14 Embodied Autonomous Agents

Andrew Feng, Ari Shapiro, Margaux Lhommet,
and Stacy Marsella

CONTENTS

14.1 INTRODUCTION

Since the last decade, virtual environments have been extensively used for a wide range of applications, from training systems to video games. Virtual humans are animated characters that are designed to populate these environments and to interact with the objects of the world as well as with the user. A virtual agent must perceive the world in which it exists, reason about those perceptions, and decide on how to act on them in pursuit of its own agenda.

The work on virtual humans has become especially concerned with the bold challenge of realizing naturalistic face-to-face interactions between virtual (and often humanlike) agents and human participants, typically seeking to create virtual agents that interact with people using the same verbal and nonverbal behavior that people use to interact with each other. These virtual agents have gone by a variety of names, most notably embodied conversational agents (Cassell, 2000) and virtual humans (Gratch et al., 2002; Rickel et al., 2002).

Virtual humans have been proposed for a wide range of educational, social, medical, and training applications, where the virtual humans operate as peers, mentors, subordinates, patients, adversaries, or background characters.

The design and evaluation of such sophisticated artifacts is a multidisciplinary effort, requiring the integration of research spanning artificial intelligence, social psychology, linguistics, and computer graphics and animation.

Designers of virtual humans aspire to satisfy multiple requirements. Foremost, virtual humans must be *responsive*; that is, they must respond flexibly to the human user and to the events in the virtual environment. Second, they must be *believable*; that is, they must provide a sufficient illusion of humanlike behavior so that the human user will be drawn into the interaction. Finally, they must be *interpretable*; the user must be able to interpret their response to situations, including their dynamic cognitive and emotional state, using the same verbal and nonverbal behaviors that people use to understand one another.

Whereas virtual human research draws on work in artificial intelligence, graphics, and dialogue systems, it also faces unique challenges that stem from virtual humans being autonomous, embodied facsimiles of people that must be responsive, believable, and interpretable. In this chapter, we therefore touch on those challenges that are more unique to the realization of virtual humans capable of interacting with human users.

In particular, the burden of realizing believable, responsive, and interpretable behavior falls, in part, on the virtual humans' outward behavior in the virtual world, for it is that behavior that a user perceives. The virtual humans must be responsive to events in the virtual world, acting and reacting. They must express realistic emotions in their facial expressions, their gestures, their postures, etc. They must be able to carry on spoken dialogues with humans and other agents, including all the nonverbal communication that accompanies human speech (e.g., eye contact and gaze aversion, postural shifts, facial displays, and gestures). Finally, they must be able to take physical actions in the world such as walking or grasping an object.

For that reason, we begin our discussion of virtual humans with a discussion of realizing its outward appearance and behavior. We then proceed to discuss the mental capabilities required and conclude with a discussion of how the daunting challenges of realizing a virtual human are often addressed.

14.2 APPEARANCE OF A VIRTUAL CHARACTER

It is important to decide which aspects of an autonomous character will be modeled, which will depend on the circumstances in which it is used. Applications that require physically close, verbally rich interactions between a single agent and a user differ, such as those used for conversational agents (Heloir & Kipp, 2009; Poggi, Pelachaud, de Rosis, Carofiglio, & De Carolis, 2005; Thiebaux, Marsella, Marshall, & Kallmann, 2008; van Welbergen, Reidsma, Ruttkay, & Zwiers, 2009), from those used for personal or small group interaction (Shapiro, 2011) and from those that require interactions at a distance with crowds (Shoulson, Marshak, Kapadia, & Badler, 2013) or groups, such as a city-scale event simulations (Yersin, Maïm, Pettré, & Thalmann, 2009).

14.2.1 Traditional 3D Character Representation

A common method to model virtual characters is to use geometry-based 3D meshes that are controlled through a hierarchical set of nodes representing the character's joints and other moving parts. A 3D mesh is constructed either manually by digital artists using 3D modeling software or automatically via a 3D scanning process and later adjusted manually. Such models are effective at representing surface features, such as the outward appearance of a character's face, skin, or clothing. In addition to the 3D mesh, surface features such as colors, textures, light reflectivity properties, and some subsurface structures can be visually modeled by adding different kinds of maps, including texture maps (2D images transferred onto the 3D geometry), and other 2D maps that create complex interactions with lighting and other external objects, such as bump, normal, specular, diffuse, and related maps. Specialized surface features can also be mimicked by creating special textures for features such as wrinkles and blushing. Such methods rely on the illusion of an outward appearance as a representation of the internal structures of a virtual character. For example, subcutaneous structures such as muscles and bones are indirectly shown through their effects on the surface features rather than explicitly modeled (Ng-Thow-Hing & Fiume, 1997). Modeling such internal structures explicitly can yield greater realism at the expense of complexity and computation time. Internal structures are often modeled for effect rather than for realism, for example, a muscle layer that does not contain the anatomically correct number of the type of muscles, but rather a representative muscular layer that contributes bulging and stretching to the skin layer above (Lee, Sifakis, & Terzopoulos, 2009). See Section 14.2.5, *Skinning,* for details of the skeleton to mesh binding algorithms.

14.2.2 GPU-BASED SHADING

In recent years, the advent of GPU-based shaders has allowed the incorporation of many high-quality techniques suitable for offline rendering available for interactive simulations. By running the algorithms on the hardware of a video card, as opposed to on software in the computer's CPU, techniques for shadowing, deformation, simulation, and display can be run in real time on commodity hardware. For example, GPU-based physics simulation enables the rapid physical simulation of rigid or soft bodies (Yeh, Faloutsos, & Reinman, 2006), and GPU-based shadowing algorithms (Kim & Neumann, 2001) allow for detailed lighting effects on volumetric surfaces. Shaders can also be designed to model complex anatomical structures such as the reflectivity and behavior of the human eye (Pamplona, Oliveira, & Baranoski, 2009) or skin (Jimenez, Sundstedt, & Gutierrez, 2009).

14.2.3 IMAGE-BASED CHARACTER REPRESENTATION

Image-based techniques for character representation show promise for character modeling. Rather than modeling a character using 3D geometry explicitly, numerous images are taken from video and then reconstructed in 3D to form a character (Casas, Tejera, Guillemaut, & Hilton, 2011; Starck, Miller, & Hilton, 2005). The advantage of such techniques is the ability to capture the dynamic aspects of character without explicit modeling. For example, the movement of a clothing as it deforms in response to a person's bending and moving can be captured automatically by an image-based technique without the need to understand the physical characteristics of the clothing material and its interactions with the person wearing it (Starck & Hilton, 2007). The advantages of image-based techniques also lend themselves to a similar disadvantage; without an understanding of the underlying system, it is easy and common to make errors during the reconstruction of a character. For example, during a particular movement, parts of the body may momentarily appear as if they are connected to each other, when they are not. These techniques represent interesting approaches that are useful in situations where characters cannot be explicitly modeled but are relatively new and are not widespread in their use.

