Acta Universitatis Sapientiae

Electrical and Mechanical Engineering

Volume 8, 2016

Sapientia Hungarian University of Transylvania
Scientia Publishing House

Contents

A. Szabó
GNU Radio Based Testbed (GRaTe-BED) for Evaluating the Communication Link of Unmanned Aerial Systems 5

Cs. Nagy, Z. Bíró-Ambrus, L. Márton
Ultrasound-Based Indoor Robot Localization Using Ambient Temperature Compensation 19

Zs. Sándor, G. Kiss
Development of Motion Detection Algorithms Based on Simultaneous Execution Using Mobile Phone Sensors 29

Ö. Darabont, K. J. Kiss, J. Domokos
Evaluation of Different Thin-Client Based Virtual Classroom Architectures 42

A. Kelemen, D. Biró, A. Zs. Fekete, L. Jakab-Farkas, R. R. Madarász
Macroscopic Thin Film Deposition Model for the Two-Reactive-Gas Sputtering Process 62

Acta Universitatis Sapientiae
Electrical and Mechanical Engineering, 8 (2016) 5-18

GNU Radio Based Testbed (GRaTe-BED) for Evaluating the Communication Link of Unmanned Aerial Systems

András SZABÓ

Department of IT, Faculty of Military Sciences and Officer Training,
National University of Public Service, Budapest,
e-mail: szabo.andras@uni-nke.hu

Manuscript received October 15, 2016; revised December 15, 2016

Abstract: UAS (Unmanned Aerial Systems) are commonly used in 3D (dull, dirty and dangerous) missions, because these are not endangering the operators life, while reduce maintenance costs and increase maneuvering capabilities. Despite of these advantages we should consider the possible vulnerabilities of this technology as well. Unmanned vehicles can be controlled via direct communication link, or they can work in a preprogrammed mode. Usually the preprogrammed mode is based on radio navigation systems, so we can draw a conclusion that both depend on the RF environment. In this paper I analyze a possibility to effectively evaluate the communication link of an UAS. Developers have to consider several key factors (type of operation, endurance, payload type and size, propulsion, communication link, etc.) during the development process. They are also responsible that the final product meets the predefined requirements. On the other side commercial UAS owners should have a possibility to compare and evaluate the UAS before the acquisition. Finally, operators and frequency management entities need tools to diagnose the possible sources of interference regarding the unmanned vehicles. To understand the consequences of interference in the RF spectrum we have to be able to measure the quality of the communication link in different usage scenarios. In my research I evaluate the usage of SDRs (Software Defined Radios) in RF Test and Evaluation processes. After analyzing the possibilities for a flexible testbed, I demonstrate the usability with some measurements in the GNU Radio signal processing framework.

Keywords: UAS, SDR, RF, Test & Evaluation, USRP, GNU Radio.

1. Introduction

There are multiple names for Unmanned Aircraft Systems (UAS) like drone, remotely operated aircraft (ROA), remotely piloted aircraft (RPA), remotely piloted vehicle (RPV), and autonomous aerial vehicle. The innovations of the

last few years let this technology become so popular today. For hobbies through commercial and governmental entities all the way up to the military.

The driving force behind this technology was originally the military, similarly to other inventions in the field of telecommunication and computer science. This technology is getting into our everyday life so suddenly and drastically, like the Internet and the mobile communication several years ago. Those caused major changes in the economy, scientific world, culture, education and most importantly transformed our social life. The question is what these unmanned devices will cause in the history of mankind. The global term for these devices is Cyber-Physical Systems, which describe the fact that they exist and act in both dimensions. The technical and legal backgrounds haven't been established yet, as it is still an emerging technology.

From the security perspective the drones are rising threats and unused opportunities in the same time. We have heard notable cases (drone crashes[1] [2], jamming incident[3], counter operations in conflicts[4] [5], football hooliganism[6], drug smuggling[7]) in the news which prove that unaware hobbyist, criminals, malicious users and even state-actors recognized the possibilities of this technology.

2. Background and motivation

The before mentioned threats have to be addressed with proper counter-measures. We have to consider the possible counter UAS techniques, and also the usable defense mechanisms. While putting together the elements of the threat model, we have to consider the basic requirements of the UAS as well as the possible exploitation methods.

The duality of this technology is represented by the fact that we have to encounter jamming [1] or dazzling attack[8] against the UAS sensors [2], [3] or attackers can disturb public event using drones (fly over with a political interest[1, 6] or endangering the public (several drones crashed in crowded public space like stadiums[9]). So we have to be prepared for the offensive and defensive operations

[1] Source: http://www.independent.co.uk/news/world/asia/man-arrested-for-landing-radioactive-drone-on-japanese-prime-ministers-roof-10203517.html
[2] Source: http://www.washingtonpost.com/wp-srv/special/national/drone-crashes/database/
[3] Source: http://www.osce.org/ukraine-smm/140586
[4] Source: https://medium.com/war-is-boring/ukraine-scrambles-for-uavs-but-russian-drones-own-the-skies-74f5007183a2
[5] Source: http://www.abc.net.au/news/2015-04-22/ukraines-diy-drone-war/6401688
[6] Source: http://www.bbc.co.uk/sport/0/football/29624259
[7] Source: http://www.bbc.co.uk/news/technology-30932395
[8] like pilots have blinded by lasers or other light source
[9] Source: http://edition.cnn.com/2015/09/06/us/drones-sports-events/
http://www.droneinjurieslawyer.com/read-me/ http://www.bbc.com/news/technology-26921504

too. It is quite contradictory, that micro UAS are publicly available for a low price, but security of the used communication standards, and the possible counter-UAS techniques are not well known. Law enforcement and security agencies need counter-UAS techniques [4], which are still available on the market (solutions for detecting[10, 11, 12, 13, 14] and for detecting and countering[15, 16]. These governmental entities are in a difficult position because these countermeasures usually haven't been inspected by independent test facilities and the results aren't available publicly (which is understandable regarding the sensitive nature of these countermeasures). Both the effectiveness and the limits have to be analyzed to ensure proper counter-UAS capability and to minimize unneeded interference with legitimate spectrum users (to decrease the footprint of the equipment only to that location which has to be secured). Test procedures have to be defined and made publicly available to standardize the requirements. It is also a challenge to navigate in the field of UAS technology, where there are multiple companies on the drone market with wide portfolio, several frequency ranges (usually 72 MHz, 433 MHz, 915 MHz, 2,4 GHz for control 900 MHz, 1,2 GHz, 2,4 GHz or 5,8 GHz for payload communication), and different protocols (Wifi, 3DR, MAVlink, etc.) for the control channel. In [5] researchers highlighted the lack of standardized protocols for civil UAS control communication. So we have to evaluate devices with different kind of RF parameters, and various proprietary, or open source upper layer protocols.

3. Objectives and Scope of the Research

In this paper I will investigate the possibilities to evaluate the communication systems of Unmanned Aerial Systems (UAS). As highlighted by several researches, the communication link [6], [7] is vital for the future UAS development and deployments of unmanned systems. Engineers meet a challenge when trying to analyze the different open source (like MAVlink[17, 18])

[10] SHARPEYETM SxV RADAR TECHNOLOGY https://www.kelvinhughes.com/security/uav-drone-detection
[11] ARRIER DSR-200 Drone Surveillance Radar http://www.detect-inc.com/DeTect%20-%20Security/TDS%20-%20HARRIER%20DSR%20200d%20150406US.pdf
[12] Army Tests New Acoustic Threat Detection System http://defensetech.org/2015/05/20/army-tests-new-acoustic-threat-detection-system/
[13] DroneTracker http://www.dedrone.com/en/dronetracker/drone-detection-hardware
[14] Domestic Drone Countermeasures http://www.ddcountermeasures.com/
[15] Anti-UAV Defence System (AUDS) http://www.securitynewsdesk.com/?post_type=post&p=50833
[16] Falcon Shield http://www.janes.com/article/54319/dsei-2015-selex-es-unveils-falcon-shield-counter-uav-system
[17] MAVLink micro air vehicle marshalling / communication library https://github.com/mavlink/mavlink
[18] MAVLINK Common Message Set https://pixhawk.ethz.ch/mavlink/

and closed-source UAS communication protocols. Nowadays the possibility offered by SDR is adequate choice for RF test and evaluation (T&E) facilities. SDR can speed up the measurement and validation process, and also migrate the bulk of the physical measurements to simulation, emulation (only essential field tests are done in the real world) [8]. SDRs already facilitate the integrating simulations to the real-word RF measurements. Subsystems or essential components can be simulated in frameworks like the open-source GNU Radio (or commercial tools like Labview and Matlab) and access the RF world with tools such as ETTUS research's USRP SDR family.

New opportunities, like measurement devices organized and controlled in a distributed network will increase the reliability of the measurement results. Device-to-device communication enabled us to automatize, synchronize and fusion different measurements (like measuring wind, temperature, humidity and other weather condition while sensing the RF spectrum and validating the position of the Device Under Test - DUT).

In my research I highlight these possibilities and proof them with a GNU radio based concept. My research was aiming to achieve the following goals: Design an integrated testbed for RF interference and jamming measurements, Implement a network centric measurement capability, Create a flexible, scenario drive testbed (*Fig. 1*), Create the SWOT analysis (Strengths and Weaknesses, Opportunities and Threats) of this approach.

