was the stability of the RMA platforms under load. The robots were stable when subjected to external forces if there was no load attached to them. However, the working regime for which they were developed, involves resisting forces while supporting the weight of the user. Figure 3 shows the response of the robot to sinusoidal input. Under a 50 lbs load, the motion was distorted and the amplitude of the robot increased slowly eventually becoming unstable.

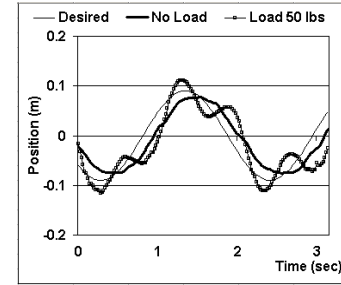


Figure 3. RMA platform response to a sinusoid input along the Y axis (back-front) with 0.5Hz frequency and 0.09m amplitude. © Rutgers University and UMDNJ. Reprinted by permission.

The cause of this problem was the addition of the desired cylinder force F_{des} with the cylinder motion force F_{mov} . Under load, the resulting forces were too high and caused the robot to become unstable. Increasing the derivative gain slowed down the instability but it didn't solve it. The solution was to reduce the proportional gains by a minimum of 42%. With lower gains, the platform was stable under load, but had very little power to move the user's foot backward during the foot support phase, hence making the system unusable. Also, when the system was unloaded, the steady state error was significantly larger. The use of an integrator term was avoided because it consistently caused a windup.

Two adaptive gains were used to bring the robot to respond properly under load. The gains added a fraction of the measured cylinder load to the proportional and derivative gains respectively. The proportional adaptive component helped increase the moving force of the platform

when under load, while the derivative adaptive component was increased to compensate for the high proportional gains and insure the stability of the system.

Figure 4 presents the response of the platform to the same sinusoidal input signal, under a 50 lbs load, with and without the adaptive component added to the lowered proportional and derivative gains. While both the constant and the adaptive response were stable, the adaptive strategy provided the necessary power to move the load closer to the desired position, and reducing the error by approximately half. The adaptive gains did bring a side effect slightly visible in Figure 4; when the robot moves with higher velocity, the adaptive derivative gains increases the damping of the system slowing it down and the releasing it as the load in the robot shifts and the force is reduced. This can be seen as a change in the slope of the adaptive curve in Figure 4.

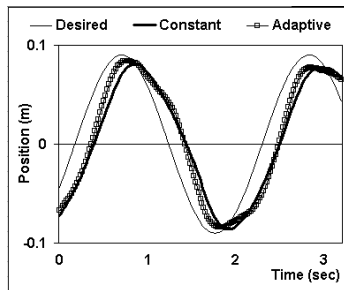


Figure 4. Response comparison lower constant gains with and without the adaptive component. © Rutgers University and UMDNJ.

C. Force Minimization in Free Motion Mode

The second major functioning mode of the RMA robot is following the foot during the swing phase of the gait. In this mode, the platform has to compensate for its own weight and for the forces applied by the user to the end-effector, hence not making itself felt to the foot. To achieve this, the servo controller disables the cylinder position control by canceling out the moving force F_{mov} , and by switching the measured forces signs by changing the value of K_t from 1 to -1. While these changes cause the RMA robot to follow the user's foot, the motion is very slow and large forces are felt at the foot. Figure 5 shows the forces measured at the foot during one swing phase.

The source of these large forces has been determined to be the damping created by the pneumatic actuators. While a K_t of 1 is sufficient to resist forces in foot support mode, in free mode, beside responding with a force to a the user's force, the robot also has to move, which involves a much large air intake/exhaust activity. The damping is mostly coming from the airflow limitations imposed by the small intake and exhaust sections of the cylinder chambers. To overcome this problem, K_t was increased in absolute value. The increase was done for each of the robot's 6 DOF. The Z-axis (up-down) translation gain was approximately four times larger than the rest of the gains, because the motion on that direction required all the cylinders to either intake or

exhaust, hence putting more airflow in a single direction. The measured forces for the increased K_t gains are shown in Figure 6. The forces are now reduced approximately 8 times to a maximum of 11N, comparable to a snow boot weight.

