MechVR: A physics-based proxy for locomotion and interaction in a virtual environment

Victor Zordan
Clemson University

Saurabh Hindlekar
Clemson University

W. Garrett Mckay
Clemson University

John Welter
Clemson University

J. Emerson Smith III
Clemson University

Kunta Lowe
Clemson University

Carlos Marti
College of Charleston

R. Austin Taylor
College of Charleston

ABSTRACT

We present an immersive Virtual Reality (VR) experience developed through a unique combination of technologies including an actuated hardware rig; a physics model with a responsive control routine; and an interactive 3D gamelike experience. Specifically, this paper introduces a physics-based communication framework that allows force-driven interaction to be conveyed to a user through a physics-based proxy. Because the framework is generic and extendable, the application supports a variety of interaction modes, constrained by the limitations of the physical full-body haptic rig. To showcase the technology, we highlight the experience of riding locomoting robots and vehicles placed in an immersive VR setting.

CCS CONCEPTS

• **Computing methodologies** → Physical simulation; Virtual reality; • **Hardware** → Haptic devices; Emerging interfaces;

KEYWORDS

Virtual reality, Motion simulator, Physics-based animation

ACM Reference format:
Victor Zordan, Saurabh Hindlekar, W. Garrett Mckay, John Welter, J. Emerson Smith III, Kunta Lowe, Carlos Marti, and R. Austin Taylor. 2017. MechVR: A physics-based proxy for locomotion and interaction in a virtual environment. In *Proceedings of MiG '17, Barcelona, Spain, November 8–10, 2017,* 5 pages.
DOI: 10.1145/3136457.3136468

1 INTRODUCTION

In this paper, we propose a force-based interaction technique that acts as a communication form to allows individuals to "feel" the influence of entities in an immersive virtual environment. Our primary application is entertainment with a 3D gamelike VR experience being the goal. To this end, custom hardware (Simcraft [SimCraft 2017], Figure 1a) is employed as a generic rig to give haptic feedback while holding the human user/player (safely) inside. This real-world physical device is engaged through a simulated (game) world with which the player interacts, Figure 1c. A physical proxy modeling the cockpit/cabin inside of a virtual entity (Figure 1b), allows force interaction with a physical simulation to create feedback for the user. Along with a specialized control system, this physics-based proxy responds to the virtual forces applied to it by producing an appropriate disturbance and correction that express the force to the human seated inside. Along with a stereovision headset (Oculus [Oculus 2017]) and stereo sound, we are able to provide an immersive experience that is independent of the form/kinematics of the hardware. We develop examples of riding biped, quadruped, and wheeled vehicles using the same system.

At the core of the technique, a physical controller acts to respond and orient the proxy in the presence of forces, torques, and impulses. This controller abstracts away the dynamics of the VR entity (e.g. rover vehicle) and reduces the active components to basic orientation control. Translation is handled like many other VR experiences via visual cues. With our technique, we can then apply any number of forces and torques directly to the proxy, for example to simulate impact from a hit of enemy fire.

Employing this system, we create the effect of legged locomotion by producing appropriately timed and placed ground reaction forces. Because the base controller can generically respond to forces, a multi-leg locomotion (e.g. a trotting quadrepad) is synthesized through an appropriate combination of ground reaction forces for each step. To the extent that our hardware rig can move with the desired acceleration and within the required range, it can express the influence each leg through the haptic feedback to the player.

The novel contributions of this paper include: force-based interaction between a user/player and virtual entities employing a generic hardware rig; a general approach for force production and response that supports legged-locomotion gaits such as walking and bounding; and developed examples that show the benefit of haptic sensory feedback for increased immersion in a VR game. In addition, a key to immersion is the consistent combination of visual and movement cues, and our gamelike examples develop the application domain through design aimed to showcase the force-driven communication embedded in our core technology.