Image-based character representation may include video-based representations. By recording and playing back a video of a live actor, a virtual character can be synthesized using a set of carefully controlled video clips (Bregler, Covell, & Slaney, 1997). This, of course, affords limited interaction and viewing perspectives but affords the possibility of photorealistic characters, which is difficult to do using traditional 3D techniques due to the resolution and fidelity requirements of photorealism.

14.2.4 FORWARD KINEMATICS/INVERSE KINEMATICS

With a virtual character's skeleton hierarchy, we are able to define new postures for the character by changing the skeletal configuration. There are mainly two different ways to adjust the skeletal configuration—forward kinematics (FK) and inverse kinematics (IK).

14.2.4.1 Forward Kinematics

The goal of FK is to obtain the position of a joint based on input joint parameters such as joint angles. It works by using both the kinematic equation and the skeleton hierarchy to recursively update the joint positions given a new set of joint parameters. This computation is simple to implement and is the most straightforward way to animate a character. However, when the hierarchy is very complicated and contains a large number of joints, FK update may become too slow to be computed in real time. This is one of the reasons that the skeleton hierarchy for interactive virtual character is usually much simpler than the one used in feature animation films; real-time performance is crucial for interactive application while animation quality is the key competence for feature animation production. To allow interactive editing and update of highly complex skeleton hierarchy, DreamWorks Animation utilized the multicore CPU by segmenting the hierarchy into multiple dependency graphs (Watt et al., 2012). Each graph can be updated independently in parallel. This helps accelerate the FK process to allow interactive update in their production pipeline. Similar technique could be applied for an interactive virtual character should the complexity of the character increase to the level for animation production.

14.2.4.2 Inverse Kinematics

IK, as its name suggests, is an inverse problem to FK. It takes the desired position(s) for some target joint(s) as constraints and computes joint parameters that will move the target joints to those positions. This can be useful when we want the virtual character to accurately execute some actions. For example, when a character is reaching for an object, it is desirable for his hand to touch the object. However, the provided joint angles from motion capture may not accurately put his hand on the right place due to various capturing errors. Thus, IK can be applied to adjust the joint angles to ensure the hand constraint is satisfied.

In general, this is a more difficult problem than FK since there are multiple sets of the feasible joint parameters that could satisfy an input constraint. This problem can be regarded as the nonlinear optimization problem that minimizes the error between the desired joint constraint and actual joint positions by adjusting joint parameters over the IK chain. Here, the IK chain is defined as the bone segments from target joint to root joint. The problem is not trivial since the mapping from joint angles to joint positions is nonlinear. There are several different methods to compute IK for a given constraint. They differ by their simplicity, performance, and robustness.

Cyclic coordinate descent (CCD) is a popular IK method widely adapted in the video game industry (Canutescu & Dunbrack, 2003; Lander & Content, 1998). It is based on a heuristic by iterating through each joint in the IK chain and rotating the bone segment to move target joints toward the constraint positions. It is simple to implement and is very efficient to compute. However, since its computation is based on a simple heuristic, it is difficult to guarantee the consistency across different animation frames. Depending on the traverse order, it may be possible that two similar constraint positions give rise to vastly different joint parameters and thus causes discontinuity in the resulting animations. This problem is even more obvious when the skeleton hierarchy is deep. In practice, this method is more suitable for real-time animation with relatively few joints involved.

Analytical method derives a direct solution for the IK problem (Badler & Tolani, 1996; Kallmann, 2008; Tolani, Goswami, & Badler, 2000). It requires the whole IK chain to consist of no more degrees of freedom (DOF) than the target joint constraints to uniquely determine the solution. Although this restriction reduces the applicability for this method, it is usually enough to use only 7-DOF to model an arm or a leg for a virtual character. For example, a typical human arm consists of 3-DOF for shoulder, 2-DOF for elbow, and 2-DOF for the wrist. A similar analogy can be made for a leg with hip, knee, and ankle joints. Therefore, a typical virtual character can be modeled by four separate IK chains to model both arms and legs. Notice that the joint parameters are not unique given the target joint position since target constraints only have 6-DOF (3 for position and 3 for rotation). Therefore, the user needs to provide a swivel angle for elbows or knees to obtain a unique solution. The analytical method is very efficient to compute and could be suitable for virtual characters with simple skeleton hierarchy.

Jacobian method approximates the solution by solving a numerical optimization problem directly through linearization (Buss & Kim, 2005; Yamane & Nakamura, 2003). It works by computing a Jacobian matrix, which encodes the partial derivative from target joint position to joint parameters. This reduces the nonlinear optimization problem into a series of linear least square problem and can be solved via a numerical linear solver. Since the problem is linearized, the solution is varying smoothly based on initial skeletal configurations and target joint constraints. Moreover, it can integrate the desired joint angles as the secondary target. Thus, the solution will match the target joint constraints while satisfying the desired posture as much as possible. This provides a nice IK solution to adjust a character's final posture while staying faithful to the original posture. However, the method has two drawbacks. The first is that it suffers from the singularity when the target position is out of reach from the designated IK chain. In this situation, the Jacobian matrix becomes singular since there are no angle adjustments that can move the target joint closer to the constraint. Thus, the solution becomes unstable and will introduce jerky motions as a result. The second issue is that the solution requires solving a dense linear system during each iteration. For complex skeleton hierarchy, this computation becomes very expensive for real-time application since the computational complexity is cubic to the number of joints in the system. Therefore, the method is more applicable when the IK chain is relatively simple.

Particle-based method is a variation of CCD based on similar heuristics (Hecker et al., 2008). It also recursively updates the bone segments to orient the target joint toward the constraint position. The difference is that it operates directly on the joint positions instead of rotations. The method adjusts the bone segments in a two-pass manner. During the first pass, the end joint of a bone segment is simply moved toward the desired position without considering the length constraint of its bone segment. The length constraint is then enforced during the second pass by moving both end joints of that bone segment. The method then iterates this two-pass process over each bone segment until converges. The analogy of the process is like attaching a spring to each bone segment. Thus, the end joint is free to move by extending its bone segment, but the bone segment must eventually return to rest length for equilibrium. Overall, this method has similar property as CCD but provides smoother solutions due to the fact that it only operates on the position domain instead of joint angles.

14.2.5 SKINNING

Skinning is the process that transforms the skeletal animation to character mesh. In general, it animates a mesh by blending joint transformations and applying them on each mesh vertex. Here, the blending weights for transformations need to be defined on each vertex and are usually manually created by the animator. With skinning, the motions only need to be defined at the skeleton level to animate the whole mesh. Thus, the effort for animating a mesh is greatly simplified. There are several different techniques involved in mesh skinning that address issues that arise during the skinning process.