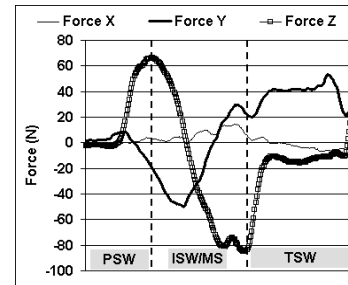


Figure 5. Free mode forces during swing for $K_t = -1$. (PSW = Preswing; ISW = Initial swing; MSW = Midswing; TSW = Terminal swing) © Rutgers University and UMDNJ. Reprinted by permission.

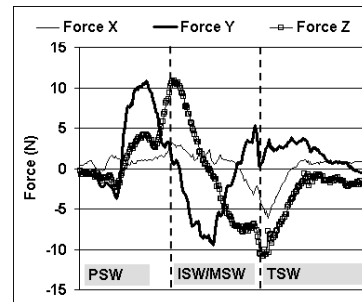


Figure 6. Free mode forces during swing for larger K_t . © Rutgers University and UMDNJ. Reprinted by permission.

IV. HAPTIC RENDERING FOR WALKING OVER VIRTUAL TERRAIN

The Mobility Simulator transforms the input position, force and functioning mode of each RMA platform into haptic feedback to the feet and visual update of the virtual scene using the virtual surface specifications. The virtual ground model is stored on the graphics workstation, as a polygonal mesh with areas characterized by haptic characteristics such as stiffness, damping, friction, haptic texture and breaking coefficient specified. The haptic output data consists of two sets of values specifying the functioning mode to be used by each platform, the 6DOF forces and the vibrations to be applied to the user's foot.

The processing necessary to calculate the graphics and haptic feedback can be divided into several stages that are executed at every simulation cycle (Figure 7). Only the swinging foot (free motion) is considered for the entire rendering process. The fixed foot (load compensation mode) is addressed only in the last stage of the process.

The process starts by reading the feet positions and functioning modes from the control interface. The functioning mode value is used to decide whether a foot should be moved or not. A foot in load compensation mode is kept fixed although the platform slides it backward.

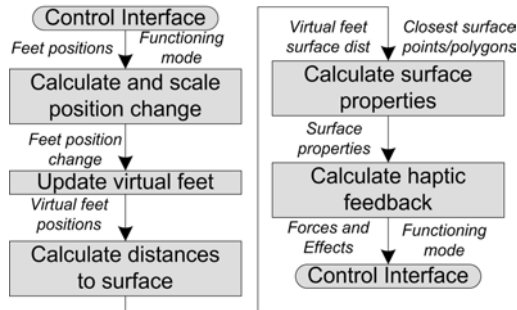


Figure 7. Haptic rendering stages © Rutgers University and UMDNJ. Reprinted by permission.

The next stage requires the calculation of the change in real foot position to be added to the virtual feet. Because the simulator's workspace cannot cover the entire range of motion of the legs, it was necessary to scale the change in each foot's position to increase the virtual walking velocity so that the simulation felt real. The scale is also applied to the vertical displacement making it possible to negotiate realistic virtual obstacles that are visually larger than the platform work envelope.

The next phase updates the positions of the virtual feet with the calculated change. The changes are applied in a frame of reference aligned with the virtual avatar's walking direction calculated in the previous cycle.

After the feet have been mapped into the virtual world, the viewpoint has to be moved accordingly. Positioning the virtual camera above the center of the segment defined by the two feet yields good results although it is not what really happens with the center of gravity of a walking person.

The next stage is calculating the distance between the foot and the surface. This stage will also find the closest surface points to the foot and their corresponding polygons. Similar to the real case, certain surface properties are manifested above the surface (e.g. mud or snow) hence the distance to the surface is relevant to the haptic feedback even if there is no collision with the underlying ground. Based on the results of the previous stage the simulator can find the surface properties around the closest points on the surface.

The last stage of the process is the calculation of forces and haptic effects to be applied to the user's foot, based on the surface properties and the depth of the collision.

A. Virtual Foot Modeling

The interaction between the virtual foot and the virtual ground surface is based on the haptic mesh concept developed by Popescu [12] as an extension to Ho's simpler haptic point concept [5]. The virtual foot implemented for the Mobility Simulator is modeled as a mesh of points positioned on the shoe sole. From a haptic point of view, the RMA platforms can only render forces in one point. The use of a mesh of points to calculate the interaction of the foot with the surface is necessary for realistic surface contact calculation.