MiG '17, Barcelona, Spain

DOI: 10.1145/3136457.3136468

Figure 1: Conceptual layout: a) User provides input while housed in hardware full-body haptic rig. b) Physical proxy for player's avatar, translates input to movement. c) Proxy interacts with MechVR game environment

2 RELATED WORK

Creating the illusion of locomotion or self movement in VR is an important and challenging task [Mohler et al. 2007a; Multon and Olivier 2013; Steinicke et al. 2013]. And several interesting findings reveal our perceptual differences between real and virtual locomotion [Banton et al. 2005; Frenz et al. 2007; Janeh et al. 2017; Mohler et al. 2007b; Riecke and Wiener 2007; Steinicke et al. 2010]. However, the core of our technique is quite unique, as our goal is to create a flexible virtual avatar that interacts generically with a (virtual) physical world. In this regard, our work is more similar to that of flight and driving simulators which model a user's experience within a (vehicle) avatar through simulation and a combination of haptic and visual feedback. Related research in the mixture of cues that lead to the perception of motion is informative. For example, [Riecke and Feuereissen 2012] suggest that minimal physical (haptic) cues can heighten the effect of vection, or the illusion of self motion primarily conveyed through the visual field. Further they find that haptic turning cues can also be used to create the visual illusion of wider turns. Overall, within VR research, our effort in building a general response system with a physical proxy and control system is unique to the best of our knowledge.

Largely, we draw inspiration for our physically based modeling approach from the related animation techniques. We do not employ full physical simulation with control because it is costly to compute and controllers are not yet robust enough. However, there has been a long history of using physical characteristics to drive animation synthesis [Arikan et al. 2005; Bruderlin and Calvert 1989; Girard and Maciejewski 1985; Popović and Witkin 1999; Sok et al. 2010; Tak and Ko 2005; Yin et al. 2005], among others. Adding forces to a physical proxy is akin to the use of mixed kinematic and dynamic systems, such as [Ishigaki et al. 2009; Majkowska and Faloutsos 2007; Nguyen et al. 2010; Shapiro et al. 2003; Yuting Ye 2008]. Notably, Ishigaki et al [2009] layer a kinematic model over a dynamic one. This work is similar to our own in that we propose a simple model and layer an articulated kinematic model on top for graphical representation. [Nguyen et al. 2012] describe an approach to create physically plausible reactions of a rigid body to external forces. The control approach of our framework has a similar whole-body system. Ours work stands out in part by its use in VR as a haptic tool, i.e. general physical interaction in a shared virtual environment. This also leads to our novel approach for stepping which creates the illusion of legged locomotion.

3 SYSTEM OVERVIEW

Our system layout is diagramed in Figure 2. Our physical rig is customized third-party hardware that we use as a 3-degree of freedom (roll, pitch, yaw) motion simulator. This device is controlled through stepper-motors that take their input from custom drivers. Specifically, we designed low-latency drivers using the .NET framework to directly transfer the proxy state to the mechanical rig. The game mechanics updates the player avatar which is directed by the players through joysticks mounted inside the rig. The user wears a stereo headset (Oculus) [Oculus 2017] that offers added immersion through look control and stereovision. With this custom hardware set-up, we design a user experience that transports the player to a virtual world, training and testing in a game experience. The game software is written in Unity3D [Unity3D 2017] and takes advantage of Unity's physics, network and game development tools.

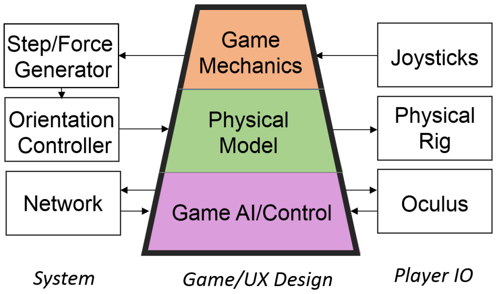
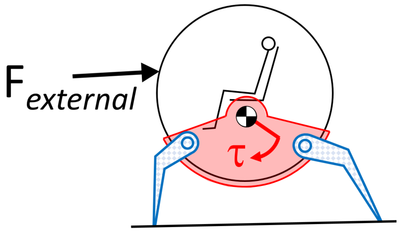


Figure 2: System layout

4 PHYSICAL PROXY

At the core of the proposed approach is a physical proxy that simulates the dynamics of the vehicle cockpit (or cabin) that is to contain the avatar pilot (or driver). We model the cockpit with a physical proxy assumed to be a rigid body that receives influences from the "external" (virtual) world and "internal" forces due to self-activation. Along with simple influences, such as impulses (e.g. derived from enemy fire) and forces due to gravity, we implement dedicated inputs to produce the desired effects of simulated behaviors. Namely, for walking robots, a stepping controller produces time-varying footstep forces for various gaits based on human and animal locomotion.