Linear blend skinning, sometimes also called skeletal subspace deformation (SSD), is the simplest method for skinning. It represents the joint transformations, which are usually rigid transformations, as 4×4 matrices and computes the weighted sum of the matrices to transform each vertex (Lander, 1998; Thalmann, 1989). This is a straightforward process that has been widely used in many interactive applications such as video games due to its simplicity (Lander). However, this simple method comes with a drawback since combining matrices by weighted average will not necessarily preserve the desired quality. For example, the average of rotation matrices will not result in a rotation matrix. Therefore, the resulting mesh deformation may have a collapsed shape and lose volume. This problem is especially obvious when the joint is bent more than 90° and twisted, which results in the notorious *candy wrap* artifacts (Mohr & Gleicher, 2003).

Pose space deformation (PSD) was introduced to alleviate the artifacts from SSD (Lewis, Cordner, & Fong, 2000). The main idea is to create a set of example shapes for some important poses and compute the residue differences between these shapes and the shapes produced from SSD. Vertex offset is used to represent these residues and it encodes the errors caused by the SSD method. To compensate these errors, the residues are interpolated for a new pose and added to the new mesh shape generated by SSD. The technique can effectively improve the artifacts common in SSD skinning. However, it requires additional efforts from artists to sculpt the example mesh shapes for various poses and adds more computation during run-time animation.

Dual quaternion skinning (DQS) is an alternative to SSD (Kavan, Collins, Žára, & O'Sullivan, 2007; Kavan, McDonnell, Dobbyn, Žára, & O'Sullivan, 2007). Similar to SSD, it blends joint transformations together to produce animated mesh. Instead of representing rigid transformations as matrices, it uses dual quaternion (DQ) to encode the transformation. DQ is a compact representation for both rotation and translation. Its advantage is that one can blend different DQs together to still produce a DQ, which is always a valid rigid transformation. Therefore, it does not suffer from the collapsing and candy wrap artifacts that plaque the SSD technique. Overall, it is a better skinning method than traditional SSD in terms of quality, though it requires more efforts to correctly implement dual DQ algebra so the conversions between DQ and rigid transformation are handled correctly.

Although skinning techniques can efficiently represent the mesh animations of a virtual character, they also have some limitations. One limitation is that it is difficult to model some deformation effects such as muscle bulges with rigid transformations from skeletal joints. PSD can partially alleviate this problem with example mesh shapes. However, for deformations with large rotations such as 180° bending or twisting, PSD may not produce high-quality results since its residues are represented as vertex offsets. Recent research works addressed this issue by adding the shear and scale components to model the muscle bulge effects in the joint transformation in addition to rigid transformation. The correct shear and scale parameters can be predicted based on skeletal pose by precomputing a regression model from example mesh shapes (Feng, Kim, & Yu, 2008; Kim, Feng, & Yu, 2010; Wang, Pulli, & Popović, 2007). The quality of results would depend on how well the learned regression model captures the muscle deformations from the example shapes and whether it can generalize to new input skeletal poses.

Another limitation is that self-collisions could happen in the resulting mesh deformations. Since skeleton is a simplified representation for the virtual character, it does not have the underlying knowledge about its corresponding character mesh. Therefore, it would be difficult to resolve potential self-collisions in the animated mesh from simply checking the skeleton configurations. One possible solution for this problem is to attach a bounding volume such as box or cylinder to each bone segment as an approximation to the corresponding mesh. These bounding volumes can then be used to detect self-collisions given a new skeleton configuration. A new set of joint parameters can then be inferred to resolve these collisions between bounding volumes. This method can work as an approximate solution to collision problems, though it cannot model more subtle deformations such as compressed skins due to collisions. This issue is still an open problem in the research. More accurate and efficient methods are desired to resolve the collisions while maintaining these subtle skin deformations.

14.2.6 CLOTH

Cloth is an important part of the appearance for a virtual character. Appropriate clothing can improve the realism and produce distinct visual styles for each character. Based on the type of clothes and the available computation resources, a cloth can be either animated from kinematic body movements or simulated physically.

Since most types of clothes we wear such as shirts or jeans fit tightly on a character, their deformations will roughly follow the deformation of body segments. Therefore, one can approximate the cloth shapes by computing its deformations in a similar manner like one computes them for the character. This is done by assigning a suitable skin blend weights for each vertex on the cloth and deforming them based on joint transformations of the character. This provides a simple method to quickly produce cloth animations with lower quality. The problem with this method is that the secondary motions due to dynamics and other fine wrinkle details are totally ignored. Therefore, it can only be used to approximate tightly fit cloth and is not suitable to model highly dynamic cloth such as skirts.

Physical simulation, on the other hand, can produce highly detailed cloth deformations with interesting dynamic effects according to character movements (Baraff & Witkin,1998; Choi & Ko, 2005). For example, a dancer with a long skirt can demonstrate highly dynamic body movements. These body movements can be visually enhanced by simulating the skirt to highlight some interesting actions such as spinning or speed changes.

The most common way of cloth simulation is to model the cloth as a set of mass particles connected by springs (Baraff & Witkin,1998; Bridson, Marino, & Fedkiw, 2003; Choi & Ko, 2005). During each simulation step, the particles are affected by both external force such as gravity and internal spring forces. The spring forces keep the particles within a reasonable distance to each other and thus model the inextensible property of a cloth sheet. In addition to mass-spring model, the properties of different textiles can also be measured from the real-world material and simulated using a finite element method (Etzmuss, Keckeisen, & Strasser, 2003). This requires significant setup to correctly measure the desired material. The run-time simulation using finite element method is also more expensive than a mass-spring model. However, this method produces more realistic and accurate simulation results that are closer to real textile material. This makes it is suitable for applications that require high-quality simulations such as virtual fashion design.

With recent advances in graphics hardware, now the cloth simulation with moderate complexity can be computed in real time (Cordier & Magnenat-Thalmann, 2002). It is done by reformulating the cloth simulation steps so they can be computed in parallel on modern graphics hardware. Specifically, it divides a simulation step into two stages, the particle simulation and the constraint limiting (Müller, Heidelberger, Hennix, & Ratcliff, 2007; Zeller 2005). The particle simulation stage treats each cloth particle as unconstrained. Thus, each particle can be simulated independently. The constraint limiting stage checks each pair of particles with a spring connection and adjusts the particle positions if their distance is too far from or too close to each other. Both stages can be run in parallel, and thus, the whole simulation can be executed efficiently on hardware. This makes it possible for an interactive virtual character to be dressed with dynamic cloth to produce interesting effects.