4. Previous Work

UAS technology is a hot research topic nowadays, there are several publications related to my field of interest. In [9], [10] authors highlighted the importance of robust communication between the UAV and the ground control station, which means that we have to test these UA in a possible not-cooperative/hostile EM environment (to be prepared to the unintentionally hazardous or hostile situations) and also develop suitable response to aerial terrorist attacks (especially which conducted with commercial UAS [10]. In spite of the high proliferation of mini UAV systems, there are only a few recommendations about testing the DUT as a hardware in the loop [11], evaluate its communication [7], [12] or analyze the performance of the sensor/payload systems [13]. Engineers are focusing on the vulnerability of the communication link against unintentional interference [14] or adversaries with RF jamming capabilities [15], the cyber exploitation of these devices are also a hot topic [16]. However, very few of the before-mentioned papers try to understand the threat against UAS, or attempt to manage this problem in holistic point of view. In fact the security of these cyber-physical devices has to be

analyzed in a comprehensive approach [17], from the physical level[19] to the application layer. In the military this classification is mentioned as CEMA (Cyber Electromagnetic Activities) [18]. There are several researches about the usage of SDR as communication link testbed [19]. They are focusing on the communication protocol under development and using SDRs only as a point-to-point link test equipment, not taking the advantage of the previously mentioned opportunities.

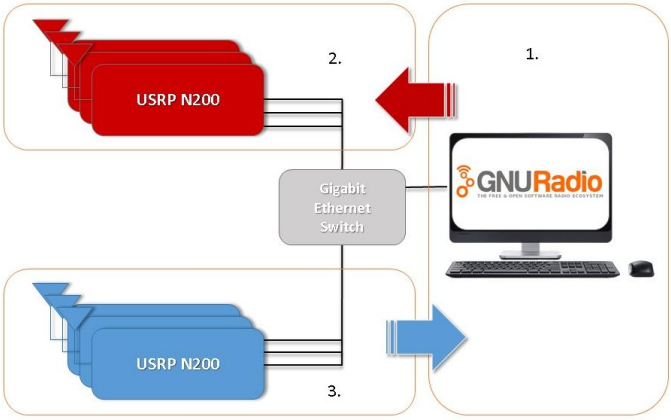

Figure 1: Networked measurement instrument (1. is the network segment of the management node(s), 2. is the team-jammer network and 3. is the RF sensor network)

Different jamming scenarios against advanced wireless standards are analyzed, and organized in a methodical manner in [15]. Researchers in [20] collected the possible threats against UAS, but they focused on the computer based simulation.

These researches encouraged me to create the GNU Radio Test BED (GRaTe-BED) for testing UAS communication link, with the capability to analyze unintentional interference or malicious jamming activity, anomalies from Layer 1 to Layer 7.

[19] Layer 1 35.100: Open systems interconnection (OSI) Source: http://www.iso.org/iso/iso_catalogue/catalogue_ics/catalogue_ics_browse.htm?ICS1=35&ICS2=100

5. Methodology

Design considerations

In the following I will present the main features which I had to consider during my research.

SDR platform

SDR solves the flexibility needs by converting most of the PHY layer signal processing blocks into a software layer. This method not only simplifies design and implementation of advanced radio system, but it also made new approaches possible. SDR receivers, transceivers can be deployed in different locations, multiple radios can be organized into a network (see *Fig. 1*), to mention a few novel features. This technology simplifies RF testing, by transforming measurement into scriptable steps. Multiple scenarios can be tested and validated, minimizing the human error (measurement errors). The repeatable steps can be automatized but the design of the scenario and the analysis of the final results should done by engineers. It's important to mention that the testbed can't operate without human interaction.

For the hardware part of the testbed I used USRP N200 with UHD_003.005.005 driver, usrp_n200 firmware and usrp_n200_r4 FPGA image (openly available from ettus.com), the software component was GNU radio (Version 3.7.2.1). There are several simulation platforms which would have been utilized for this project (for example Labview, Matlab and Simulink). My choice was Gnu Radio. The reasons behind this decision were my previous experience with this tool, and also to make the results publicly, and freely available for other researchers. The hardware parameters are well documented (for us it's important that maximum output power is between 17-20 dBm, usable between 400 to 4400 MHz with SBXv3, and 68.75 to 2200 MHz with WBXv3 frontend boards [21]).

USRP hardware can work both with command line tools like (UHD_FFT, UHD_SIGGEN) or graphical interface of the GNU Radio Companion. Both have advantages and some drawbacks as well.

To automate different scenarios cli tools are useful, because they can be organized and executed in a script, without user interaction (besides that usually users triggers the start and check the integrity of the output).

On the other hand if the analysis needs human interaction, the results are not predictable, then it is recommended to conduct the evaluation under supervision of a human operator.

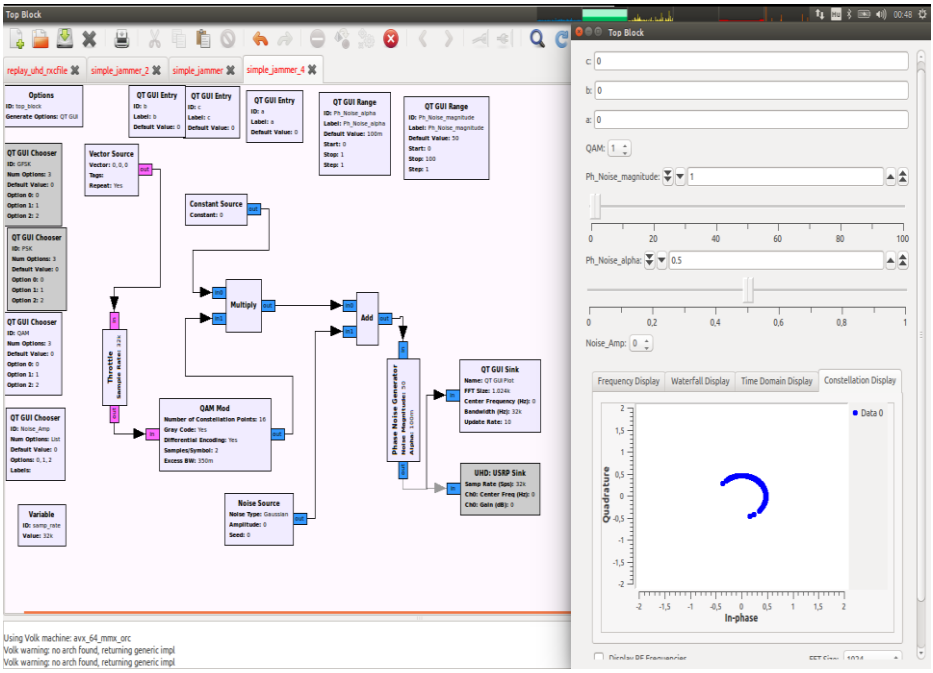

Figure 2: Flowgraph of the jammer in GNU Radio (left), with the constellation diagram of the output (right)

To organize the graphical interface we can choose from a notebook view (different graphical elements are on different "pages") or we can create only one "common View", and we can orient different elements with a Grid Position parameter. I find it much handier to organize into a notebook view then to a complex diagram.

At the jammer node similarly pre-programed steps like scripts using uhd_siggen (*Fig. 3*) or interactive user control with uhd_siggen_gui can be utilized (as a signal generator it can create basic signals sine, sweep, square, noise). If we are using GNU Radio Companion then we can use virtual measurement devices (spectrum analyzer, waterfall display, oscilloscope, histogram view, constellation diagram, etc.) to monitor the state of the flow graph. On the other hand the output should be inspected with external device (to check that the SDR generating the predefined signals with the allowed power).

Figure 3: DJI Phantom 2 vision control channel (1) with a pulsed jammer signal (2) generated with uhd_siggen script

Threat library

The effectiveness of the testbed is highly dependent on the interference/jam library, and the topology represented by the scenario. The utilized error detection/correction algorithms, media access schema, the channel coding made sophisticated RF standards cause that the effectiveness of the jammer signal is difficult to represent with mathematical models. Likewise in [22] the author mentioned the difficulties of simulating complex Electronic Warfare systems to measure the effectiveness. In this case SDRs can solve the problem by interconnecting the simulation and the real world RF measurements.

Unknown RF standards can be observed in a black box approach, with replaying signals we can spoof pre-recorded control signals and analyze the system responses [23]. For the "replay attack" (*Fig. 4*) we can save multiple different samples, and replay them in different orders, but it can increase the complexity of the software if we do not merge these samples.