Torque applied to proxy due to external force

To respond to the disturbances due to stepping and external impacts, we introduce an orientation controller that rights the proxy following an impact from a footstep (or otherwise). The concept behind this choice is to allow the vehicle/robot to self-correct and this controller replaces a body-attitude controller that may be present in legged simulation control [Coros et al. 2011; Yin et al. 2007]. In the case of a wheeled vehicle, we also add a passive controller that acts as virtual shocks that comply in the presence of impacts, such as those derived from rough terrain.

4.1 Orientation Controller

The orientation controller corrects body attitude following disturbances. Specifically, the controller uses torque-based proportional derivative (PD) servos along three axes of rotation. The PD servo computes τ_{orient} based on a desired value, $\bar{\theta}$, set by the game AI/control. This is set to zero to represent an upright orientation or may be set otherwise to mimick a sloped ground plane. Specifically, the controller computes joint torques based on the tracking error $(\bar{\theta} - \theta)$ and joint velocities $(\dot{\theta})$ as

$$\tau_{\text{orient}} = k_{\theta}(\bar{\theta} - \theta) - d_{\theta}(\dot{\theta}) \tag{1}$$

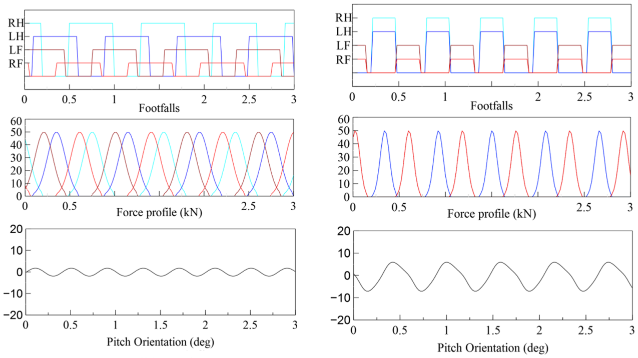
which corrects for disturbances based on the gain value k_{θ} and damping d_{θ}. Note, because we design our implementation to stay close to upright, due to limitations of the physical rig, a 3-axis hinge model is sufficient. If one were to design for more extreme rotations, a ball-socket would be more advisable.

In-game disturbances create accelerations in the physical proxy which lead to state changes for the proxy. This excites the orientation controller while the angles are piped into the rig hardware, resulting real-world/physical movement of the player. The control system can be tuned to give a range of experiences based on the size of the influences and gain values. We opted for a slightly under-damped system after empirical testing. The qualitative result is that as the orientation controller receives forces, it responds by yielding immediately in the direction of the force. It damps subsequent accelerations and, eventually, returns the proxy to the desired pose.

With the orientation controller in place, animation for locomotion is reduced to force production for footsteps. Likewise, design of other interactions can be built through the development of appropriate forces and their application to the physical proxy.

4.2 Stepping Controller

One goal we set out to achieve in this work was to build the experience of piloting a walking robot and to this end realistic stepping and legged locomotion are fundamental behaviors that immerse the player in the virtual world. With our physical proxy as an interface, locomotion production is built through the control of where and when footsteps appear, along with physically realistic forces applied with each step. Ground reaction forces (GRF) for a myriad of animal gaits [Griffin et al. 2004] provide a foundation for gaits for our imaginary biped and quadruped robots.