14.3 MENTAL PROCESSES

To create a socially responsive embodied facsimile of a human capable of interacting with a person face to face requires the modeling of an array of capabilities in a virtual human.

First, the virtual human must *perceive* the environment in which it is embodied, including any humans it interacts with. It must *reason* about how events in that environment impact its goals and react to those events appropriately. More specifically, for the virtual human to seem humanlike, it should *react emotionally* appropriately.

We may additionally want it to support spoken language interaction with humans that cohabit in the virtual environment, including the use of nonverbal behaviors such as facial expressions, head movements, gaze, and gestures that play such a central role in human face-to-face interaction. In this section, we discuss the challenges that underlie such capabilities and requirements.

14.3.1 Perception, Attention, and Perceptual Understanding

For a virtual human to be responsive to what happens in the environment, it must perceive and understand the situation, that is, to what extent it impacts its goals and what action can be made to change the situation.

14.3.1.1 Environment

One effort here is to ensure that the virtual human's perceptual and attentional capabilities are consistent with human limitations. Thus, the virtual human should not be omniscient, not hear or look through walls, and show semblance of a focus of attention. Computational models that drive the virtual human perceptual attention and perception ensure behavior consistent with human capabilities. For example, the level of detail at which a virtual human will perceive objects and their properties in the virtual world can be predicted (Hill, 1999; Kim, Hill, & Traum, 2006). The types of visual attention required for several basic tasks (such as locomotion, object manipulation, or visual search), as well as the mechanisms for dividing attention among multiple tasks can be determined (Khullar & Badler, 2001).

Two methods can be used for the agent to understand the perceived situation.

In informed environments, the virtual human can directly read inside the objects what actions can be realized with or upon it. Such objects are called *smart objects* (Kallmann & Thalmann, 1999) and they have been extensively used in virtual environments.

More compatible with the reality, some virtual humans have their own model of the world that they maintain by using their perceptions. For example, STEVE has a symbolic model of the world that he updates whenever he perceives changes of objects and attributes (Rickel & Johnson, 1997, 2000).

14.3.1.2 Human Interactant

Another key effort here is to ensure the virtual human can perceive the behavior of humans immersed in the virtual environment. Most notably, there is considerable effort being undertaken to give virtual humans the ability to perceive and recognize the nonverbal behavior of human interactant including facial expressions, postural shifts, gestures, head movements, and gaze. As we noted earlier, such behavior plays a critical role in human face-to-face interaction. A complete review of the recent work in this area can be found in Scherer et al. (2012).

14.3.2 REASONING AND REPRESENTATION

The basis that allows virtual humans to act as autonomous agents and to be perceived by human users as possessing agency is the goal of a virtual human. Virtual humans do not usually react only to the environment, but they also are proactive. To achieve these goals, virtual humans must be able to generate plans or intentions, decide on appropriate actions, and react to unexpected events. They need to represent their beliefs about past events, present circumstances, and future expectations, especially in terms of how those impact their goals. Additionally, to interact effectively with other agents and humans, a virtual human may maintain beliefs about them as well, including what others believe and what their goals are.

To achieve this functionality, work in virtual human research has explored a variety of techniques drawn from research in artificial intelligence, cognitive science, and robotics. This includes planning and decision-making, cognitive architectures, work on belief, desire and intention frameworks for agent design, and robotics work on path planning.

14.3.3 EMOTION

Emotion has a central, powerful effect on human behavior. It affects how people perceive the world, think, act, and speak. Studies have identified its critical role in human decision-making, influencing the subjective value of alternative choices (Busemeyer, Dimperio, & Jessup, 2007) and guiding decision-making (Bechara, Damasio, Damasio, & Lee, 1999). Work by Simon (1967) argued that emotions serve a critical function in human behavior, interrupting cognition when unattended goals need to be addressed and therefore providing a means for a person to balance competing goals as well as supporting reactions to unexpected events. Research has also argued how social emotions such as anger and guilt may reflect a mechanism that improves group utility (Frank, 1988). Collectively, these findings underline that in many respects, it is obvious to us all that in everyday life, emotions have important influences in human behavior.

Emotion expression plays a powerful role in shaping human behavior and social interaction. From emotional displays, observers can form interpretations of a person's beliefs (e.g., frowning at an assertion may indicate disagreement), desires (e.g., joy gives information that a person values an outcome), and intentions/action tendencies (e.g., fear suggests flight). With such a powerful signal, it is not surprising that emotions can be a means of social influence and control (Campos, Thein, & Owen, 2003; Fridlund, 1997; Waal, 2003). For example, anger can coerce reactions in others and enforce social norms; displays of guilt can elicit reconciliation after some transgression, distress can be seen as a way of recruiting social support, and displays of joy or pity are a way of signaling such support to others. Other emotion displays seem to exert control indirectly by inducing emotional states in others and thereby influencing an observer's behavior. Specific examples of this are empathy and emotional contagion that can lead individuals to *catch* the emotions of those around them (Hatfield, Cacioppo, & Rapson, 1994).

These findings on the functional, often adaptive, role that emotions play in human behavior have led researchers to incorporate models of human emotion and emotional expression as core capabilities in virtual human systems in order to realize more humanlike behavior (Dias & Paiva, 2005; Rickel & Johnson, 1997). Emotional displays can make the virtual human seem human or lifelike and thereby influence the user to respond to, and interact with, it as if it were a person. In that people utilize these behaviors in their everyday interpersonal interactions, modeling the function of these behaviors is essential for any application that hopes to faithfully mimic face-to-face human interaction. More importantly, however, the ability of emotional behaviors to influence a person's emotional and motivational states could potentially, if exploited effectively, guide a user toward more effective interactions.

Since emotions are so pervasive and influence all cognitive processes, incorporating them into virtual human systems poses certain constraints. Marsella, Gratch, and Petta (2010) present an overview of the existing computational models of emotions and their applications. For example, researchers have looked at emotion and emotional expression in characters as a means to engender empathy and bonding between users and virtual characters (Marsella & Gratch, 2003; Paiva et al., 2005).

14.3.4 NATURAL LANGUAGE DIALOGUE

The ability of a virtual human to engage in natural language interactions with a human user must address a range of challenges common to work in dialogue systems generally. This includes speech recognition to determine what words are being uttered by a human user, natural language understanding to comprehend the meaning of the utterance, dialogue management to determine the role in the ongoing conversation and how to respond, natural language generation to transform the virtual human's response into utterance text, and speech synthesis to transform that text into spoken language. In practice, virtual humans differ dramatically in how these functions are realized. For example, at one simpler extreme, the natural language understanding and dialogue management could be realized by a system that simply maps words recognized by a speech recognizer to an appropriate response. A more sophisticated approach may employ understanding and dialogue management techniques that attempt to match the interpretation of the utterance against the virtual human's representation of the context, including their beliefs about the past, present, and future as well as the state of the conversation and their goals (see Traum, Swartout, Gratch, and Marsella, 2008; Traum & Larsson, 2003; Larsson, Staffan, & Traum, 2000). Such sophistication allows the agent to make determinations, for example, about whether to listen, take the dialogue turn, seek clarification on an issue, change topics, or take nondialogue actions instead, overall providing for more flexible, natural interactions. Similarly, speech synthesis may employ more flexible generative text-to-speech synthesis techniques for generating the spoken language or simpler techniques of having a fixed prespecified set of utterances recorded by a voice actor and simply playing back that recorded audio to generate the virtual human's spoken language.