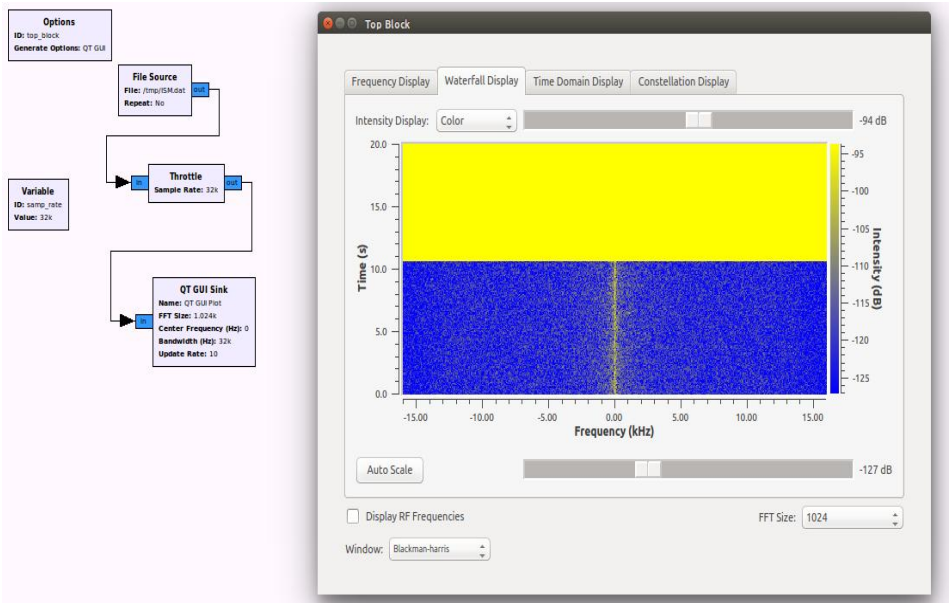

Figure 4: Replay attack demonstrated in GNU Radio Companion

Channel modeling

Author in [24] compared the possibilities for channel simulation in GNU Radio. With these models we can introduce different channel degradation effects (multipath, frequency selective fading, etc.) to observe the DUT response to the possible RF anomalies (not only the Transmitter but also the jammer performance can be degraded by environmental conditions).

Networking

While creating a high density SDR network array we have to consider the difficulties caused by multiple measurement device sets to high sampling rate (like 100 Msamp/sec with 20 MHz RF bandwidth) [25].

USRP N200 has Gigabit Ethernet network interface, supporting IPv4, and only the IP address can be changed (not even the netmask). So theoretically the maximum number of sensors in the network is 254 (because USRP doesn't support routing, so we can't create a routed topology).

Sampling rate is a critical parameter to the network load (*Fig. 5*), and the proper configuration of the TCP/IP stack is also vital [26].

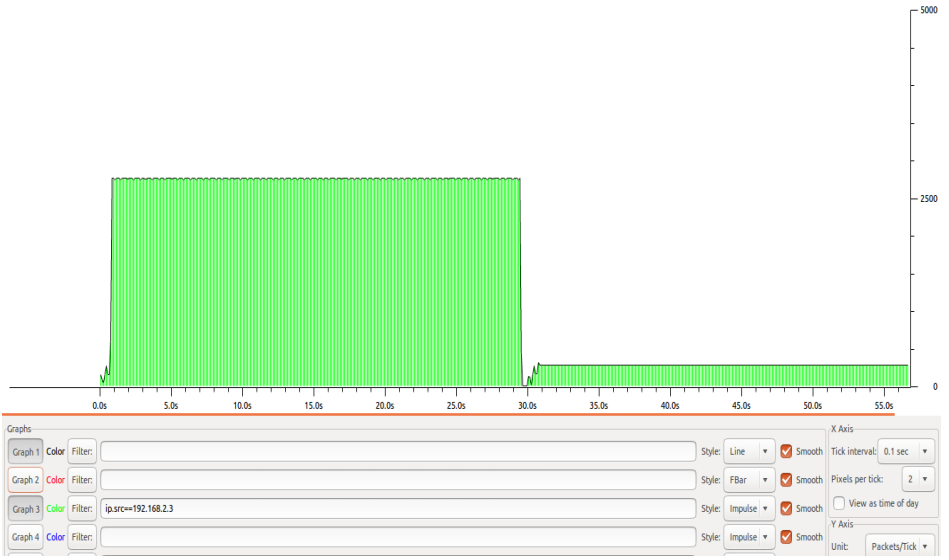

Figure 5: Network load in Wireshark IO Graph (the network traffic of uhd_fft was monitored)

The script used for measurement:

/usr/sbin/tcpdump -i eth1 -s 65535 -w /tmp/USRP_10M+1M_uhd_FFT.pcap & timeout -sHUP 30 /usr/bin/uhd_fft --args="addr=192.168.10.2" --fft-size=2048 --fft-rate=40 -f 2G -s 10000000;timeout -sHUP 30 /usr/bin/uhd_fft --args="addr=192.168.10.2" --fft-size=2048 --fft-rate=40 -f 2G -s 1000000;pkill -HUP -f /usr/sbin/tcpdump

The first 30 sec. uhd_fft was configured to sample 10 Msps, than 30 sec. with only 1 Msps.

RF Navigation

Global (like GPS, GNSS, etc.) or autonomous [9] radio navigation systems are commonly used in UAS for the autopilot system, and also have important role in remotely operated device in “link lost” situations. Likewise, these technologies can be tested on SDR platform [27]

Test scenario building

It’s a mandatory requirement for T&E equipment to perform in a repeatable, flexible way and to be able to adapt to new technologies (like new protocols). In GNU radio basic signal processing blocks are available (modulators, demodulators, filters, etc.), if a new / special purposed processing function is

required, it is straightforward to integrate into GNU Radio (as block is develop in C++).

Evaluation of the results

We have to define metrics before performing the test, for example a 4 state metric look like this: I. "*No effect*" II. "*No operator control over the vehicle*" III. "*Position change caused by jam signal*" "*Full control achieved by the attack*". The RF measurement results and these metrics should be logged to the final report.

Fuzzing

If we are analyzing a standardized protocol with a reason to create a protocol aware jamming scenario, then we need information about the communication protocol. CGRAN (Comprehensive GNU Radio Archive Network)[20] hosts a lot of "out-of the tree modules"[21] (like GSM, LTE, Bluetooth, IEEE 802.11 protocols). These modules can be utilized in a high protocol level fuzzing. To check the availability of the DUT, we have to create an entity which analyzes its responses. In IT security this called as *oracle* [28]. If the DUT can cooperate (it can measure and log the signal quality), then we only have to synchronize the jamming scenario with this logging mechanism, and after the tests we get information about the jamming efficiency. It is difficult to analyze the onboard navigation, guidance and control loop, if the DUT is in flight. Most commercial UAS have proprietary control channel and debugging procedures (there are reverse engineering attempts like [29])

If the DUT isn't capable to log, than we should create some external sensor to evaluate the jammer performance in an indirect mode. The testbed for cyber-physical systems can utilize multiple sensors to analyze the DUT response to environmental changes. Visual recording of the responses is the easiest, but difficult to organize and analyze after the measurements. Atmospheric conditions can be useful, with low cost devices like Raspberry Pi or Arduino we can create a sensor network, and logs can be collected to the management node which schedule the RF tests.

SWOT

The Strengths and Weaknesses of the testbed, the identified Opportunities open to us and the Threats we have to face are shown in *Fig 6*.

[20] http://cgran.org/
[21] more details can be found at https://gnuradio.org/redmine/projects/gnuradio/wiki/OutOfTreeModules

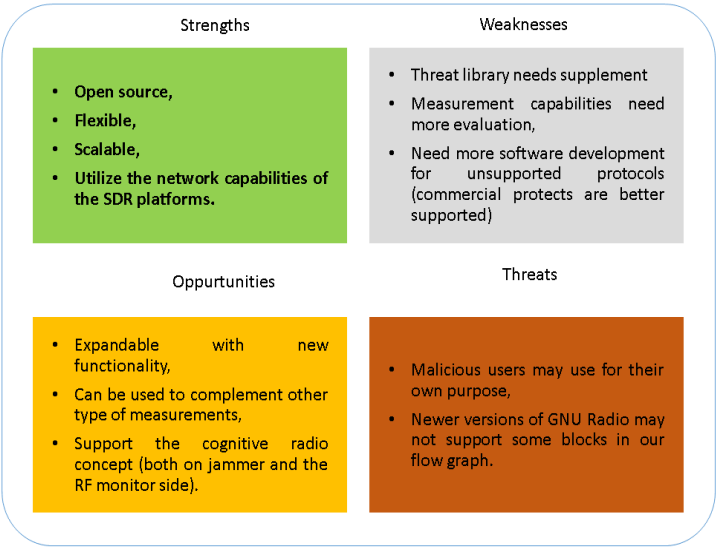

Figure 6: SWOT analysis

6. Conclusion and recommendations

Several researchers analyzed possibilities to exploit the control subsystem of UASs. However, these analyses are mostly focusing on the cyber part of the security (in a network oriented view only the upper OSI layers are covered) or on just one type of communication link/protocol. To understand and evaluate the threat against this technology we have to manage this problem in a comprehensive manner. Security of the unmanned technology should not be based on "security through obscurity". Communication protocols have to be publicly available to enable adequate security analysis.

I demonstrated a possible approach with a flexible, open source, easily implementable framework. I already highlighted the importance of the comprehensive threat modeling. Full disclosure [30] of UAS vulnerabilities was out of my scope. It would be unethical, because changing the hardware in embedded systems is in most of the cases impossible, modifying the software is difficult, and in most of the cases it is impossible for the end-user. I recommend to all my colleagues in this field of research to follow this procedure. In my paper I highlighted possible testing techniques and toolsets helping the reliable evaluation of this technology.

Future plans

- Creating a database back-end to manage the measurement and to store the results;
- Testing with different UAS;
- Expanding the RF spectrum (the current hardware can analyze up to 4,4 GHz, with down converters it is possible to analyze upper portion of the spectrum);
- Analyzing RF immunity and interference in UAS swarming operations.

