

Co-simulation framework for Networked Control Systems over multi-hop mobile ad-hoc networks

M Shahidul Hasan, Hongnian Yu, Alison Griffiths, and T C Yang*

*Faculty of Computing, Engineering and Technology, Staffordshire University, UK
(e-mail: {m.s.hasan, h.yu, a.l.griffiths}@staffs.ac.uk).*

**Department of Engineering and Design, University of Sussex, UK*

Abstract: Networked Control Systems over mobile ad-hoc networks have drawn the attention of the researchers because of its suitability in various ad-hoc applications. This paper presents the investigation of such systems using both real wireless communication and co-simulation of MATLAB and OPNET. The plant and the wireless node models are simulated on a computer by MATLAB and OPNET, respectively. The controller runs on a laptop. These two computers communicate using a real wireless link. The interactive co-simulation is applied to a double-pendulum plant with two sensors and two actuators. Both the co-simulation technique and the results are presented and discussed.

1. INTRODUCTION

Networked Control Systems (NCS) are now being implemented over wireless networks because of the latest development of high speed and reliable wireless communication technologies. These systems are known as Wireless Networked Control Systems (WNCS). The simplest WNCS includes a plant and a controller with point to point wireless communication between them. An advanced version of WNCS applies the control mechanism over multi-hop Mobile Ad-hoc Networks (MANET). These networks offer very dynamic and flexible wireless networks; these are self-organising and can be easily deployed without any infrastructure (Royer and Toh, 1999). Nodes perform routing operations and are capable of maintaining the dynamic topology of the network. Routes can have multiple hops between the source and the destination node as shown in Fig. 1. WNCS over MANET are suitable for various applications, for instance, military or rescue mission, exploring hazardous environment etc. However, this area of research has brought many challenges, such as unpredictable packet delay and dropouts, random node movement etc. to the researchers because of the unpredictable behaviours of the MANET.

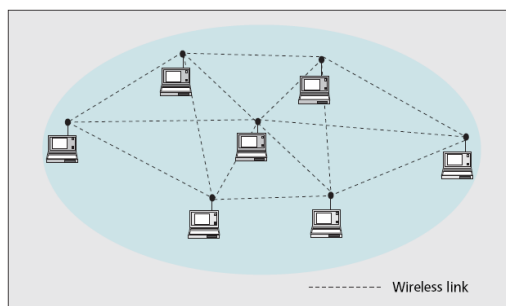


Fig. 1. MANET (Conti and Giordano, 2007b).

Research on wireless networks, e.g. MANET mostly rely on simulation studies since launching real experiments are expensive and time consuming (Kotz et al., 2004), (Conti and Giordano, 2007b). Optimised Network Engineering Tool (OPNET) is a simulation package that allows detailed communication network simulation (Chang, 1999). Many aspects of the network such as the number of nodes, network data rate, node movement etc. can be specified in OPNET. However, it is a tedious task to implement dynamic models and control algorithms of systems in real-time using the Proto-C language of OPNET. On the other hand, MATLAB/SIMULINK is a very powerful tool for modelling systems and implementing control algorithms. However, it has limitations in simulating computer networks. Therefore, combining the strengths of OPNET and MATLAB will accelerate the WNCS research to produce more realistic simulation results. Many research works consider offline co-simulation where the output data from one simulation package is stored in a file. Then the other simulation package reads the data file and generates the final results. Unlike those papers, under this interactive co-simulation, both MATLAB and OPNET execute in parallel interactively in a synchronised fashion. The major contributions of the paper are listed below.

- Developing a co-simulation framework that combines wireless communication hardware with simulation packages.
- Using wireless ad-hoc network to carry real time data.
- Applying the distributed NCS architecture using multiple sensors and a single controller.
- Implementing a realistic wireless communication model using OPNET simulation.
- Using interactive co-simulation of SIMULINK and OPNET to simulate the plant model and network model, respectively.