The number of points in the mesh should be minimized because it is directly proportional to the amount of collision detection calculations, and it increases factorially the number of contact stability calculations. The minimum number of mesh points has been determined to be five. One point is positioned in the center of the mesh, while the rest are positioned on a rectangle around it. The dimensions of the rectangle match the shape of the end-effector foot attachment plate to which the user's foot is secured.

B. Ground Contact Evaluation

When a foot touches the haptic surface the swinging phase of the foot is over and the support phase is about to begin. The switch between these two phases is tightly connected to the functioning of the servo-controller, which has to be notified to start sliding the support foot backward.

The foot/surface interaction is evaluated by calculating the forces applied by the surface to each of the points in the virtual foot mesh. The vertical components of these forces are then compared with the vertical components of the forces applied by the user's foot to the simulator. The latter forces are calculated by transforming the output of the force sensor mounted on top of the RMA platforms to each point in the mesh. A point of the foot haptic mesh is "supported" if the resultant of these vertical components is pointing upward.

The three contact possibilities that can be differentiated based on the "supported" status of the haptic mesh points are presented in Figure 8. A stable contact occurs when at least three non-collinear points are supported.

The resultant of the haptic point forces is transmitted to the control interface to be applied to the swinging foot. If the swinging foot made a stable contact with the surface, the controller is notified to switch the functioning mode. The friction and surface textures are sent to the controller for both feet, regardless of their state.

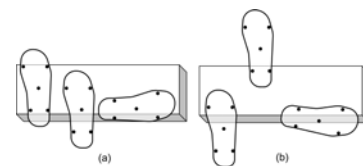


Figure 8. Foot/surface contact types: (a) stable, (b) unstable. © Rutgers University and UMDNJ. Reprinted by permission.

C. Low-level Haptic Effects

The mobility simulator system is designed to execute most of the haptic calculations on the workstation, and render the results on the Stewart platforms using a reduced set of basic level effects: apply a 6DOF force, change in position (positional jolt), and vibrations.

The simulation can request the controller to apply a certain force during both free or foot support functioning modes. If the request is made while in free motion mode, the given forces are scaled using positive values of the free

mode gains and then applied. The scaling is necessary to counteract the same damping behavior mentioned above. If a force request is made during the foot support mode, the given forces become essentially τ_{des} in Figure 2.

The changes in position are used as an alternative force requests during foot support mode. For instance, to simulate slipping on the ice, a lateral displacement is used instead of applying a lateral force. This is preferred because a known displacement is more controllable and can be adjusted to a comfortable level easier than applying a force which will have a different effect from a user to another, mainly due to differences in weight.

The vibrations are used to simulate surface textures. The frequency and amplitude of the vibrations are calculated on the graphical workstation and sent to the controller. The vibrations are rendered only as changes in orientation around the Y-axis (back-front direction) because it interferes the least with the rest of the foot measurements necessary to calculate the direction of motion, or intersection with the virtual surface.

V. INITIAL VALIDATION STUDY

A preliminary validation study was recently performed comparing gait on the mobility simulator with over-ground walking [4]. Three healthy subjects completed three walks at their self-selected speed across a GAIT-Rite mat and several walks of approximately 20 steps on the Mobility Simulator. Two of these consisting of walking on the simulator without the virtual environment present, two trials were done with the virtual environment present, and one trial with the virtual environment present, and the view point was stationary as the subjects walked. Kinematic data of Subject's trunk and lower extremities were captured using a digital camera and compared to data extracted from the RMA sensors. Averaged across all subjects, the velocity on the simulator without the virtual environment present was only 11.55% (.166m/sec) of their respective over-ground walking speed (1.45 m/sec). However, after displaying the street-crossing simulation, the velocity on the simulator was 15% (.191 m/sec) higher than walking without graphics feedback. Interestingly this increased in gait speed was achieved by increasing both step length and cadence. The greatest correspondence between the kinematics of over-ground walking and mobility simulator gait occurred at the knee. Given the preliminary nature of the findings the implications for training are still unclear.