14.3.5 Nonverbal Behavior

The flip of a hand, a raising of an eyebrow, a gaze shift: the physical, nonverbal behaviors convey a wide variety of information that powerfully influences face-to-face interactions. The relation between nonverbal behavior and speech is complex. Nonverbals can stand in different, critical relations to the verbal content, providing information that embellishes, substitutes for, and even contradicts the information provided verbally (Ekman & Friesen, 1969; Kendon, 2000). For example, a nod can convey agreement, and a beat gesture emphasizes a point. Nonverbal behaviors also serve a variety of rhetorical functions. Shifts in topic, for example, can be cued by shifts in posture or shifts in head pose. In addition, a wide range of mental states and character traits can be conveyed: gaze reveals thought processes, blushing suggests shyness, and facial expressions intentionally or unintentionally convey emotions and attitudes. Finally, nonverbal behavior helps manage conversation, for example, by signaling the desire to hold onto, get, or hand over the dialogue turn (Argyle & Cook, 1976; Bavelas, 1994). A speaker's aversion of gaze reflects they are thinking, in essence, regulating cognitive load as they consider what to say next while also signaling they want to hold onto the dialogue turn.

Nonverbal behaviors are so pervasive in every moment of face-to-face interaction that their absence also signals information—that something is wrong, for example, about the physical health or mental state of the person. Integrating nonverbal behaviors is therefore important to improve the quality of interaction. However, the issue encountered is the absence of a computational model of nonverbal behaviors that would answer the question of what behaviors to exhibit and when to exhibit them. Creating such a model faces several challenges.

First, the context in which the behavior occurs can transform the interpretation, as can even subtle changes in the dynamics of the behavior: head nods signaling affirmation versus emphasis typically have different dynamics. Behaviors can also be composed with each other, further transforming their interpretation. The generation of the behaviors must additionally take into account that the behaviors are synchronized, often tightly, with the dialogue, and changes in this synchronization can lead to significant changes in what is conveyed to a listener. For instance, the stroke of a hand gesture, a nod, and eyebrow raise performed individually or together are often used to emphasize the significance of a word or phrase in the speech. To achieve that emphasis, the behavior must be closely synchronized with the utterance of the associated words or phrases being emphasized. Alteration of the timing will change what words are being emphasized and consequently change what is conveyed to a listener. Achieving such synchronization in a virtual character can be difficult, especially in the case of behaviors such as hand gestures that involve relatively large-scale motion and preparatory phases to bring the hand into position to perform the gesture.

Such challenges make the pattern and timing of the behavior animations that accompany utterances unique to the utterance and the state of the character.

14.4 GROUP AND CROWDS

Some virtual environments integrate crowds or large groups of characters. Although aforementioned techniques can indeed be applied to multiple characters, several aspects need to be considered. In particular, animating a large number of high-quality intelligent virtual humans requires significant computation resources.

14.4.1 Level of Detail

Animating a large number of high-quality virtual characters requires significant computation resources, so it may not be feasible to animate and render all characters with full resolution.

The common technique is to apply level-of-detail control to dedicate more computation resources to characters that are closer to the user (Di Giacomo, Moccozet, Kim, & Magnenat-Thalmann, 2007; Luebke, Watson, Cohen, Reddy, & Varshney, 2002). This would require a multiresolution mesh for the virtual character; the mesh will be simplified or refined based on the current animation and camera view (Feng, Kim, Yu, Peng, & Hart, 2010; Kircher & Garland 2005).

14.4.2 Generating Different Appearances

Finally, it is desirable to create variations in both appearance and behavior when simulating crowds. The appearance variation can be achieved using a multiple of different texture images, face geometry, body sizes (McDonnell, Larkin, Hernández, Rudomin, & O'Sullivan, 2009). Behavior variations can be produced by applying a different personality behavior for each character during steering simulation (Guy, Kim, Lin, & Manocha, 2011) or synthesizing motions variations from original motions (Lau, Bar-Joseph, & Kuffner, 2009).

14.4.3 Pathfinding and Steering

The next is crowd movement control. Naively animating the crowds to navigate in the environment will result in a monotonous behavior for each character. Moreover, intersections between characters cannot be prevented or resolved since the character is not aware of the environment. Therefore, a steering system is needed to coordinate the movements of each character to generate desired crowd behaviors (Reynolds, 1987, 1999). In general, the steering method needs to plan the valid path for each character based on the environment. The environment would include static obstacles as well as dynamically moving objects and other moving characters. For navigation in static environment, it is possible to precompute the maneuverable space and store them in a structure such as navigation mesh (Kallmann, 2010) or probabilistic roadmap (Kavraki, Svestka, Latombe, & Overmars, 1996). These structures can be utilized at run time by steering system for efficient path planning. On the other hand, dynamic obstacles, such as other moving characters, can only be resolved at run time. Reactive methods have the character check the adjacent environment periodically and try to move away from obstacles when close to an obstacle (Reynolds, 1999). Predictive method such as reciprocal velocity obstacles (RVO) (Van den Berg, Lin, & Manocha, 2008) controls each character to avoid each other before collision by anticipating the movements from incoming characters. For more complex situation such as resolving deadlocks inside a tunnel, more expensive space-time planning techniques are required (Levine, Lee, Koltun, & Popović, 2011). Crowd behavior can be simplified by organizing members into smaller units of crowds, groups and individuals (Musse & Thalmann, 2001).

In addition to environment navigation, the steering method can also be extended to simulate crowd movement behavior to adapt for stressful situation (Kim, Guy, Manocha, & Lin, 2012).

14.5 CONCLUSION: MAKING COMPROMISES

As we have noted, the creation of virtual humans faces a set of imposing challenges. What has made the realization of virtual humans feasible is the bounded context of the environment and task in which a human user interacts with them. Open-ended face-to-face interaction with a virtual human is currently considerably beyond the state of the art, especially in terms of representing the wealth of human experience and dialogue capabilities that a person brings to an open-ended interaction. However, if the user is performing a specific task with the virtual human, then the interaction becomes more constrained. For example, the virtual human Max can play with a user (Kopp, Sowa, & Wachsmuth, 2004). STEVE instructs a user how to operate a machine (Rickel & Johnson, 1997). Similarly, if the virtual environment's setting elicits well-defined associations, then the human user's behavior becomes more predictable. For example, the Gunslinger system (Hartholt, Gratch, Weiss, & The Gunslinger Team 2009) places a human user in a scenario from the American Old West, where the user plays a sheriff facing a gunslinger in a bar. The scenario draws on people's common experiences watching Hollywood films about the Old West to help constrain the way they interact with the various virtual humans they encounter.