The rest of the paper is organised as follows. Section 2 discusses the related work, section 3 explains the models used in the simulation. Section 4 presents the results and finally some conclusions are drawn in section 5.

2. RELATED WORKS

The investigation of WNCS performance based on 802.11B protocol for the inverted pendulum and tracking problem can be found in (Ploplys, 2003), (Ploplys et al., 2004). TrueTime (Cervin et al., 2003), (Cervin et al., 2002) is a MATLAB/SIMULINK based toolbox that allows performance evaluation of NCS with the support of wired and wireless network protocols. However, the network blocks have limited support for detailed network simulation. Simulation of WNCS using TrueTime can be found in (Andersson et al., 2005). The interface between MATLAB and OPNET has been considered in (Dham, 2003). A co-simulation of control and network, implemented in MATLAB/SIMULINK, is presented in (Colandairaj et al., 2005) that investigated NCS performance for various data rates, traffic, load etc. A general profile of wireless fieldbus has been introduced in (Pellegrini et al., 2006). The physical and data link layers of the proposed profile have been taken from existing wireless local area network and personal area network whereas the application layer is derived from wired fieldbus to provide better reliability. An overview of implementation of wireless networks in industrial applications can be found in (Willig et al., 2005). The simulation of control mechanism over MANET for a simple first order system (water level control) has been discussed in (Hasan et al., 2005). Simulation study of NCS over MANET using OPNET can be found in (Hasan et al., 2007). The Simulation of control of an inverted pendulum on a cart over wireless network is discussed and compared with hardwired control in (Colandairaj et al., 2006). Paper (Al-Hammouri et al., 2007) presents a co-simulation platform for NS-2 and Modelica. NS-2 and Modelica are simulation packages to simulate computer networks and large-scale physical systems, respectively.

3. SIMULATION MODEL

The co-simulation configuration of this paper using the wireless link, OPNET and MATLAB is shown in Fig. 2. The top and bottom parts of the figure show the physical setup and the logical view, respectively. At the physical level, the co-simulation involves a desktop PC and a laptop connected by a wireless link. At the logical level, the PC executes the MATLAB/SIMULINK plant and the OPNET multi-hop MANET models. The OPNET plant model communicates to the laptop that runs the controller through a real wireless link. The wireless communication between the OPNET MANET model and the controller is implemented using socket and user datagram (UDP) protocol. UDP is connection less transport protocol that does not need to create or maintain connections between the source and destination node. Moreover, it exhibits a robust routing policy as packets can follow different routes depending on the current status of the network. It does not have any retransmission mechanism for failed transmission. UDP offers low overheads and discards

obsolete or lost packets, therefore, it is preferable for networked control applications (Ploplys et al., 2004). UDP is also chosen to validate the simulation results for wireless networks (Liu et al., 2004).

The OPNET plant model sends a simulated packet containing the state information to the gateway node (e.g. node 3) during sampling. The gateway node puts the information from the simulated packet into a real time datagram packet and sends the datagram packet to the controller running on the laptop over the wireless link. Again when the gateway node receives a real time datagram packet from the controller, it creates a simulated packet, puts the control information into it and sends it to the plant model for actuation. Both the OPNET simulated packet and the real time packet delays are used to produce the total delay for the closed loop control mechanism.

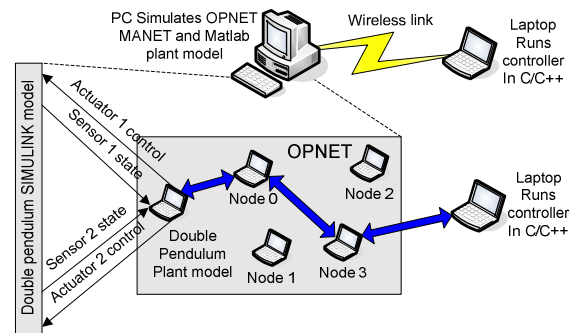


Fig. 2. Co-simulation framework of the real wireless link, OPNET and MATLAB.