References

[1] Kerns, A. J., "Unmanned aircraft capture and control via gps spoofing", *http://radionavlab.ae.utexas.edu/images/stories/files/papers/unmannedCapture.pdf.*

[2] United States Air Force Scientific Advisory Board, "Report on operating next-generation remotely piloted aircraft for irregular warfare", *Section 2.4.2 http://info.publicintelligence.net/USAF-RemoteIrregularWarfare.pdf*, 2011.

[3] Federal Aviation Administration, "Laser hazards in navigable airspace" *https://www.faa.gov/pilots/safety/pilotsafetybrochures/media/laser_hazards_web.pdf* .

[4] Makkay, I., "Fight of drones" ("Drónok harca"), 2015 *http://www.repulestudomany.hu/ folyoirat/2015_1/2015-1-05-0192-Makkay_Imre.pdf.*

[5] Bob Kerczewski, Jeff Wilson, Bill Bishop – "Frequency Spectrum for Integration of Unmanned Aircraft", *NASA http://ieeexplore.ieee.org/stamp/stamp.jsp?arnumber=6719706.*

[6] Yanmaz, E.; Kuschnig, R. ; Bettstetter, C.," Channel Measurements over 802.11a-based UAV-to Ground Links" *http://ieeexplore.ieee.org/xpl/freeabs_all.jsp?arnumber= 6162389 &abstractAccess =no&userType=inst.*

[7] Steve Gardner, "A communication link reliability study for small unmanned aerial vehicles", *Enerdyne Technologies, Inc.- COMM OPS Trends in Communication Systems For ISR UAVs*, January 2009, *http://www.milsatmagazine.com/story.php?number=893938022.*

[8] Pywell, M., Midgley-Davies, M. "EW test and evaluation - assuring survivability and operational effectiveness", *Electromagnetic Engineering Department BAE SYSTEMS, Military Air & Information http://tangentlink.com/wp-content/uploads/2014/07/2.-Electronic- Warfare-Test-Evaluation-Mitch-Midgley-Davies.pdf* .

[9] Miko, G. , Nemeth, A., "Combined communication and radio navigation system for small UAVs", *Radioelektronika 2013 23rd International Conference*, pp. 284-288, ISBN: 978-1-4673-5516-2.

[10] Turan, M., Gunay, F., Aslan, A. , "An analytical approach to the concept of counter-UA ops (CUAOPS)", *Journal of Intelligent & Robotic Systems*, January 2012, Volume 65, Issue 1-4, pp 73-91, ISSN 1573-0409, *http://link.springer.com/article/10.1007%2Fs10846-011-9580-6.*

[11] Zheng, Y., Wang, Y., "Hardware in the Loop Simulation for Low-altitude UAV Link in the Complex Terrain", *Applied Mechanics and Materials* Vols. 336-338 (2013), pp. 1907-1912, ISBN:9783037857519.

[12] Yanmaz, E.; Bettstetter, C., " Channel measurements over 802.11a-based UAV-to-ground links", *GLOBECOM Workshops (GC Wkshps) IEEE*, 2011, ISBN:978-1-4673-0039-1.

[13] Wan, J., Suo, H., Yan, H., Liu, J., "A general test platform for cyber-physical systems:unmanned vehicle with wireless sensor network navigation", *2011 International Conference on Advances in Engineering, http://ac.els-cdn.com/S1877705811054658/1-s2.0-S1877705811054658-main.pdf?_tid=3c747952-6b8b-11e5-bae9-00000aab0f6b&acdnat=1444068326_18bdf0fe9b500d0fcc69e141753afc30.*

[14] Koepke, G., Young, W., Ladbury, J., Coder, J., "Complexities of testing interference and coexistence of wireless systems in critical infrastructure", 2015, *http://nvlpubs.nist.gov/nistpubs/TechnicalNotes/NIST.TN.1885.pdf* .

[15] Slater, D., Tague, P., Poovendran, R., Li, M., "A game-theoretic framework for jamming attacks and mitigation in commercial aircraft wireless networks" *http://www.ee.washington.edu/research/nsl/papers/aiaaInfotech09c.pdf*.

[16] Jamshidi, M., ‚Jaimes Betancourt, A. S., Gomez, J., "Cyber-physical control of unmanned aerial vehicles", *http://ac.els-cdn.com/S1026309811000691/1-s2.0-S1026309811000691-main.pdf?_tid=71385718-66c6-11e5-9137-00000aab0f6b&acdnat=1443543999_0cc41af19de0b2765c48bdb1e55402c3*.

[17] Saeed, A., Neishaboori, A., Mohamed, A., Harras, K. A., "Up and away: a cheap UAV *cyber-physical testbed" http://arxiv.org/pdf/1405.1823v1.pdf*.

[18] Department of the Army, "Cyber electromagnetic activities, field manual no. 3-38", Washington, DC, 12 February 2014 *http://armypubs.army.mil/doctrine/DR_pubs/dr_a/pdf/fm3_38.pdf*.

[19] Zainudin, A., Sudarsono, A., Astaw, I. G. P., "Reliability analysis of digital communication for various data types transmission using GNU Radio and USRP" *http://www.researchgate.net/publication/259477837_Reliability_Analysis_of_Digital_Communication_for_Various_Data_Types_Transmission_Using_GNU_Radio_and_USRP*.

[20] Javaid, A., Sun, W., and Alam, M., "A Cost-Effective Simulation Testbed for Unmanned Aerial Vehicle Network Cyber Attack Analysis", *Safe & Secure Systems & Software Symposium (S5)* June 9-11, 2015, *http://www.mys5.org/Proceedings/2015/Day_3/2015-S5-Day3_0805_Sun.pdf*.

[21] Ettus Research Application Note, "Selecting an RF Daughterboard", *http://www.ettus.com/content/files/kb/Selecting_an_RF_Daughterboard.pdf*.

[22] Tucker, T.W. "Jammer testing and chaos", *Tactical Technologies Inc.http://tti-ecm.com/uploads/resources_technical/jammer%20testing%20and%20chaos.pdf*.

[23] Chen, J., Zhang, S.†, Wang, H., Zhang, X., "Practicing a record-and-replay system on USRP", *Sigcomm Conference 2013, http://conferences.sigcomm.org/sigcomm/2013/papers/srif/p61.pdf* .

[24] O'Shea, T. "GNU Radio channel simulation: trolling sub-par modem algorithms and implementations for fun and profit", *Research Faculty, Virginia Polytechnic Institute and University, Arlington, VA*, 1 Oct 2013 *http://static1.1.sqspcdn.com/static/f/679473/23654472/1381240802597/grcon13_oshea_chansim.pdf?token=iRbiWsmfTNpfPqITN708izU3fQU%3D*.

[25] Department of Electrical and Computer Engineering, Ben-Gurion University of the Negev, "Introduction to USRP", *http://www.researchgate.net/file.PostFileLoader.html?id=545b5550d039b12d7c8b4567&key=80f8366d-3355-40ad-a657-7ffb6b904ff8&assetKey=AS:272121316478982@1441890186007*.

[26] Ettus Research, "Latency", *https://github.com/EttusResearch/uhd/wiki/Latency*.

[27] Brown, A., Tredway, R., and Taylor, R., "GPS signal simulation using open source GPS receiver platform" *https://wireless.vt.edu/symposium/2011/posters/GPS%20Signal%20Simulation_Brown.pdf*.

[28] Knudsen, J., Varpiola, M., "What is fuzzing: the poet, the courier, and the oracle", 2015, *http://www.codenomicon.com/resources/white-paper/pdf/WhatisFuzzing.pdf* .

[29] "Hijacking DJI Phantom 2 Vision and P2V+", *https://github.com/noahwilliamsson/dji-phantom-vision*.

[30] Cencini, A., Yu, K., Chan, T., "Software vulnerabilities: full-, responsible-, and non-disclosure", December 7, 2005, *http://courses.cs.washington.edu/courses/csep590/05au/whitepaper_turnin/software_vulnerabilities_by_cencini_yu_chan.pdf p. 10*.

Acta Universitatis Sapientiae
Electrical and Mechanical Engineering, 8 (2016) 19-28

Ultrasound-Based Indoor Robot Localization Using Ambient Temperature Compensation

Csaba NAGY, Zalán BIRÓ-AMBRUS, Lőrinc MÁRTON

Department of Electrical Engineering, Faculty of Technical and Human Sciences,
Sapientia Hungarian University of Transylvania, Tg. Mureş, Romania
e-mail: csaba3nagy@yahoo.com, birozali@yahoo.com, martonl@ms.sapientia.ro

Manuscript received January 24, 2016; revised September 15, 2016

Abstract: This paper focuses on the hardware development of an indoor ultrasound based robot localization system. The problems related to the ultrasound based distance measurements are presented and solutions are proposed related to Time of Flight measurements and measurement synchronization. An optimization based compensation method is introduced to attenuate the effect of the ambient temperature on the distance measurement precision. Experimental measurements were performed to analyze the applicability of the developed system and measurement methods.

Keywords: robot tracking, ultrasound, mobile robot, localization, distance measurement.

1. Introduction

Robot positioning and tracking is an important issue when developing complex applications in the fields of robotics. The position measurement part of the mobile robot control system should provide accurate information and high measurement rate when tracking control algorithms are executed with high speeds. Generally, the robots are equipped with dead reckoning sensors such as inertial measurement units and encoders. Those are supplying the robot's position with high update rate but the measurement cumulates error over time, it becomes inaccurate. The solution is to use a secondary position measurement source with lower update rate and reduce the cumulated error. For outdoor applications GPS is suitable to provide secondary position data for mobile robots and flying drones. However, the GPS can hardly be used for indoor localization.