Nevertheless, the design of these artifacts for specific tasks and scenarios requires an iterative design process. For example, early versions of the system will be used to explore how users interact and what they say and do. That may include the so-called Wizard of Oz studies where the virtual human is no more than a puppet controlled by a hidden person. Results of such explorations lead to refinements in the world and dialogue knowledge encoded in the virtual human or changes in the scenario that subtly constrain how users tend to interact with the system. Assuming sufficient data are acquired from these studies, machine learning techniques can be employed to create the dialogue models used by the virtual human.

REFERENCES

Argyle, M., & Cook, M. (1976). *Gaze and mutual gaze.* Cambridge, U.K.: Cambridge University Press.

Badler, N. I., & Tolani, D. (1996). Real-time inverse kinematics of the human arm. *Presence*, *5*(4), 393–401.

Baraff, D., & Witkin, A. (1998). Large steps in cloth simulation. *Proceedings of the 25th Annual Conference on Computer Graphics and Interactive Techniques* (pp. 43–54). Los Angeles, CA: ACM.

Bavelas, J. B. (1994). Gestures as part of speech: Methodological implications. *Research on Language and Social Interaction*, *27*(3), 201–221.

Bechara, A., Damasio, H., Damasio, A., & Lee, G. (1999). Different contributions of the human amygdala and ventromedial prefrontal cortex to decision-making. *The Journal of Neuroscience*, *19*(13), 5473–5481.

Bregler, C., Covell, M., & Slaney, M. (1997). Video rewrite: Driving visual speech with audio. *Proceedings of the 24th Annual Conference on Computer Graphics and Interactive Techniques*. Los Angeles, CA: ACM Press.

Bridson, R., Marino, S., & Fedkiw, R. (2003). Simulation of clothing with folds and wrinkles. *Proceedings of the 2003 ACM SIGGRAPH/Eurographics Symposium on Computer Animation* (pp. 28–36). Granada, Spain: Eurographics Association.

Busemeyer, J., Dimperio, E., & Jessup, R. (2007). Integrating emotional processes into decision-making models. In W. Gray (Ed.), *Integrated models of cognitive systems* (pp. 213–229). Cambridge, MA: MIT Press.

Buss, S. R., & Kim, J. S. (2005). Selectively damped least squares for inverse kinematics. *Journal of Graphics, GPU, and Game Tools*, *10*(3), 37–49.

Campos, J. J., Thein, S., & Owen, D. (2003). A darwinian legacy to understanding human infancy. *Annals of the New York Academy of Sciences*, *1000*(1), 110–134.

Canutescu, A. A., & Dunbrack, R. L. (2003). Cyclic coordinate descent: A robotics algorithm for protein loop closure. *Protein Science*, *12*(5), 963–972.

Casas, D., Tejera, M., Guillemaut, J., & Hilton, A. (2011). Parametric control of captured mesh sequences for real-time animation. *Proceedings of the Fourth International Conference on Motion in Games* (pp. 242–253). Edinburgh, U.K.: Springer.

Cassell, J. (2000). *Embodied conversational agents*. Cambridge, MA: MIT Press.

Choi, K. J., & Ko, H. S. (2005). Stable but responsive cloth. *ACM SIGGRAPH 2005 Courses* (p. 1). Los Angeles, CA: ACM.

Cordier, F., & Magnenat-Thalmann, N. (2002). Real-time animation of dressed virtual humans. *Computer Graphics Forum*, *21*(3), 327–335.

Di Giacomo, T., Moccozet, L., Kim, H., & Magnenat-Thalmann, N. (2007). State-of-the-art on level-of-detail for virtual human animation and representation. In L. de Floriani & M. Spagnuolo (Eds.), *In shape analysis and structuring*. Berlin, Germany: Springer.

Dias, J., & Paiva, A. (2005). Feeling and reasoning: A computational model for emotional characters. *Progress in Artificial Intelligence*, 3808, 127–140.

Ekman, P., & Friesen, W. V. (1969). The repertoire of nonverbal behavior: Categories, origins, usage, and coding. *Semiotica*, *1*, 49–98.

Etzmuss, O., Keckeisen, M., & Strasser, W. (2003). A fast finite element solution for cloth modelling. *Proceedings of the 11th Pacific Conference on Computer Graphics and Applications* (pp. 244–251). Canmore, Alberta, Canada: IEEE.

Feng, W. W., Kim, B. U., & Yu, Y. (2008). Real-time data driven deformation using kernel canonical correlation analysis. *ACM Transactions on Graphics (TOG)*, *27*(3), 91.

Feng, W. W., Kim, B. U., Yu, Y., Peng, L., & Hart, J. (2010). Feature-preserving triangular geometry images for level-of-detail representation of static and skinned meshes. *ACM Transactions on Graphics (TOG)*, *29*(2), 11.

Frank, R. (1988). *Passions within reason*. New York, NY: W. W. Norton & Company.

Fridlund, A. J. (1997). The new ethology of human facial expressions. In J. A. Russell & J. M. Fernandez-Dols (Eds.), *The psychology of facial expression* (p. 103). Cambridge, U.K.: Cambridge University Press.

Gratch, J., Rickel, J., André, E., Cassell, J., Petajan, E., & Badler, N. (2002). Creating interactive virtual humans: Some assembly required. *Intelligent Systems, IEEE*, *17*(4), 54–63.

Guy, S. J., Kim, S., Lin, M. C., & Manocha, D. (2011). Simulating heterogeneous crowd behaviors using personality trait theory. *Proceedings of the 2011 ACM SIGGRAPH/Eurographics Symposium on Computer Animation* (pp. 43–55). ACM, Wales, U.K.

Hartholt, A., Gratch, J., Weiss, L., & The Gunslinger Team. (2009). At the virtual frontier: Introducing gunslinger, a multi-character, mixed-reality, story-driven experience. In *Intelligent virtual agents, Vol: 5773*, *Lecture Notes in Computer Science* (pp. 500–501). Berlin, Germany: Springer.

Hatfield, E., Cacioppo, J. T., & Rapson, R. L. (1994). *Emotional contagion*. Cambridge, MA: Cambridge University Press.