3.1 Plant/Controller model: Double pendulum coupled by a spring

In many of research works numeric examples are used to evaluate performance of the system. Unlike those works, this paper considers a benchmark case plant model (Ikeda and Siljak, 1990), (Siljak, 1991) that implements the distributed nature of NCS as shown in Fig. 3. The system constants and variables for the model are given in Table 1. The plant model is based on the mathematical equations (1-5) and the decentralised PID control law is given in (6). It is assumed that the mass of each pendulum is uniformly distributed and the mass of the spring is zero. The length of the spring is chosen so that $F = 0$ when $\theta_1 = \theta_2$ which implies

$(\theta_1 \ \theta_1 \ \theta_2 \ \theta_2)^T = 0$ is an equilibrium of the system if $\tau_1 = \tau_2 = 0$. The initial conditions of the two pendulums are noted as $\theta_1(0) = x_{01}$ $\theta_2(0) = x_{02}$. In this model, if any angle of the pendulums exceeds 60 degrees (1.04 radians) from their central positions, the simulation will stop and the system is considered as unstable.

The states of the pendulums are sent at different sampling rates to the controller through two different wireless channels. The control objective of the system is to keep both the pendulums upright or to follow a particular reference/trajectory by applying the controls to the both

actuators separately as depicted in Fig. 3. This architecture guarantees that the system has a distributed structure with two sensors and two actuators. The challenging issues are to maintain suitable communication network packet delays, packet losses etc. so that the system does not become unstable.

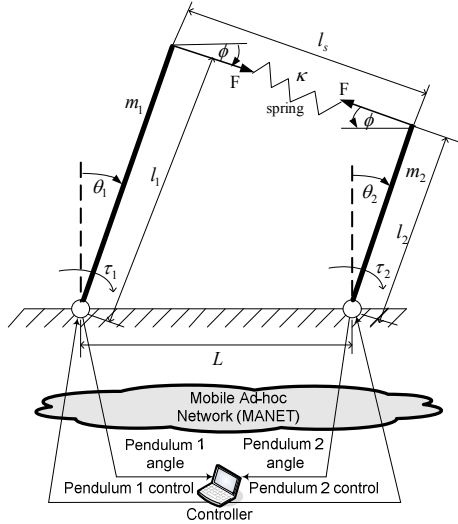


Fig. 3. Double pendulum coupled by a spring control.

Table 1. Double pendulum model variables and constants.

Symbol	Description
θ_i	Angular displacement of pendulum i ($i=1, 2$)
τ_i	torque input generated by the actuator for pendulum i ($i=1, 2$)
F	Spring force
l_s	Spring length
ϕ	Slope of the spring to the earth
l_i	Length of pendulum i ($i=1, 2$)
m_i	Mass of pendulum i ($i=1, 2$)
L	Distance of two pendulums
κ	Spring constant

$$[m_1(l_1)^2 / 3] \ddot{\theta}_1 = \tau_1 + m_1 g (l_1 / 2) \sin \theta_1 + l_1 F \cos(\theta_1 - \phi) \quad (1)$$

$$[m_2(l_2)^2 / 3] \ddot{\theta}_2 = \tau_2 + m_2 g (l_2 / 2) \sin \theta_2 - l_2 F \cos(\theta_2 - \phi) \quad (2)$$

$$F = \kappa (l_s - [L^2 + (l_1 - l_2)^2]^{1/2}) \quad (3)$$

$$l_s = [(L + l_2 \sin \theta_2 - l_1 \sin \theta_1)^2 + (l_2 \cos \theta_2 - l_1 \cos \theta_1)^2]^{1/2} \quad (4)$$

$$\phi = \tan^{-1} \frac{l_1 \cos \theta_1 - l_2 \cos \theta_2}{L + l_2 \sin \theta_2 - l_1 \sin \theta_1} \quad (5)$$

$$\begin{bmatrix} \tau_1 \\ \tau_2 \end{bmatrix} = - \begin{bmatrix} 11.327 & 3.001 & 0 & 0 \\ 0 & 0 & 7.594 & 1.861 \end{bmatrix} \begin{bmatrix} \theta_1 \\ \dot{\theta}_1 \\ \theta_2 \\ \dot{\theta}_2 \end{bmatrix} \quad (6)$$