VI. CONCLUSIONS AND FUTURE WORK

The paper presented control aspects of a mobility simulator using two Rutgers Mega Ankle robots and the modeling of the interaction between the virtual feet coupled to the RMA robots and a virtual surface. The preliminary validation study on healthy subjects showed that walking speed on the simulator was substantially slower than normal

over-ground walking. This is due to the limited work envelope of the system, but also due to sub-optimal control algorithms available at the time of the study. The control was subsequently optimized and the system installed in a clinical research laboratory at the University of Medicine and Dentistry of New Jersey. A larger validation study is underway and will be extended to individuals post-stroke. Training studies to rehabilitate the gait of individuals post-stroke are planned. Work still needs to be done on rendering smoother interaction forces and fidelity issues regarding the rendering of haptic textures need to be addressed. At this point it is unclear how properties of the system can be applied to individuals post-stroke. In principle the system is able to assist gait, which may be more beneficial for individuals in the acute-phase rehabilitation, and can resist gait, or induce disturbances, which may be more appropriate for rehabilitating gait of individuals with chronic strokes. Questions about manipulations of the virtual environment to augment training effects will also be addressed.

REFERENCES

- [1] R.F. Boian, J.E. Deutsch, C.S. Lee, G.C. Burdea, J. Lewis, "Haptic Effects for Virtual Reality-based Post-Stroke Rehabilitation", Proceedings of the Eleventh Symposium on Haptic Interfaces For Virtual Environment And Teleoperator Systems, Los Angeles, CA, March 2003, pp. 247-253.
- [2] R.F. Boian, G.C. Burdea, J.E. Deutsch, and S. H. Winter Street Crossing Using a Virtual Environment Mobility Simulator, Proceedings of IWVR 2004, Lausanne, Switzerland, pp. 27-33, 2004.
- [3] R. Boian, M. Bouzit, G. Burdea, and J.E. Deutsch Dual Stewart Platform Mobility Simulator, Proc. of IEEE EMBS 2004, Sept. 2004.
- [4] R.F. Boian, "Robotic Mobility Rehabilitation System Using Virtual Reality," Electrical and Computer Engineering, Rutgers University, January 2005.
- [5] Ho, C., Basdogan C., Srinivasan, M., (1997), "Haptic Rendering: Point- and Ray-Based Interactions," Proceedings of the Second PHANToM Users Group Workshop, Dedham, MA, Oct. 20-21.
- [6] Hollerbach, J.M., "Locomotion interfaces," in: Handbook of Virtual Environments Technology, K.M. Stanney, ed., Lawrence Erlbaum Associates, Inc., 2002, pp. 239-254
- [7] H. Iwata, Walking about virtual environments on infinite floor, In Proc. of IEEE Virtual Reality'99, Houston, TX, pp. 286-293, 1999.
- [8] H. Iwata, H. Yano, and F. Nakaizumi, Gait master: A versatile locomotion interface for uneven virtual terrain, in Proceedings of the IEEE VR2001 Conference, pp. 131-137, 2001.
- [9] D. Jaffe, D. Brown, C. Pierson-Carey, E. Buckley, H. Lew, "Stepping Over Obstacles to Improve Walking in Individuals with poststroke hemiplegia," Journal of Rehabilitation Research and Development," 41(3A), pp. 283-292, May/June 2004.
- [10] T. Miyasato. Tele-nursing system with realistic sensations using virtual locomotion interface. In Proceedings of the 6th ERCIM Workshop User Interfaces for All, Florence, Italy, October 2000.
- [11] H. Noma, T. Sugihara, T. Miyasato, "Development of Ground Surface Simulator for Trl-E-Merge System," Proceedings of IEEE Virtual Reality 2000 Conf., pp. 217-224, March 2000.
- [12] Popescu, V., G. Burdea, M. Bouzit, Virtual Reality Simulation Modeling for a Haptic Glove, Computer Animation'99, Geneva, Switzerland, pp. 195-200, May 26-28, 1999.
- [13] H. Schmidt, "HapticWalker - A novel haptic device for walking simulation", Proc. of EuroHaptics 2004, Munich, Germany, 2004.
- [14] H. Sveistrup, "Motor rehabilitation using virtual reality", Journal of NeuroEngineering and Rehabilitation 2004, 1:10. <http://www.jneuroengrehab.com/content/1/1/10>.