For indoor localization multiple solutions can be applied. The work floor can be covered with RFID (Radio Frequency Identification) tags and a tag reader mounted to the robot. This method provides a precision in the order of centimeters [1]. Other solutions try to locate the robot based on Bluetooth [2] or on Wi-Fi [3] signal strength. This method offers a precision of tens of centimeters and the update rate is not suitable for robot control.

Other common method is the ultrasound based time of flight measurement. In the work [4] a setup consisting of 1 transmitter and 2 receivers is used to position a mobile robot. The transmitter uses radio frequency signals to send the reference for the measurement. The measuring error is corrected using a Linear Kalman Filter. Using this technique a maximum error of 2cm can be achieved in a 3.5m × 2.2m working environment. The paper [5] presents a method to combine a RF and ultrasound based wireless positioning with the measurements of dead-reckoning sensors using Unscented Kalman Filter.

Our goal is to design and implement an ultrasound time of flight based localization system with multiple robot control capability. In our system the ultrasonic transmitters are located on the robots, and they are synchronized to the receivers with infrared signals, and ceiling mounted ultrasonic receivers interconnected with RS485 bus to the control computer. The measured distances can be processed using 2D or 3D localization algorithms and transmitted to the robots where the position values are used. This study is an extended version of the conference paper [10].

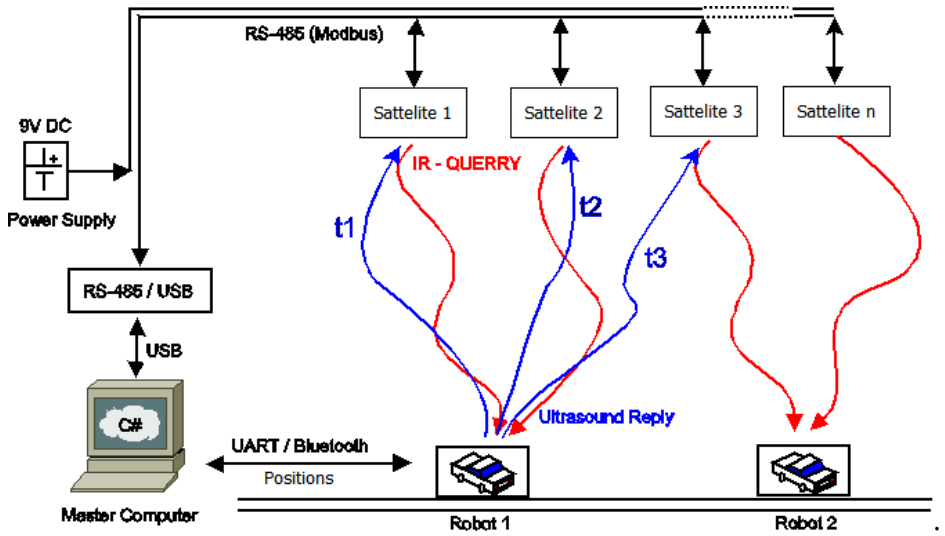

Figure 1: The block diagram of the proposed system

2. Ultrasound-based positioning systems

2.1 General system description

The block diagram of the localization system is shown in *Fig. 1*. The measurement is initiated by the master computer, which configures the next robot ID and broadcasts a synchronization command for all the anchors through the RS485 bus. The controllers of the anchors receive this command through their interrupt inputs and transmit an infrared (IR) packet containing the robot ID for the ultrasonic receiver node placed on the robot. If the node recognizes

his own ID, it responds with a phase-modulated ultrasonic (US) chirp signal [6]. Simultaneously the controllers of the anchors start to sample the ultrasonic receiver. The phase change detection a offers superior precision of the arrival time instant. The ultrasonic signal arrives at the anchor's receiver with Time-of-Flight plus the constant delay between the received IR signal and the transmitted ultrasonic response added by the beacon. After the measurement is finished the Master computer queries the measured times from each node.

2.2. Time of flight measurement

We implemented a distance measurement method based on amplitude and phase modulation similarly as in [6], see *Fig. 2.* The method reduces the error induced by the pendulum effect at the transmitter and the receiver, by introducing a phase modulation in the middle of the measurement signal. The theoretical precision achievable by this method is 0.1% of the wavelength.

In the case of the classic Time of Flight (ToF) measurement the time is calculated until the first instance when the received signal reaches a certain threshold. Because of high inertia of the ultrasound sensor, the resonating time wearies with the distance, and the measurement error can be 2-3 wavelengths long which correspond to a couple of centimeters.

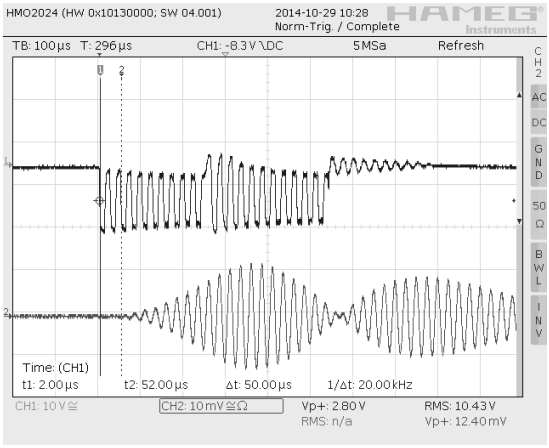

Figure 2: The transmitted and received waveform

Ultrasonic trans-receiver is constructed with a low jitter Infrared receiver [7], a wide angle ultrasonic transmitter [8] along with an amplifier and a dsPIC microcontroller. The task of the trans-receiver is to monitor for the infrared packets, synchronize for the packet start and, if the packet matches the ID, a modulated ultrasonic pulse is transmitted.

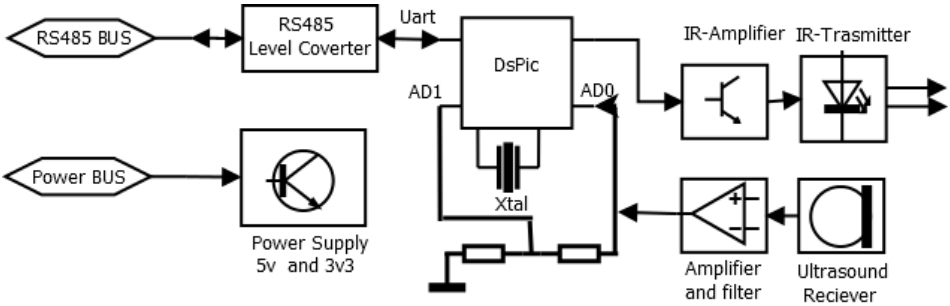

Figure 3: Block diagram of the anchor

The anchor contains a power supply, a high power infrared LED-s, an ultrasonic receiver, combined with a 40 kHz band pass filter (see *Fig. 3*). The received signal is sampled simultaneously by two AD inputs of the microcontroller at 1.2Ms/s speed. One of the inputs expects signals with lower amplitude the other expects signals with higher amplitude. When the data is analyzed, if the higher amplitude inputs are clipping, then the lower amplitude input is used. The reference for the AD converter is 3.3V and, because the amplifier is powered from 5V, the maximum input voltage can be up to 5V.This solution increases the precision when the received signal has small amplitude, but keeps working properly at the minimal distance even if the amplified signal is over 5V. Because the signal is not affected by separate amplifying stages, the phases of the two signals are exactly the same.

2.3. Robot position computation

In order to calculate the robot's position simple calculations are performed as shown in *Fig. 4*.Denote with *d1*, *d2*, *d3* and *d4* are the distances measured by the anchors, with each anchor placed in the same *h* height and with a distance *D* between the two anchors employed.

di is computed as the product of the ultrasound propagation speed and the measured time of flight.

The (X_p, Y_p) coordinates of the robot can be calculated from equations given in (5) and (6):

$$\begin{cases} Xp^2 + Yp^2 = s1^2 \\ Xp^2 + (D - Yp)^2 = s2^2 \end{cases} \tag{5}$$

$$s1 = \sqrt{{d_1}^2 - h^2},\ s2 = \sqrt{{d_2}^2 - h^2} \tag{6}$$

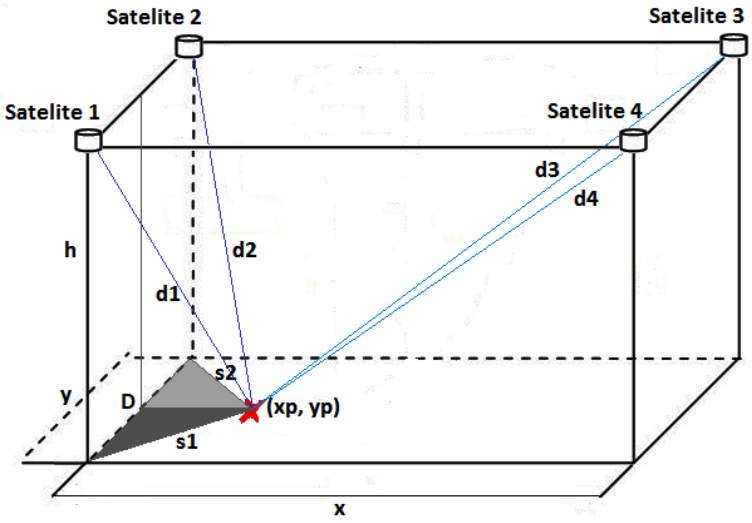

Figure 4: The robot positioning based on the measured distances

2.3. Online temperature approximation and compensation

When the time of flight measurement is available, the distance between the anchor and the robot can be computed using the speed of the ultrasound. The speed of sound varies in function of the environment temperature. The effect of the temperature change is the following formula [5]:

$$\boldsymbol{c = 331.3 + 0.6 \cdot t_{ambient}} \tag{1}$$

where c is the propagation speed of the ultrasound signal and $t_{ambient}$ is the temperature of the sensor's environment. Based on the equation (1) we can conclude that in the case of high ambient temperature the sound travels at higher speeds and the measured distance will be smaller than the real distance, i.e. the calculated position will also drift from the real. Using more measurement pairs, from different anchors and minimizing the distance between the measurement results, we can compute the environment temperature, using which the temperature can be calibrated. The criteria function which is used during the calibration is defined as:

$$J^T = \sum_{i=1}^{n-1} \sum_{j=i+1}^{n} dist\left(\begin{pmatrix} X_i^T \\ Y_i^T \end{pmatrix}, \begin{pmatrix} X_j^T \\ Y_j^T \end{pmatrix}\right)^2 \tag{2}$$

where n is the number of anchor pairs used for distance measurements and $\begin{pmatrix} X_i^T \\ Y_i^T \end{pmatrix}$ denotes the calculated position for the i-th pair at temperature T.

During the distance measurements we modify the t_{act} estimated temperature using the following approach: let $t_1 = t_{act}+0.1$ and $t_2=t_{act}-0.1$.

$$if\ (J_{t1} < J_{tact})then t_{act} = t_1 \tag{3}$$

$$if\ (J_{t2} < J_{tact})then t_{act} = t_2 \tag{4}$$

With the temperature computed as above, the real ultrasound speed can be computed on-line.

Remark: The proposed method works when multiple measurements are available because the method uses the difference between the multiple results.

3. Experimental measurements

The installed system uses 4 anchors and one robot. The anchors are installed at height of 2.9m and the work area is 3 m ×3m. The trans-receiver is mounted on the KUKAyouBot at the height 0.8m. The robot's position is calculated using 2 distances and the two-dimension localization described above hence the robot moves in a horizontal plane. The position is calculated for each pair of anchors and the result is averaged. The achieved precision is under 1cm when all the anchors are visible and the update rate is 8 Hz. The anchors were calibrated using a laser distance measurement tool having 2mm precision.

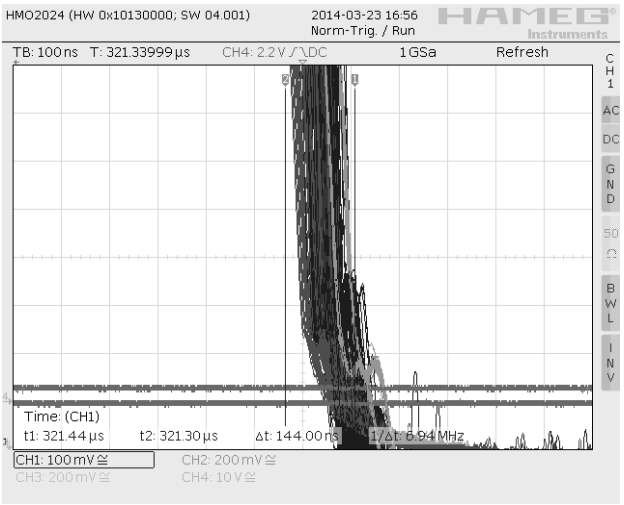

Figure 5: The synchronization jitter over the R485 bus

The synchronization jitter was measured using a digital oscilloscope in persistence mode and multiple measuring cycles were performed. Four ultrasound receivers are connected to the four inputs of the oscilloscope, see *Fig. 5*. The worst case synchronization error due to the interrupt handling is under 144 ns.

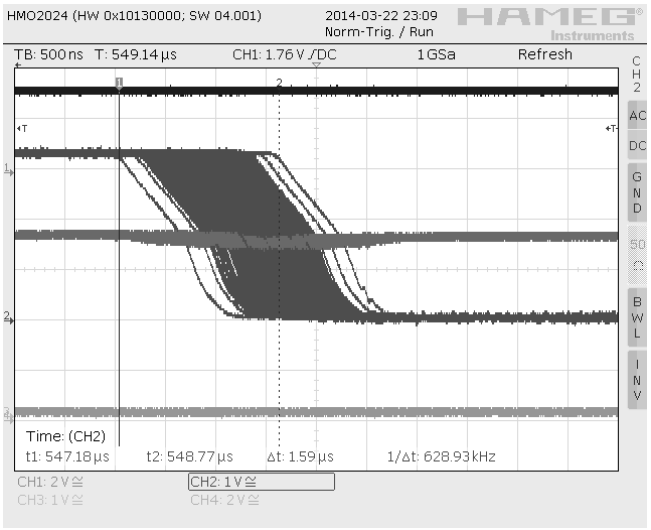

Figure 6: The synchronization jitter on the infrared link

The synchronization jitter on the infrared link is dependent on many variables such as background illumination, distance from the transmitter, but in our case the synchronization jitter, using the same method is under 1.6 µs (see *Fig. 6*), approximately 0.6mm in distance. If the ambient light is too powerful or

the received signal is too small, the jitter can jump up to 40µs equal to4.6mm in the distance measurement. To maximize the received light power, all satellites transmit simultaneously the infrared code.

The cost function based temperature estimation method is shown in *Fig.7.* As it can be seen, the figure algorithm estimates well the ambient temperature in the vicinity of the ultrasound sensor, using only distance measurements. As a results the distance measurement offset can be considerably reduced.

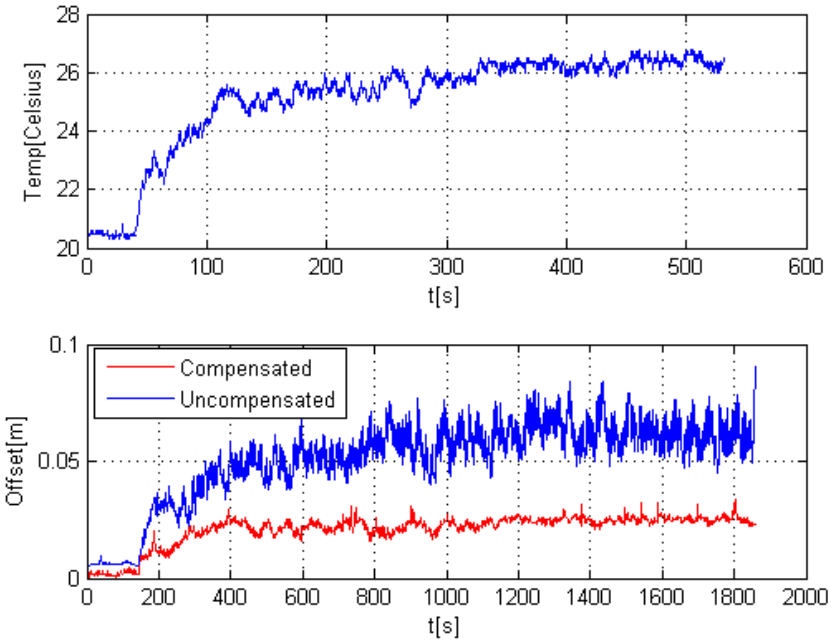

Figure 7: Comparison of the localization error between the compensated and the uncompensated position error while the temperature is changing

Fig. 8 shows the localization of a KUKA youBot mobile platform, on which the ultrasonic receiver is mounted. The robot's reference path and the measured real path are presented in this figure.

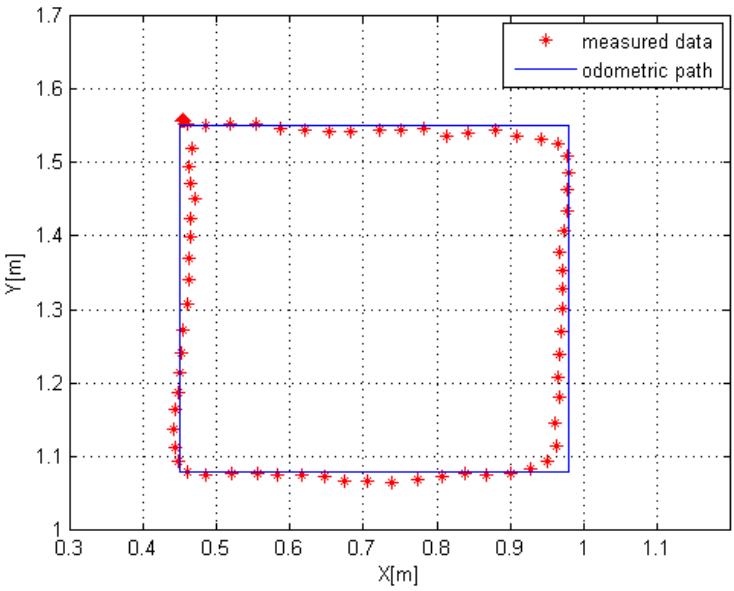

Figure 8: The path of the robot using a rectangular reference

7. Conclusions

We developed an ultrasound-based localization system capable of tracking multiple mobile robots in indoor environment. The proposed solution applies ultrasonic signal Time of Flight distance measurement. To obtain precise measurements in the presence of varying ambient temperature a Monte Carlo method based temperature estimation technique was proposed. From distance measurements the robot's position was computed using triangulation formulas. Based on the measurements we approximated the environment temperature. The proposed solution was successfully applied for robot position calculation in the vertical plane.