3.2 SIMULINK-OPNET interactive co-simulation

In the interactive co-simulation environment, OPNET executes as the master simulator and maintains the simulation

time. The OPNET plant node invokes a MATLAB engine server to execute the plant SIMULINK model as shown in Fig. 4. During the sampling task, the state of the particular sensor is read from the SIMULINK model by the OPNET plant model to generate the state packet. When a control packet arrives at the plant, it is passed to the corresponding actuator. As OPNET and SIMULINK maintain independent simulation time concepts, their times have to be synchronised to run them interactively. The synchronisation mechanism between the OPNET network and the SIMULINK plant is explained in Fig. 5. OPNET begins execution and pauses at simulation time 0. The OPNET plant node model invokes the corresponding SIMULINK model. After initialisation the SIMULINK plant model pauses at time 0. OPNET resumes execution and pauses at sampling time T1. It passes a command to SIMULINK to execute until SIMULINK time T1. When SIMULINK pauses at time T1, the OPNET plant node model reads the plant state from SIMULINK and generates a sample packet.

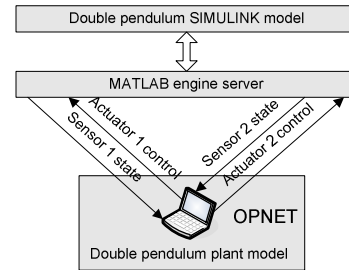


Fig. 4. Interactive OPNET and SIMULINK co-simulation.

Upon receiving a control packet at time $T1 + \tau_1$, OPNET issues a command to SIMULINK to execute until SIMULINK time $T1 + \tau_1$ with previous input u_0 and then change input to u_1 . When SIMULINK finishes execution, OPNET continue to run the simulation in this fashion. The same synchronisation mechanism has been used for co-simulation of NS2 and Modelica in (Al-Hammouri et al., 2007).

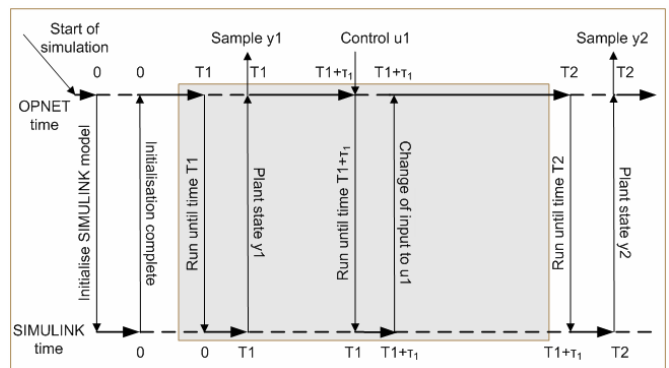


Fig. 5. Synchronisation mechanism of OPNET and SIMULINK time for the interactive co-simulation.

3.3 MANET environment

One of the aims in this paper is to implement a realistic wireless signal propagation model for WNCs over MANET. A comparison between computer simulation and real world wireless network experiments can be found in (Kotz et al., 2004), (Liu et al., 2004), (Newport, 2004). Three different radio signal propagation models have been investigated in (Kotz et al., 2004) as shown in Fig. 6. *Model 1* involves two components: path loss exponent and fading. *Model 2* is the two-ray-ground reflection model that uses only the path loss component. Finally, *model 3* represents the ideal propagation model. The comparison, shown in Fig. 6, revealed that *model 1* exhibits the closest behaviour to the real world experiment (Kotz et al., 2004).