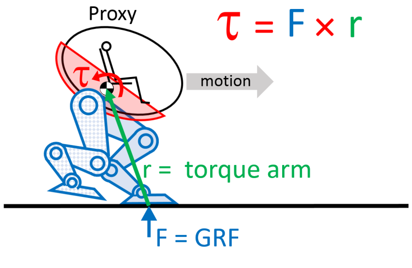


a) walking gait

b) bounding gait

Figure 3: Quadraped gait examples for walking (a) and bound (b) with diagrams for: timing of footfalls (top); force profiles (middle); and physical proxy's pitch axis response (bottom)

Footstep Force Design. We design gaits based on their basic characteristics, such as footfall patterns (Figure 3, top) and their relative timing. To match the profile of GRF of natural animals, we build force profiles using sets of Gaussian curves and add parameters to vary the mean and variance of the curve (Figure 3, middle). We set these values based on estimated mass and size of the imaginary robot and matching (scaled) natural biped and quadraped shape data. Likewise, we designed both the step placement and the forward motion of the robot in the game through step length and speed estimates of our avatar. While we attempted to build these strictly through real-world comparisons, we intentionally designed a stepping profile that felt "right" through iteration with user input.



Torques from step force

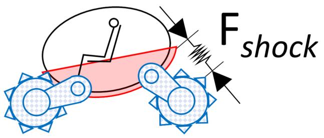
During gameplay, we compute a set of torques for each step by crossing the GRF with the the torque arm computed from the foot to the proxy center of mass. Applying the resulting torques to the proxy final leads to the orientation trajectories for locomotion (Figure 3, bottom). Note, the forward motion of the proxy through the environment leads to relative movement with respect to fixed step positions. Thus, as the step positions are approached and passed, torque (for pitch, in this case) changes appropriately. While further experimentation is needed, we found a convenient balance in the design with the game (and subsequently the player) controlling the speed of the character and the resulting translation leading to length variation in the torque arm.

As the stepping torques influence the orientation controller in the proxy, that control adds its own reaction torques via the PD servos. The aggregate is what is experienced by the player every time a stepping force is created in the virtual locomotion.

4.3 Wheeled Vehicle Suspension

For wheeled vehicles, we simulate suspension through additional forces for each tire that act as an independent shock. For each wheel assembly, we compute $\mathbf{F}_{\mathbf{shock}}$ using a Cartesian-based PD-servo as

$$\mathbf{F}_{\mathbf{shock}} = k_s(\bar{s} - s) - d_s(\dot{s}) \tag{2}$$



Shock suspension response force

where $\bar{s}$ is the rest length of the shock, s is the current length, and gains k_s and d_s are manually tuned constants. The effect of this controller is to act as a springlike actuator for the wheel with the spring pulling toward the rest length.

5 GAME DEMONSTRATIONS

To showcase the utility of our technique, we produced a number of indepth examples. Three characters are used in testing, a biped and quadruped robot and a wheeled rover. For each, we develop a rich game, loosely surrounding a location. The game experience was designed so that the player gradually eases into the world. We start with basic navigation, target practice, finally progressing through more complex challenges. The gradual increase allows the player to master nuanced control over the game itself. A sample of the worlds we created appears in Figure 4.

The design process is aligned with a standard development pipeline with the exception that the force-driven information has one additional layer. Namely, interaction force is communicated directly from its source – e.g. a force driven by enemy fire as the result of an opponent shot behavior – to a target, potentially the player proxy. The force then impacts the proxy to unpack the physics-driven response of the interaction. Locomotion acts similarly, repetitively disturbing the avatar's cockpit with each step. Resulting translation from the locomotion is conveyed through the visual field, along with crafted graphics (and sounds) for the game (Figure 5, top). Detail such as the external forces for enemy fire is conveyed through designed graphical representations to the player – for example, an explosion. In this manner, the effect of forces is seen from the player's point of view just as one may see the interaction in a generic VR setting or game. Note, for our game demonstrations, we focused on the haptic experience more so than the visual fidelity of the player avatar, i.e. its walking motion, since the full-body avatar is not readily visible from the player's first person camera. Thus the aim of the behavior synthesis is primarily haptic feedback.

We showcased the MechVR game at the SIGGRAPH VR Village in 2016 [Hindlekar et al. 2016] and the system was stress-tested by more than 400 participants in five days (Figure 5, bottom). Many players completed the demo level in the required time. We received a plethora of useful and critical feedback. While anecdotal, the concensus was that the experience was indeed extremely immersive and especially that our biped "really feels like walking" (a direct quote from one user).