Hecker, C., Raabe, B., Enslow, R. W., DeWeese, J., Maynard, J., & van Prooijen, K. (2008). Real-time motion retargeting to highly varied user-created morphologies. *ACM Transactions on Graphics (TOG)*, *27*(3), 27.

Heloir, A., & Kipp, M. (2009). EMBR—A realtime animation engine for interactive embodied agents. In *Intelligent Virtual Agents* (pp. 393–404). Berlin, Germany: Springer.

Hill, R. (1999). Modeling perceptual attention in virtual humans. *Proceedings of the 8th Conference on Computer Generated Forces and Behavioral Representation* (pp. 563–573). Orlando, FL.

Jimenez, J., Sundstedt, V., & Gutierrez, D. (2009). Screen-space perceptual rendering of human skin. *ACM Transactions on Applied Perception (TAP)*, *6*(4), 3–17.

Kallmann, M. (2008). Analytical inverse kinematics with body posture control. *Computer Animation and Virtual Worlds*, *19*(2), 79–91.

Kallmann, M. (2010). Navigation queries from triangular meshes. *Proceedings of the Third International Conference on Motion in Games* (pp. 230–241). Zeist, the Netherlands: Springer.

Kallmann, M., & Thalmann, D. (1999). A behavioral interface to simulate agent-object interactions in real time. *Computer Animation, Proceedings* (pp. 138–146). Geneva, Switzerland: IEEE.

Kavan, L., Collins, S., Žára, J., & O'Sullivan, C. (2007). Skinning with dual quaternions. *Proceedings of the 2007 Symposium on Interactive 3D Graphics and Games* (pp. 39–46). Seattle, WA: ACM.

Kavan, L., McDonnell, R., Dobbyn, S., Žára, J., & O'Sullivan, C. (2007). Skinning arbitrary deformations. *Proceedings of the 2007 Symposium on Interactive 3D Graphics and Games* (pp. 53–60). Seattle, WA: ACM.

Kavraki, L. E., Svestka, P., Latombe, J. C., & Overmars, M. H. (1996). Probabilistic roadmaps for path planning in high-dimensional configuration spaces. *IEEE Transactions on Robotics and Automation*, *12*(4), 566–580.

Kendon, A. (2000). Language and gesture: Unity or duality. In D. McNeill (Ed.), *Language and gesture* (pp. 47–63). Cambridge, MA: Cambridge University Press.

Khullar, S. C., & Badler, N. I. (2001, March 1). Where to look? Automating attending behaviors of virtual human characters. *Autonomous Agents and Multi-Agent Systems, 4*(1–2), 9–23. doi:10.1023/A:1010010528443

Kim, B. U., Feng, W. W., & Yu, Y. (2010). Real-time data driven deformation with affine bones. *The Visual Computer*, *26*(6–8), 487–495.

Kim, S., Guy, S. J., Manocha, D., & Lin, M. C. (2012). Interactive simulation of dynamic crowd behaviors using general adaptation syndrome theory. *Proceedings of the ACM SIGGRAPH Symposium on Interactive 3D Graphics and Games* (pp. 55–62). Costa Mesa, CA: ACM.

Kim, T., & Neumann, U. (2001). Opacity shadow maps. *Rendering techniques 2001*. Vienna, Austria: Springer.

Kim, Y., Hill Jr., R., & Traum, D. R. (2006). *A computational model of dynamic perceptual attention for virtual humans*. Los Angeles, CA: University of Southern California.

Kircher, S., & Garland, M. (2005). Progressive multiresolution meshes for deforming surfaces. *Proceedings of the 2005 ACM SIGGRAPH/Eurographics Symposium on Computer Animation* (pp. 191–200). Dublin, Ireland: ACM.

Kopp, S., Timo S., & Ipke, W. (2004). Imitation games with an artificial agent: From mimicking to understanding shape-related iconic gestures. *Gesture-based communication in human-computer interaction* (pp. 436–447). Berlin Heidelberg: Springer.

Lander, J. (1998). Making kine more flexible. *Game Developer Magazine*, *1*, 15–22.

Lander, J. (1999). Over my dead, polygonal body. *Game Developer Magazine*, *17*(1), 1–4.

Larsson, S., & Traum, D. R. (2000). Information state and dialogue management in the TRINDI dialogue move engine toolkit. *Natural Language Engineering*, *6*(3 & 4), 323–340.

Lau, M., Bar-Joseph, Z., & Kuffner, J. (2009). Modeling spatial and temporal variation in motion data. *ACM Transactions on Graphics (TOG)*, *28*(5), 171.

Lee, S., Sifakis, E., & Terzopoulos, D. (2009). Comprehensive biomechanical modeling and simulation of the upper body. *ACM Transactions on Graphics (TOG)*, *28*(4), 99.

Levine, S., Lee, Y., Koltun, V., & Popović, Z. (2011). Space-time planning with parameterized locomotion controllers. *ACM Transactions on Graphics (TOG)*, *30*(3), 23.

Lewis, J. P., Cordner, M., & Fong, N. (2000). Pose space deformation: A unified approach to shape interpolation and skeleton-driven deformation. *Proceedings of the 27th Annual Conference on Computer Graphics and Interactive Techniques* (pp. 165–172). New Orleans, LA: ACM Press.

Luebke, D., Watson, B., Cohen, J. D., Reddy, M., & Varshney, A. (2002). *Level of detail for 3D graphics*. New York, NY: Elsevier Science Inc.

Marsella, S., & Gratch, J. (2003). Modeling coping behavior in virtual humans: Don't worry, be happy. *Proceedings of the Second International Joint Conference on Autonomous Agents and Multiagent Systems. AAMAS'03* (pp. 313–320). New York, NY: ACM.

Marsella, S., Gratch, J., & Petta, P. (2010). Computational models of emotion. In K. R. Scherer, T. Bänziger, and E. B. Roesch (Eds.), *A blueprint for a affective computing: A sourcebook and manual.* Oxford, U.K.: Oxford University Press.

McDonnell, R., Larkin, M., Hernández, B., Rudomin, I., & O'Sullivan, C. (2009). Eye-catching crowds: Saliency based selective variation. *ACM Transactions on Graphics (TOG)*, *28*(3), 55.

Mohr, A., & Gleicher, M. (2003). Building efficient, accurate character skins from examples. *ACM Transactions on Graphics (TOG)*, *22*(3), 562–568.

Müller, M., Heidelberger, B., Hennix, M., & Ratcliff, J. (2007). Position based dynamics. *Journal of Visual Communication and Image Representation*, *18*(2), 109–118.

Musse, S. R., & Thalmann, D. (2001). Hierarchical model for real time simulation of virtual human crowds. *IEEE Transactions on Visualization and Computer Graphics*, *7*(2), 152–164.