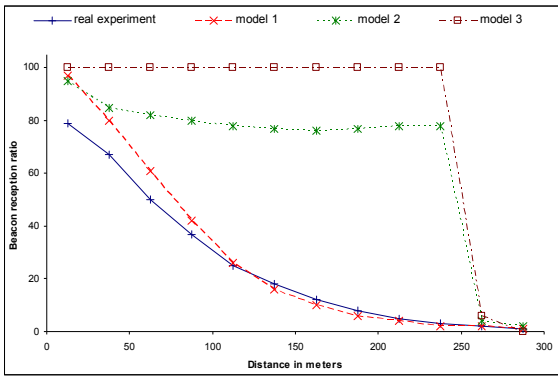


Fig. 6. Comparison of three simulation models with real world experiment (Kotz et al., 2004).

The radio propagation model used in this paper considers both path loss and fading (*model 1*) to achieve very realistic signal propagation. The model is expressed in (7) where P_r is the received wireless signal power, β is path loss exponent, d is the distance between the transmitter and the receiver, d_0 is the reference distance and X_{dB} is a Gaussian random variable with zero mean and standard deviation σ_{dB} . Here σ_{dB} is called fading deviation that can be obtained by measurement. This model extends *model 3* to a sensible statistic model in which nodes communicate probabilistically at the edge of the communication range (Fall and Varadhan, 2006). The simulation model implements IEEE 802.11b technology that can support up to 11 Mbps data rate. A square open field of size 174m by 174m with twelve MANET nodes (eleven OPNET nodes and the controller laptop) equipped with wireless network cards are taken into consideration. Under 802.11 technologies, multi-hop ad-hoc network exists at two-three hops and ten-twenty nodes (Conti and Giordano, 2007b), (Conti and Giordano, 2007a). To reflect the open field environment, path loss exponent, $\beta=2.8$ and $X_{dB}=6$ dB fading effect have been implemented in the OPNET simulation as suggested in (Kotz et al., 2004), (Liu et al., 2004), (Newport, 2004).

$$\left[\frac{P_r(d)}{P_r(d_0)} \right]_{dB} = -10\beta \log\left(\frac{d}{d_0}\right) + X_{dB} \quad (7)$$

Both the PC and the laptop are equipped with Belkin wireless USB network cards that support 802.11x technologies. The cards were setup in ad-hoc mode and the wireless link supported up to 11Mbps data rate between the PC and laptop. The experiments are carried out in the research area located in the Octagon building at Staffordshire University with many pieces of interfering equipment such as other computers, PDAs etc. The wireless link produced reliable communication within a distance of a few meters.

3.4 Simulated and real time packet format

The OPNET gateway node converts the simulated packet into real time datagram packets and vice versa. In 802.11 protocols each packet has a 34 byte MAC and 24 byte PHY header (Colandairaj et al., 2006). Each pendulum state can be included in an 8-byte field. The plant id, sensor id and sequence number can be transmitted using three separate 4-byte fields. Therefore the total information for a state or control packet can be contained in an 86-byte packet as depicted in Fig. 7.

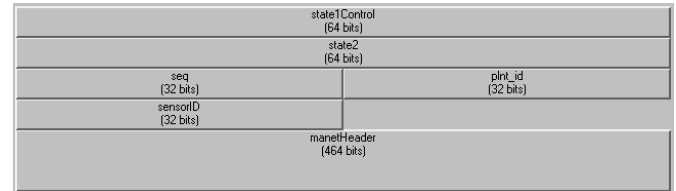


Fig. 7. Simulated/real time packet format for state/control.

3.5 Assumptions

The following assumptions have been made in this paper.