6 CONCLUSION

The various aspects, from technology to design, culminates in an immersive VR experience that is unique and engaging. We anticipate that this line of research is uncovering new potential in motion simulation that has a strong footing in entertainment, but may also see benefits in serious testing and training.

Limitations to date include the limited degrees of freedom and ranges of the hardware rig. Because the rig only has orientation (and no translation), direct forces, e.g. hits aimed at the center of the proxy cannot be communicated effectively (through haptics alone) because their torque is negligible. Six degrees of freedom would be necessary to be completely generic, but such systems are cost prohibitive and raise (interesting) engineering challenges.

To extend this type of work to real world applications, beyond entertainment, more rigorous testing and verification is necessary. For future work, we are preparing a series of perceptual experiments that will expose the effect of the technology in more depth, and further, we hope to continue development and follow-on work in support of this burgeoning area of VR-based motion simulation.

ACKNOWLEDGMENTS

We would like to thank the Clemson School of Computing and Watt Innovation Center for their generous donations in support of this project, Simcraft for their help and support, as well as our anonymous reviewers and Siggraph VR Village participants for their constructive inout and feedback.

REFERENCES

Okan Arikan, David A. Forsyth, and James F. O'Brien. 2005. Pushing people around. In *SCA '05: Proceedings of the 2005 ACM SIGGRAPH/Eurographics symposium on Computer animation*. ACM Press, New York, NY, USA, 59–66.

Tom Banton, Jeanine Stefanucci, Frank Durgin, Adam Fass, and Dennis Proffitt. 2005. The perception of walking speed in a virtual environment. *Presence: Teleoperators and Virtual Environments* 14, 4 (2005), 394–406.

A. Bruderlin and T. W. Calvert. 1989. Goal-directed, Dynamic Animation of Human Walking. *SIGGRAPH Comput. Graph.* 23, 3 (July 1989), 233–242.

Stelian Coros, Andrej Karpathy, Ben Jones, Lionel Reveret, and Michiel Van De Panne. 2011. Locomotion skills for simulated quadrupeds. In *ACM Transactions on Graphics (TOG)*, Vol. 30. ACM, 59.

Harald Frenz, Markus Lappe, Marina Kolesnik, and Thomas Bührmann. 2007. Estimation of travel distance from visual motion in virtual environments. *ACM Transactions on Applied Perception (TAP)* 4, 1 (2007), 3.

Figure 4: Top: Land rover driving over rough terrain. Below: Quadbot enters enemy territory.

Figure 5: Top: First-person footage with visual and game elements added. Bottom: MechVR Siggraph VR Village 2016 demonstration

Michael Girard and A. A. Maciejewski. 1985. Computational Modeling for the Computer Animation of Legged Figures. In *Proceedings of the 12th Annual Conference on Computer Graphics and Interactive Techniques (SIGGRAPH ’85)*. ACM, New York, NY, USA, 263–270.

Timothy M. Griffin, Russell P. Main, and Claire T. Farley. 2004. Biomechanics of quadrupedal walking: how do four-legged animals achieve inverted pendulum-like movements? *Journal of Experimental Biology* 207, 20 (2004), 3545–3558. `DOI`: http://dx.doi.org/10.1242/jeb.01177

Saurabh Hindlekar, Victor B. Zordan, Emerson E. Smith, John C. Welter, and William Garrett Mckay. 2016. MechVR: Interactive VR Motion Simulation of "Mech" Biped Robot. In *ACM SIGGRAPH 2016 VR Village (SIGGRAPH ’16)*. ACM, Article 14, 2 pages.

Satoru Ishigaki, Timothy White, Victor B Zordan, and C Karen Liu. 2009. Performance-based control interface for character animation. In *ACM Transactions on Graphics (TOG)*, Vol. 28. ACM, 61.

Omar Janeh, Eike Langbehn, Frank Steinicke, Gerd Bruder, Alessandro Gulberti, and Monika Poetter-Nerger. 2017. Walking in Virtual Reality: Effects of Manipulated Visual Self-Motion on Walking Biomechanics. *ACM Trans. Appl. Percept.* 14, 2, Article 12 (Jan. 2017), 15 pages.