Acknowledgements

The research and publication has been supported by the European Union and Hungary and co-financed by the European Social Fund through the project TAMOP-4.2.2.C-11/1/KONV-2012-0004 – National Research Center for Development and Market Introduction of Advanced Information and Communication Technologies (subproject I.6).

References

[1] Park, S. and Hashimoto, S., "Autonomous mobile robot navigation using passive RFID in indoor environment", *IEEE Transactions on Industrial Electronics*, vol. 56, no. 7, July 2009, pp. 2366–2373.

[2] Raghavan, A., Ananthapadmanaban, H., Sivamurugan, M., and Ravindran, B.,"Accurate mobile robot localization in indoor environments using bluetooth", *in IEEE International Conference on Robotics and Automation (ICRA)*, May 2010, pp. 4391–4396.

[3] Biswas, J., and Veloso, M., "WiFi localization and navigation for autonomous indoor mobile robots", in *IEEE International Conference on Robotics and Automation (ICRA)*, May 2010, pp. 4379–4384.

[4] Kim, D.E., Hwang, K. H., Lee, D. H., Kuc, T. J., "A Simple Ultrasonic GPS System for Indoor Mobile Robot System using Kalman Filtering", SICE-ICASE, 2006, pp. 2915–2918.

[5] Zhang, Y., Wu, C., Cheng, L., Chu, H., "Localization and Tracking of Indoor Mobile Robot with Ultrasonic and Dead-reckoning Sensors", *Journal of Computational Information Systems*, 2012, vol. 8, no. 2, pp. 531–539.

[6] Huang,Y. P., Wang, J. S., Huang, K. N., Ho, C. T., Huang, J. D., and Young, M. S., "Envelope pulsed ultrasonic distance measurement system based upon amplitude modulation and phase modulation," *Review of Scientific Instruments*, vol. 78, no. 6, 2007.

[7] Vishay, "IR Reciever Modules for 3D Synchronization Signals",*TSOP35D25*.

[8] Prowave," Air Ultrasonic Ceramic Transducers", *400ET080*.

[9] Boyd, S., Vandenberghe, L.,"Convex Optimization", Cambridge university press. ISBN 978-0-521-83378-3.

[10] Nagy, Cs., Biro, Z., Marton, L., "Development of an Ultrasound Based Tracking System for Indoor Robot Localization", *in 5th International Conference on Recent Achievements in Mechatronics, Automation, Computer Sciences and Robotics*, Tg. Mures, Romania, pp. 155–162, 2015.

ActaUniversitatisSapientiae
Electrical and Mechanical Engineering, 8 (2016) 29-41

Development of Motion Detection Algorithms Based on Simultaneous Execution Using Mobile Phone Sensors

Zsófia SÁNDOR[1], Gergely KIS[2]

[1] Data scientist, research manager, Budapest, e-mail: sandorzsofi@yahoo.com
[2] Department of Infocommunication, Corvinus University of Budapest, Budapest, e-mail: gergely.kis@uni-corvinus.hu

Manuscript received February 15, 2016; revised December 24, 2016

Abstract: The proliferation of sensor networks employing wireless data transmission technologies has paved the way for the collection of large amounts of measurement data. Several research teams have used this opportunity to develop algorithms aimed at gaining information from sensor data. Motion detection is one of the most actively researched areas. In this article, we present a system for examining motion detection in a general environment. In other words, motion forms are not identified with various wearable sensors; instead, we use the data collected by the sensors of mobile phones kept with almost all members of society now.

Keywords: Motion detection, sensor data.

1. Introduction

A boom of wireless communication technologies and specifically mobile phones has brought about institutional, social and cultural changes, just as predicted by the American writer George Gilder [1]. Technological innovation offers solutions to an increasing number of issues that used to seem insoluble due to technological constraints and the lack of measured data.

Within technological innovation in general, progress in terms of decreasing calculation costs has been especially fast. Consequently, tasks can now be tackled that require the processing of large quantities of information. A current buzzword is “Big Data”, meaning the handling of data quantities that cannot be managed with conservative data processing methods and tools. Various sensory information is a typical source of such high-volume data. The second relevant direction of technological development is related to sensor development: sensors are decreasing in size, becoming more and more accurate, and consume less energy. The spread of wireless sensor networks (WSN) allows for the real-

time transfer and processing of the data collected by a sensor, and also for connecting several sensors.

Studying human behavior is a key social research area. Gaining an insight into human behavior could yield widely applicable knowledge, for example in the fields of medicine, psychology, economic marketing, and health care. One of the projections of human behavior is physical movement; observing movement may lead to conclusions about the individual's behavior and habits.

Thus, scientists are facing a specific range of issues, but also have the toolset that could provide answers to those issues. Hence today's wide-ranging research is made about the usability of sensor-collected data for motion detection. Most of that research is conducted in laboratories, with sensors attached to test subjects' bodies.

This article presents a system based on the processing of data generated in everyday life, outside of the laboratory environment. The definition of the problem at hand is followed by the description of the data collection and transfer system, by the presentation of processing algorithms built on each other, and finally by a summary and conclusions.

2. Problem definition

A. Historical research

Sensors built into smartphones and other wearable devices can be used in a variety of ways. With the right method, they are suitable for detecting practically all forms of motion, from basic step recognition to recognizing complex actions and providing motivation for a healthy lifestyle. This variety (both in terms of methods and areas of utilization) is reflected by international technical literature.

Some research is aimed at specific areas, e.g. at finding the most accurate step counting method [2]. Using accelerometers, the margin of error is just a few steps. In an advanced version of this solution, speed can also be taken into consideration as a factor [3]. In this research, steps were counted during slow and fast walking, downhill and uphill walking, and while climbing stairs. The researchers used gyroscopes because they had found that an accelerometer does not yield accurate step numbers if the test subject walks slowly. The results are encouraging: the accuracy of step counting in slow walk was above 96% on a flat surface, more than 95% on inclining and declining routes, and higher than 90% when climbing stairs.

Basic actions such as walking, running, climbing stairs, sitting, standing, using an elevator, and jumping have been examined by several research teams. Some of these achieved 99.97% accuracy in action recognition using the

Random Forest method [4], while the Nearest Neighbor method yielded 93% accuracy in [5]. The limitations were only one phone, held in the test subject's trouser pocket in [4]; in another case [5], a single phone model was used, but at two locations.

Resting positions (standing, sitting, lying) can be detected highly accurately [6]. Besides these, walking and climbing stairs were examined in this research, which resulted in 86% accuracy using a decision tree. Others went farther than that, detecting car driving in addition to walking, running, cycling, sitting and standing in [7]. In the research, the effectiveness of the QDA (Quadratic Discriminant Analysis) and the k-Nearest Neighbor algorithms was tested in online and offline mode. In online mode, the accuracy of QDA was 95.8%, and that of k-NN was 93.9%; offline, QDA yielded 95.4%, and k-NN 94.5%. The phone position was fixed in these cases, too, and in research no. [7], its direction was also fixed.

In [8], the goal was to identify actions regardless of the position of the phones; but in this case, too, only the basic actions were examined.

In certain publications [9, 10, 11], the objective was to present a connection between a healthy lifestyle and movement detection. In one case [9], the researchers developed a step counter using accelerometer data and a neural network, aiming to detect false steps (a common error in step counters). The ultimate goal was to increase the reliability of the health preservation system. In another research, algorithms were developed for the recognition of uphill and downhill walking, walking on a flat track, climbing and descending stairs, and running in [10]. Again, the goal was health preservation. The accuracy of recognizing these actions was 93.2%, 97.4%, 97.6%, 98.8%, 92.2%, and 90.8% respectively. Not all analyses are aimed at the processing of health data: based on a market approach, a convenience function has been developed, focused on automatically changing the phone's settings at the start of certain activities such as running [11].

Besides the basic actions, the recognition of complex activities (such as hand washing, house cleaning, cooking etc.) has also been researched in [12]. A wide range of methods was employed. These (apart from the Naïve Bayes method) were suitable to identify basic actions with over 90% accuracy. However, the highest accuracy in the case of complex actions was merely 50%.

In most cases, wearable sensors provide good estimations for the user's activities; because of the fixed location [13, 14, 15].

As shown above, research is progressing towards several dimensions of complexity. Our goal is to separate as many medium-complexity actions as possible. (These include, for example, riding a bus, but not dish washing.) For wide-ranging usability, that should be achieved with mobile phones regardless of model and location.

B. The challenge

The objective of the system presented is to accurately identify movement in its natural environment, without additional devices. A solution may not become wide-spread if the users find the required device inconvenient or uncomfortable. One example is the need to use an ankle strap and a wristband at the same time may force users to change their daily routines. This is why mobile phone sensors were selected in our solution. People keep their phones with them during the active part of the day, so the phones usually move together with the users.

However, mobile phones present challenges other than algorithm development.

The first challenge is related to eliminating the above-mentioned inconvenience. Mobile phone usage is convenient if the system meets certain basic requirements such as low energy usage and data transfers, and optimized resource (e.g. memory) usage. These requirements must definitely be met for a solution to be easy to use.