- All the state and control information can be carried by an 86 byte packet. Small packet size guarantees short packet transmission time.
- The current control packet arrives at the plant before the next sampling packet is created. No delay compensation mechanism is implemented. Therefore, the total control loop delay must not exceed the sampling period. For the packet arriving after the sampling period, it will be treated as the packet loss.
- Sampling, actuation and control computation times are negligible compared to network delays. These computation tasks can be executed in the order of microseconds on most modern computers

4. RESULTS

A pulse signal of amplitude 5 radians, period 1.5s and 10% pulse width has been applied as the reference signal. Two simulation sequences are executed with sampling periods of (0.05s, 0.06s) and (0.08s, 0.09s) for pendulums 1 and 2. The following sub-sections present the results for the total closed loop delay and pendulum angles.

4.1 Total delay

The total delays of the closed loop control for the two simulation sequences are shown in Fig. 8. It is obtained by adding the sensor-to-controller and controller-to-actuator delays. It is noted that the most of the total delays stayed below 0.08s. However, for some packets, the delays were larger than the sampling periods. These control packets are considered as packets dropouts and are discarded by the plant.

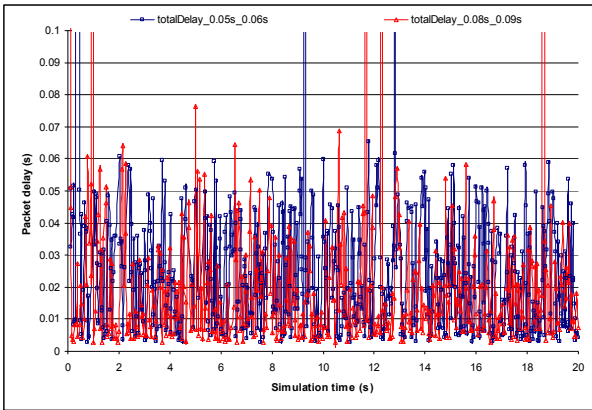


Fig. 8. Total closed loop delay.

4.2 Pendulum angle

The angle of the pendulum 1 under direct control is compared with the angles with sampling periods 0.05s and 0.08s under WNCS in Fig. 9. The direct control (MATLABPend1) and WNCS control (MATLABOpnetPend1_0.05s, MATLABOpnetPend1_0.08s) were implemented using only SIMULINK and wireless-link-SIMULINK-OPNET co-simulation, respectively. It is noted that when WNCS control is applied with 0.05s sampling period, the pendulum 1 showed several angle fluctuations during the simulation. This is because many control packets could not arrive at the plant before the next sampling and they were discarded. For 0.08s sampling, control packets had higher probability to reach the plant on time. Therefore, it is more stable and showed better performance than the 0.05s sampling. However, the system maintained the overall stability and the pendulum angle stayed within 1.04 radians as discussed in section 3.1.

The comparison of the angle of pendulum 2 for direct control (MATLABPend2) and control over WNCS (MATLABOpnetPend2_0.06s, MATLABOpnetPend2_0.09s) is shown in Fig. 10. Again the sampling period of 0.06s exhibited angle fluctuations as the control packet delays were very close to the sampling period. On the other hand, sampling period of 0.09s showed a more stable performance.

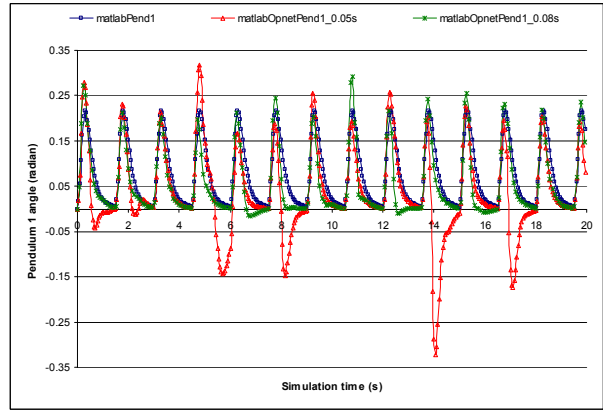


Fig. 9. Comparison of pendulum 1 angles for direct control (MATLAB) and WNCS (online co-simulation of MATLAB and OPNET).

