Simulink Toolbox for Real-time Virtual Character Control

Ulysses Bernardet✉, Maryam Saberi, and Steve DiPaola

Simon Fraser University Vancouver, Canada
{ubernard,msaberi,sdipaola}@sfu.ca

Abstract. Building virtual humans is a task of formidable complexity. We believe that, especially when building agents that interact with biological humans in real-time over multiple sensorial channels, graphical, data flow oriented programming environments are the development tool of choice. In this paper, we describe a toolbox for the system control and block diagramming environment Simulink that supports the construction of virtual humans. Available blocks include sources for stochastic processes, utilities for coordinate transformation and messaging, as well as modules for controlling gaze and facial expressions.

Keywords: Virtual character · Simulink · Toolbox · Data Flow graphical programming · Real-time interaction

1 Introduction

We operate under the assumption that, to build truly lifelike, autonomous humanoid agents that are indistinguishable from biological humans, we need to not only copy the "surface properties" of the biological model system but make a "deep copy" that is based on the same underlying principles. Even if we build IVAs at a more abstract level than the neuronal one, we are still facing the task of constructing highly complex systems. Graphical data flow programming environments are firmly established in the design of industrial control systems and are equally well suited for building complex models at different levels of abstraction [1]. In this paper, we present ongoing work on the development of a toolbox that supports the construction of real-time control systems for virtual characters using the graphical programming environment MathWorks Simulink. We are motivated by the experience that control systems for virtual humans tend to become complex rapidly and that graphical tools are very useful in supporting the construction by providing a better overview and understanding of what is happening within the system. Our goal is to provide a library of re-usable components e.g. for gaze control and facial expression. The toolbox is open source and the code available from https://github.com/bernuly/VCSimulinkTlbx.

2 A Simulink library for virtual character control

A number of open and closed source environments support the graphical construction of control systems, e.g. Dymola and OpenModelica respectively. General purpose graphical data flow oriented environments include LabView, which in the domain of interactive media MAX is widely used. The Simulink environment is one of the best-established block diagram environment for continuous and discrete domain simulations. Unlike other data flow oriented programming environments, Simulink natively integrates a finite state machine component (StateFlow), allowing to construct hybrid control systems within a single application. Key advantages, next to general graphical system construction and the availability of real-time data visualization tools, are encapsulation and introspection. Encapsulation means that the system can be organized into an arbitrarily deeply nested hierarchy of subsystems, hence shielding the user from complexity he/she is not currently not interested in. However, unlike in the case of code level encapsulation, the subsystems created within Simulink are open to introspection, granting the user access to the system at all levels of abstraction.

2.1 Available blocks

The blocks in the toolbox fall into the broad categories of "input/sources", utilities, and output behavior generation. The toolbox is neutral regarding the transportation layer, but examples shown here are using the m+m middleware [2].

Sources In most of the cases, input into the system will be provided by external processes such as tracking system, analysis of facial expression etc. For typical model intrinsic sources such as step functions and sine waves, Simulink provides blocks.

Poisson pulse A number of natural processes follow the temporal characteristics of a Poisson process [3]. The "Poisson pulse" block provides an input sequence where the delay between (binary) events follows a Poisson distribution. The only parameter used in this block is the λ that controls the shape of the distribution.

Utilities *3D coordinate transformation* Transformation between different coordinate system is an issue frequently encountered when working with 3-dimensional data. This block converts between input coordinates, their respective centers and scaling factors and output coordinates.

Messaging Simulink does not allow to use strings as signals between blocks in the model. To be able to pass character values, e.g. BML messages from one block to the next, they need to be converted to numeric arrays. We provide a utility function ("encStr2Arr") that can be used to convert strings to double array at the level of Matlab code. All blocks of our toolbox, however, handled the conversion transparently. This means that the user does not need to explicitly convert between types, but can simply enter textual information directly.

createMessage This block creates text messages that are used for triggering atomic actions such as animations.

conc msg Often different nodes inside the model will create complementary control message, e.g. if one subsystem generates gaze control commands while the other determines the facial expression of the agent. To send these commands in a single message, they need to be strung together. The "conc msg" block concatenates messages in a vector into a single output string.

Add prefix/suffix This block allows to add a prefix and suffix to message e.g. to add "bracket" a message with <?xml version="1.0" ?><act><bml> and </bml></act> as is required by some BML realizers such as SmartBody.