Ng-Thow-Hing, V., & Fiume, E. (1997). Interactive display and animation of B-spline solids as muscle shape primitives. *Computer Animation and Simulation'97*. Vienna, Austria: Springer.

Pamplona, V. F., Oliveira, M. M., & Baranoski, G. V. (2009). Photorealistic models for pupil light reflex and iridal pattern deformation. *ACM Transactions on Graphics (TOG)*, *28*(4), 106.

Paiva, A., Dias, J., Sobral, D., Aylett, R., Woods, S., Hall, L., & Zoll, C. (2005). Learning by feeling: Evoking empathy with synthetic characters. *Applied Artificial Intelligence*, *19*(3–4), 235–266.

Poggi, I., Pelachaud, C., de Rosis, F., Carofiglio, V., & De Carolis, B. (2005). Greta: A believable embodied conversational agent. In O. Stock, & M. Zancarano (Eds.), *Multimodal intelligent information Presentation* (pp. 3–25). Dordrecht, the Netherlands: Springer.

Reynolds, C. W. (1987). Flocks, herds and schools: A distributed behavioral model. *ACM SIGGRAPH Computer Graphics*, *21*, 25–34.

Reynolds, C. W. (1999). Steering behaviors for autonomous characters. *Game Developers Conference*. San Francisco, CA. http://www. Red3d. Com/cwr/steer/gdc99

Rickel, J., & Johnson, W. L. (1997). Integrating pedagogical capabilities in a virtual environment agent. *Proceedings of the First International Conference on Autonomous Agents* (pp. 30–38). New York, NY.

Rickel, J., & Johnson, W. L. (2000). Task-oriented collaboration with embodied agents in virtual worlds. In J. Cassell, J. Sullivan, S. Prevost, & E. Churchill (Eds.), *Embodied conversational agents* (pp. 95–122). Cambridge, MA: MIT Press.

Rickel, J., Marsella, S., Gratch, J., Hill, R., Traum, D., & Swartout, W. (2002). Toward a new generation of virtual humans for interactive experiences. *Intelligent Systems, IEEE*, *17*(4): 32–38.

Scherer, S., Glodek, M., Layher, G., Schels, M., Schmidt, M., Brosch, T., Tschechne, S., … Palm, G. (2012). A generic framework for the inference of user states in human computer interaction: How patterns of low level communicational cues support complex affective states. *Journal on Multimodal User Interfaces, Special Issue on: Conceptual Frameworks for Multimodal Social Signal Processing*, *6*(3), 117–141. doi:DOI 10.1007/s12193-012-0093-9

Shapiro, A. (2011). Building a character animation system. *Proceedings of the 4th International Conference on Motion in Games* (pp. 98–109). Edinburgh, U.K.: Springer.

Shoulson, A., Marshak, N., Kapadia, M., & Badler, N. I. (2013). ADAPT: The agent development and prototyping testbed. *Proceedings of the Symposium on Interactive 3D Graphics and Games, I3D*. Orlando, FL: ACM.

Simon, H. A. (1967). Motivational and emotional controls of cognition. *Psychological Review*, *74*(1), 29–39.

Starck, J., & Hilton, A. (2007). Surface capture for performance-based animation. *Computer Graphics and Applications, IEEE*, *27*(3), 21–31.

Starck, J., Miller, G., & Hilton, A. (2005). Video-based character animation. *Proceedings of the 2005 ACM SIGGRAPH/Eurographics Symposium on Computer Animation (SCA '05)* (pp. 49–58). New York, NY: ACM. doi:10.1145/1073368.1073375 http://doi.acm.org/10.1145/1073368.1073375

Thalmann, D. (1990). Robotics methods for task-level and behavioral animation. *Scientific Visualization and Graphics Simulation* (pp. 129–147). Chichester, UK: John Wiley.

Thiebaux, M., Marsella, S., Marshall, A. N., & Kallmann, M. (2008). Smartbody: Behavior realization for embodied conversational agents. *Proceedings of the 7th International Joint Conference on Autonomous Agents and Multiagent Systems* (Vol. 1, pp. 151–158). Estoril, Portugal: International Foundation for Autonomous Agents and Multiagent Systems.

Tolani, D., Goswami, A., & Badler, N. I. (2000). Real-time inverse kinematics techniques for anthropomorphic limbs. *Graphical Models*, *62*(5), 353–388.

Traum, D., Swartout, W., Gratch, J., & Marsella, S. (2008). A virtual human dialogue model for non-team interaction. In L. Dybkjoer & W. Minker (Eds.), *Recent trends in discourse and dialogue* (pp. 45–67). Dordrecht, the Netherlands: Springer. http://link.springer.com/chapter/10.1007/978-1-4020-6821-8_3

Traum, D. R., & Larsson, S. (2003). The information state approach to dialogue management. *Current and New Directions in Discourse and Dialogue*, *212*, 325–353.

Van den Berg, J., Lin, M., & Manocha, D. (2008). Reciprocal velocity obstacles for real-time multi-agent navigation. *IEEE International Conference on Robotics and Automation, 2008. ICRA 2008* (pp. 1928–1935). Pasadena, CA: IEEE.

van Welbergen, H., Reidsma, D., Ruttkay, Z. M., & Zwiers, J. (2009). Elckerlyc. *Journal on Multimodal User Interfaces*, *3*(4), 271–284.

Waal, F. (2003). Darwin's legacy and the study of primate visual communication. *Annals of the New York Academy of Sciences*, *1000*(1), 7–31.

Wang, R. Y., Pulli, K., & Popović, J. (2007). Real-time enveloping with rotational regression. *ACM Transactions on Graphics (TOG)*, *26*(3), 73.

Watt, M., Cutler, L. D., Powell, A., Duncan, B., Hutchinson, M., & Ochs, K. (2012). LibEE: A multithreaded dependency graph for character animation. *Proceedings of the Digital Production Symposium* (pp. 59–66). Glendale, CA: ACM.

Yamane, K., & Nakamura, Y. (2003). Natural motion animation through constraining and deconstraining at will. *IEEE Transactions on Visualization and Computer Graphics*, *9*(3), 352–360.

Yeh, T. Y., Faloutsos, P., & Reinman, G. (2006). Enabling real-time physics simulation in future interactive entertainment. *Proceedings of the 2006 ACM SIGGRAPH Symposium on Videogames* (pp. 71–81). Boston, MA: ACM.

Yersin, B., Maïm, J., Pettré, J., & Thalmann, D. (2009). Crowd patches: Populating large-scale virtual environments for real-time applications. *Proceedings of the 2009 Symposium on Interactive 3D Graphics and Games* (pp. 207–214). Boston, MA: ACM.

Zeller, C. (2005). Cloth simulation on the GPU. *ACM SIGGRAPH 2005 Sketches* (p. 39). Los Angeles, CA: ACM.