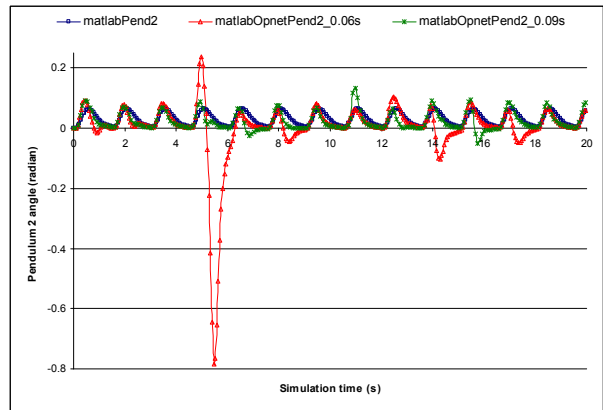


Fig. 10. Comparison of pendulum 2 angles for direct control (MATLAB) and WNCS (online co-simulation of MATLAB and OPNET).

The packet routes through the MANET are shown in Fig. 11. The top left corner node functions as the gateway node between the controller and the OPNET MANET simulation. The gateway node could maintain communication to the OPNET plant through one intermediate node (node_3). This route was established by the DSR routing protocol (Johnson et al., 2001) used in OPNET.

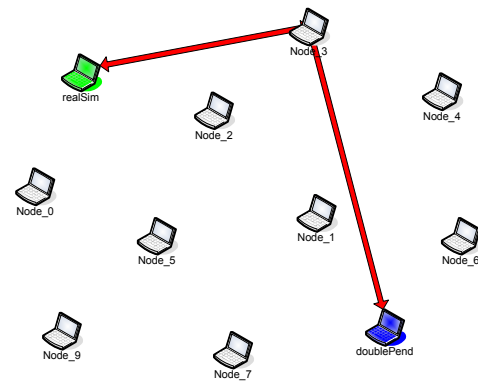


Fig. 11. Packet routes in the MANET.

5. CONCLUSIONS

The main focus of this paper is to implement an interactive co-simulation environment for MATLAB, OPNET and real wireless network to obtain realistic simulation results. To synchronise the simulation times from both packages, OPNET is executed as the master process and MATLAB is as child process. The plant is modelled by SIMULINK and run by MATLAB engine server. The double pendulum coupled by a spring plant could maintain stability over the wireless network under the co-simulation. Both the pendulum angles stayed within 1.04 radians that is defined by the stability condition of the system. The following proposals can be made for future work-

- Implementing the SIMULINK plant model and controller on two laptops, MANET model on the desktop PC using OPNET. Both the plant and controller maintain communication to the MANET model over two wireless links.
- Applying other control methods such as model predictive control on the laptop using SIMULINK etc.