Behavior control At the output side the toolbox is generating control messages in the Behavioral Markup Language (BML)[4]. By using the BML standard, the toolbox should be compatible with wide range of BML realizers such as Greta [5] or AsapRealizer [6]. However, the toolbox so far has only been tested with the SmartBody character animation system [7].

facial Expression This block generates BML face commands with action unit weights for the categorical facial expression of anger, disgust, fear, joy, sadness, surprise. Each category is given a weight ($[0 \ldots 1]$), and several categories can be combined.

PAD2AU female/male The circumplex model of affect proposed by Russel [8], is a widely used representation of internal affective states. When creating affective facial expressions, we need to map the circumplex dimensions of pleasure, arousal, and dominance (PAD) onto facial action units. [9] empirically determined this mapping. The "PAD2AU female/male" block uses the regression weights from [9] to map locations in the 3-dimensional PAD space onto facial action units for females and males.

gaze control This block allows direct control over the target location for the gaze. Additionally, the "extent" parameter determines which joints are involved in the gaze behavior ($0 < e < .25$: eyes only, $.25 < e < 5$: eyes, neck, $.5 < e < 75$: eyes...chest, $.75 < e < 1$: eyes...back).

Mutual/non-mutual gaze control This block form mutual and non-mutual gaze behavior provides high-level control for a relatively complex gaze behavior, where the agent is alternating between looking at a predetermined, e.g. speaker location, and a randomly chosen alternative location. The control logic is loosely based on the system described in [10]. The random distribution of the gaze dwell times presented in [10] in our block is approximated using two Poisson processes, one for mutual, and one for non-mutual gaze. This has the advantage of providing two easily tunable parameters in the form of the λ of the Poisson distribution. The non-mutual gaze direction is drawing its horizontal and vertical saccade amplitude from a normal distribution with the fixed location as the mean. Tuning the variance for the horizontal and vertical distributions, the user can easily shape the gaze frustum. As the direct gaze control block described above, this block also allows the control of the joints involved in the gazing behavior by means of the “extent”.

3 Conclusion

In this paper, we have presented an open source Simulink toolbox that supports the graphical design and control of real-time virtual human systems. We believe that real-time data inspection and visualization, in combination with the ability to organize the system into encapsulated subsystems are powerful methods for tackling the complexity of virtual human control systems. In the future, we will expand the toolbox with additional generic block e.g. for procedural animation control, add support for middleware components such as Apache ActiveMQ, and interface to a wider range of BML realizers.

Acknowledgments

This work was partially supported by ”Moving Stories” Canadian SSHRC grant.

References

1. Bernardet, U., Verschure, P.F.: iqr: A Tool for the Construction of Multi-level Simulations of Brain and Behaviour. Neuroinformatics **8**(2) (2010) 113–34
2. Bernardet, U., Schiphorst, T., Adhia, D., Jaffe, N., Wang, J., Nixon, M., Alemi, O., Phillips, J., DiPaola, S., Pasquier, P.: m+m: A Novel Middleware for Distributed, Movement Based Interactive Multimedia Systems. In: Proceedings of the 3rd International Symposium on Movement and Computing - MOCO ’16, New York, New York, USA, ACM Press (2016) 21:1–21:9
3. Weisstein, E.W.: Poisson Process
4. Reidsma, D., Welbergen, H.V.: BML 1.0 Standard. SAIBA. Retrieved from http://www. mindmakers. org/projects/bml-1-0/wiki
5. Mancini, M., Niewiadomski, R., Bevacqua, E., Pelachaud, C.: Greta: a SAIBA compliant ECA system. In: Troisiéme Workshop sur les Agents Conversationnels Animés. (2008)
6. Van Welbergen, H., Yaghoubzadeh, R., Kopp, S.: AsapRealizer 2.0: The next steps in fluent behavior realization for ECAs. Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics) (2014) 449–462
7. Shapiro, A.: Building a character animation system. In: Motion in Games. (2011) 98–109
8. Russell, J.A.: A Circumplex Model of Affect. Journal of Personality and Social Psychology **39**(6) (1980) 1161–1178
9. Boukricha, H., Wachsmuth, I., Hofstatter, A., Grammer, K.: Pleasure-arousal-dominance driven facial expression simulation. In: 2009 3rd International Conference on Affective Computing and Intelligent Interaction and Workshops, IEEE (sep 2009) 1–7
10. Lee, S.P., Badler, J.B., Badler, N.I.: Eyes alive. ACM Transactions on Graphics **21**(3) (2002) 637–644