Anna Majkowska and Petros Faloutsos. 2007. Flipping with physics: motion editing for acrobatics. In *Proceedings of the 2007 ACM SIGGRAPH/Eurographics symposium on Computer animation*. Eurographics Association, 35–44.

Betty J Mohler, Jennifer L Campos, M Weyel, and Heinrich H Bülthoff. 2007a. Gait parameters while walking in a head-mounted display virtual environment and the real world. In *Proceedings of Eurographics*. 85–88.

Betty J Mohler, William B Thompson, Sarah H Creem-Regehr, Herbert L Pick, and William H Warren. 2007b. Visual flow influences gait transition speed and preferred walking speed. *Experimental brain research* 181, 2 (2007), 221–228.

Franck Multon and Anne-Hélène Olivier. 2013. Biomechanics of walking in real world: naturalness we wish to reach in virtual reality. In *Human Walking in Virtual Environments*. Springer New York, 55–77.

N Nguyen, R Arista, K C Liu, and V Zordan. 2012. Adaptive Dynamics with Hybrid Response. *Siggraph Asia (Technical Brief)* (2012).

Nam Nguyen, Nkenge Wheatland, David Brown, Brian Parise, C Karen Liu, and Victor Zordan. 2010. Performance capture with physical interaction. In *Proceedings of the 2010 ACM SIGGRAPH/Eurographics Symposium on Computer Animation*. Eurographics Association, 189–195.

Oculus. 2017. https://www.oculus.com/en-us/. (2017).

Zoran Popović and Andrew Witkin. 1999. Physically Based Motion Transformation. In *Proceedings of the 26th Annual Conference on Computer Graphics and Interactive Techniques (SIGGRAPH ’99)*. ACM Press/Addison-Wesley Publishing Co., 11–20.

Bernhard E Riecke and Daniel Feuereissen. 2012. To move or not to move: can active control and user-driven motion cueing enhance self-motion perception (vection) in virtual reality?. In *Proceedings of the ACM Symposium on Applied Perception*. ACM, 17–24.

Bernhard E Riecke and Jan M Wiener. 2007. Can people not tell left from right in VR? Point-to-origin studies revealed qualitative errors in visual path integration. In *Virtual Reality Conference, 2007. VR’07. IEEE*. IEEE, 3–10.

Ari Shapiro, Frederic Pighin, and Petros Faloutsos. 2003. Hybrid Control For Interactive Character Animation. In *Pacific Graphics 2003*. 455–461.

SimCraft. 2017. http://www.simcraft.com/. (2017).

Kwang Won Sok, Katsu Yamane, Jehee Lee, and Jessica Hodgins. 2010. Editing Dynamic Human Motions via Momentum and Force. In *Proceedings of the 2010 ACM SIGGRAPH/Eurographics Symposium on Computer Animation (SCA ’10)*. Eurographics Association, Aire-la-Ville, Switzerland, Switzerland, 11–20. http://dl.acm.org/citation.cfm?id=1921427.1921430

Frank Steinicke, Gerd Bruder, Jason Jerald, Harald Frenz, and Markus Lappe. 2010. Estimation of detection thresholds for redirected walking techniques. *IEEE transactions on visualization and computer graphics* 16, 1 (2010), 17–27.

Frank Steinicke, Yon Visell, Jennifer Campos, and Anatole Lécuyer. 2013. Human walking in virtual environments. (2013).

Seyoon Tak and Hyeong-Seok Ko. 2005. A Physically-based Motion Retargeting Filter. *ACM Trans. Graph.* 24, 1 (Jan. 2005), 98–117.

Unity3D. 2017. http://www.unity3d.com/. (2017).

KangKang Yin, Kevin Loken, and Michiel van de Panne. 2007. SIMBICON: Simple Biped Locomotion Control. *ACM Transactions on Graphics* 26, 3 (2007), 105:1–105:10.

K. Yin, D.K. Pai, and M. van de Panne. 2005. Data-driven interactive balancing behaviors. *Pacific Graphics* (2005).

C. Karen Liu Yuting Ye. 2008. Responsive Characters with Dynamic Constraints in Near-Unactuated Coordinates. *ACM Transactions on Graphics* 27, 5 (Dec. 2008).