REFERENCES

- AL-HAMMOURI, A., LIBERATORE, V., AL-OMARI, H., AL-QUDAH, Z., BRANICKY, M. S. & AGRAWAL, D. (2007). A co-simulation platform for actuator networks. In: *the 5th international conference on Embedded networked sensor systems*, 383-384.
- ANDERSSON, M., HENRIKSSON, D., CERVIN, A. & ARZEN, K. E. (2005). Simulation of Wireless Networked Control Systems. In: *the 44th IEEE Conference on Decision and Control and European Control Conference (ECC)*, 476-481.
- CERVIN, A., HANRIKSSON, D., LINCOLN, B., EKER, J. & ARZEN, K. E. (2003). How does control timing affect performance? Analysis and Simulation of Timing using Jitterbug and TrueTime. *IEEE Control Systems Magazine*, **23**, 16-30.
- CERVIN, A., HENRIKSSON, D., LINCOLN, B., EKER, J. & ARZEN, K. E. (2002). Jitterbug and TrueTime: Analysis Tools for Real-Time Control Systems. In: *the 2nd Workshop on Real-Time Tools*.
- CHANG, X. (1999). Network simulations with OPNET. In: *the 31st conference on Winter simulation: Simulation – a bridge to the future*, 307-314.
- COLANDAIRAJ, J., IRWIN, G. W. & SCANLON, W. G. (2005). Analysis and Co-Simulation of an IEEE 802.11B Wireless Networked Control System. In: *the 16th IFAC World Congress*.
- COLANDAIRAJ, J., IRWIN, G. W. & SCANLON, W. G. (2006). An integrated approach to wireless feedback control. In: *UKACC International Control Conference*.
- CONTI, M. & GIORDANO, S. (2007a). Multihop Ad Hoc Networking: The Reality. *IEEE Communications Magazine*, **45**, 88-95.
- CONTI, M. & GIORDANO, S. (2007b). Multihop Ad Hoc Networking: The Theory. *IEEE Communications Magazine*, **45**, 76-86.
- DHAM, V. (2003) Link Establishment in Ad Hoc Networks Using Smart Antennas. *Electrical and Computer Engineering*. Virginia Polytechnic Institute and State University.
- FALL, K. & VARADHAN, K. (2006) The Ns Manual (formerly ns Notes and Documentation), The VINT project.
- HASAN, M. S., HARDING, C., YU, H. & GRIFFITHS, A. (2005). Modelling delay and packet drop in Networked Control Systems using Network Simulator NS2. *the International Journal of Automation and Computing (IJAC)*, **2**, 187-194.
- HASAN, M. S., YU, H., GRIFFITHS, A. & YANG, T. C. (2007). Simulation of Distributed Wireless Networked Control Systems over MANET using OPNET. In: *the 2007 IEEE International Conference on Networking, Sensing and Control*, 699-704.
- IKEDA, M. & SILJAK, D. D. (1990). Optimality and robustness of linear quadratic control for nonlinear systems. *Automatica*, **26**, 499-511.
- JOHNSON, D. B., MALTZ, D. A. & BROCH, J. (2001) *Ad Hoc Networking*, Boston, USA, Addison-Wesley.
- KOTZ, D., NEWPORT, C., GRAY, R. S., LIU, J., YUAN, Y. & ELLIOTT, C. (2004). Experimental Evaluation of Wireless Simulation Assumptions. In: *the ACM/IEEE International Symposium on Modelling, Analysis and Simulation of Wireless and Mobile Systems (MSWiM)*, 78-82.
- LIU, J., YUAN, Y., NICOL, D. M., GRAY, R. S., NEWPORT, C. C., KOTZ, D. F. & PERRONE, L. F. (2004). Simulation validation using direct execution of wireless ad-hoc routing protocols. In: *the 18th Workshop on Parallel and Distributed Simulation (PADS'04)*.
- NEWPORT, C. (2004) Simulating mobile ad hoc networks: a quantitative evaluation of common MANET simulation models. Dartmouth College Science.
- PELLEGRINI, F. D., MIORANDI, D., VITTURI, S. & ZANELLA, A. (2006). On the Use of Wireless Networks at Low Level of Factory Automation Systems. *IEEE Transactions on Industrial Informatics*, **2**, 129-143.
- PLOPLYS, N. J. (2003) Wireless feedback control of mechanical systems. *Mechanical Engineering*. Champaign, University of Illinois.
- PLOPLYS, N. J., KAWKA, P. A. & ALLEYNE, A. G. (2004). Closed-loop Control over Wireless Networks. *IEEE Control Systems Magazine*, **24**, 58-71.
- ROYER, E. M. & TOH, C. K. (1999). A Review of Current Routing Protocols for Ad Hoc Mobile Wireless Networks. *IEEE Personal Communications*, **6**, 46-55.
- SILJAK, D. D. (1991) *Decentralized control of complex systems*, Boston, MA, Academic Press.
- WILLIG, A., MATHEUS, K. & WOLISZ, A. (2005). Wireless Technology in Industrial Networks. *Proceedings of the IEEE*, **93**, 1130-1151.