The second group of challenges arises from variances across both users and devices. Movement detection is difficult because of differences between people's physiques, habits and movements; the same actions vary from person to person if we rely on time-series data of sensors. In addition, differences between mobile phone makes and models should also be addressed; the quality and accuracy of built-in sensors vary, and even the measurement units may differ. These challenges need to be resolved for the wide-spread proliferation of a system.

The third challenge involves ensuring the right response time with sufficient accuracy. The system will not be used in a lab environment, so, according to international literature, we should not expect the models to be as accurate as in a lab, especially because some of the activities to be differentiated are similar to each other.

C. Technical solution concept

Before outlining the solution concept, it is important to note that two options are currently available to researchers concerning the technicalities of data transmission and processing. Each of these options has its advantages and disadvantages, as explained below.

The problem is that sensor data are available in the mobile phone or other sensor-equipped device. They either need to be transferred to a processing server, or the processing must be performed by the device itself. In the former case, the issue of high data transfers – and thus higher energy consumption – must be resolved as this may be inconvenient to users if the data are transferred

via a mobile Internet connection. Also, the increased data intensity means that the server's capacity must be sufficient. On the other hand, the advantage of this option is that there is no data loss.

The main argument for in-device processing is that not all the data are required; algorithms running on the device can be used to obtain material information, and only that is transferred to a server. This is a major advantage because it saves both device and server resources. On the other hand, running the processing algorithms uses more memory, and the running jobs make the energy consumption higher, which can also be inconvenient to users. The advantages and disadvantages of the two methods are summarized in *Table 1* below.

Table 1: Assessment of data processing methods

	Data processing on server	**Data processing on device**
Amount of data to be transmitted	high	low
Local memory usage	low	high
Server storage space requirement	high	low
Information loss	no	yes
Energy cost	higher due to the data transfer	higher due to the running algorithms

We have opted for data processing by a server, primarily because we strove to keep all information obtained during the research.

In order to resolve the above issues and tackle the challenges, the system shown in *Fig.1* has been created. As we collect data from mobile phone sensors, a mobile phone application is one of the central elements of the system. This application collects data based on the configuration received from the messaging server, writes those data into a file, compresses the file, and transfers it to the processing servers in the required format. The role of the data server is to pre-process the high quantity of received data, and to send them to the algorithm servers. The processing servers run the algorithms, and then estimate which activities the user is probably performing. The final decision is made based on several partial decisions.

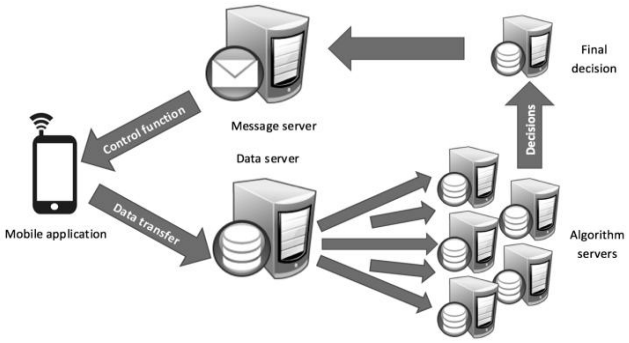

Figure 1: Data processing

3. Measurement and data collection

Defining the right data collection methodology is essential both for modelling and maintaining the operability of the system. As customizability as well as easy and convenient usage are paramount to users, we aimed not only for collecting and transferring data in the highest quality possible, but also for optimal resource utilization.

The mobile phone application we have developed has a built-in algorithm to sense low-activity periods when data are not collected. Naturally, this so-called sleeping mode is customizable via the configuration of the messaging server, i.e. the running and discontinuation of measurement can be controlled. The sleep mode optimizes both data traffic and battery usage and also the energy consumption.

Sensor data are obtained through the sensor's API (Application Programming Interface) and are stored in a manner that minimizes data size as much as possible. This size reduction consists of two parts: transforming and encoding the file in an optimized size; and compression. The resulting file (ready for transfer) is about 12% of the original data size. In addition to this size reduction, sleep mode may significantly lower the data transfer volume, depending on the activity ratio. The range of sensors to be used for measurement can also be configured; the algorithms have been developed with the accelerometer, GPS and gyroscope in focus.

Transferring the collected data to the server in the manner described above causes minimal inconvenience to the user. This means that the energy consumption is reduced, and also the amount of processing on the device itself

keeps the user experience high. All further work phases of the system are independent of the user, i.e. they do not influence the user at all.

The data are received by the data server's pre-processing module which prepares the data for processing, i.e. decompresses and decodes them, and runs processes required for subsequent algorithms. These include ensuring the coherence of the time series, decreasing their dimensions, and interpolation in order to eliminate inaccuracies (minor differences in sampling at a given frequency) in the sensor API. One of the gravest problems of data collection with mobile phones is the variance in the type and quality of sensors built into the devices of various manufacturers; the measurement scales may differ, and the accuracy of the sensors almost certainly varies. Figure 2 shows deviations in accuracy. The y axis is the acceleration in m/s^2, and the x axis is the millisecond of the measurement.

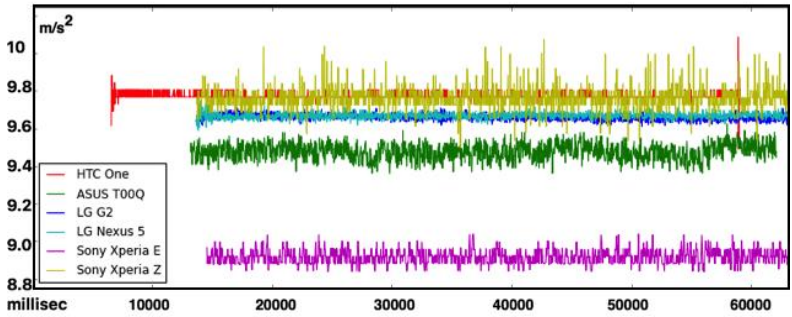

Figure 2: Comparison of the operation of mobile phone sensors

*Fig. 2*shows time series of the accelerometer sensors of the mobile phones tested. All phones were stationary, but the sensor readouts indicate different mean values and standard deviations. These differences must be managed with algorithms. The figure above shows data from devices running the Android operating system only; iOS measurements are very different due to the data being recorded in different measurement units. Android phones measure acceleration in m/s^2, while iOS in g. The gravity in common is measured in units of acceleration, g means the measurement unit of gravity, thus 1 g equals to approximately 9.81 m/s^2 in our case. So unifying the measurement scales is an important task within pre-processing phase.

4. Algorithm development

A. Activities

The first step in algorithm development involves defining the activities to be identified. Several scenarios are examined in the related technical literature

using the methods described above, from simple step count to attempting to identify the placement of the device. In our own research, we attempted to identify actions regardless of the placement of the user's phone, i.e. we seek patterns that are common to all phone placements. As an example, let us take a phone kept in the user's trouser pocket or purse. These cases differ because a phone kept in a trouser pocket moves together with the user's body, while a telephone kept in a purse moves partly independently of the body. But the basic movement pattern must be the same in both cases; it is this pattern that we aim to find with algorithms, ignoring the noise caused by the placement of the device.

During the research, we collected as detailed reference data as possible, but some of the actions were aggregated because they were similar. Reference measurements for the following actions were made:

- Human movement: immobility (phone placed on a desk or kept with the user); walk; run; ascending and descending stairs; phone usage (calls and screen usage);
- Vehicular movement: bus; metro; tram; trolley; train; suburban train;
- Other: lift and escalator (ascending or descending).

B. Concurrent algorithms

After pre-processing of accelerometer data, the system processes branch off so that several algorithms can run concurrently for the best possible end-result. As each algorithm has strengths and weaknesses, they would not be sufficient individually. But they support and improve each other for a robust overall system.

For example, one algorithm, which is based on GPS usage, can significantly improve the results of other algorithms. But for that, the user must turn on GPS-based location identification in the phone's menu; and even if that is enabled, there is no GPS connection indoors. So, that algorithm is used as an accessory function only; our algorithms are mostly based on an accelerometer.

Pre-processing makes the data easily usable for the concurrent algorithms. This means, that all the data packages that the algorithm servers receive do not need any additional calculations.

There are two types of algorithms. The first type is when we identify episodic types of events while the second is when we identify the activities as macro-processes.

All of the activities have typical episodes. For example, when a vehicle accelerates; when we take a step; when we sit down and so on. These episodes last for very short periods but they are pretty easy to identify. The first algorithm catches these episodes and is based on clustering. The variables of the clustering are the characteristics of the time series. We used the k-means

clustering algorithm for generating the static clusters for the episodes, and the distances from the cluster centers helps us to assign the new samples to the clusters and recognize the activities this way.

The second group is the macro-process type algorithms. The first amongst these algorithms is the GPS algorithm. The GPS algorithm is mostly based on the calculation of the speed based on the time and the distance. GPS coordinates are pretty accurate, so we can use the haversine formula to calculate the distance between two coordinate pairs (1).

$$d = 2r * arcsin\left(\sqrt{sin^2\left(\frac{\phi_2 - \phi_1}{2}\right) + cos\,\phi_1 * cos\,\phi_2 * sin^2\left(\frac{\lambda_2 - \lambda_1}{2}\right)}\right)$$

Where r is the radius of the Earth that is approximately 6371km, λ2- λ1 is the difference of the two longitudes and ϕ2-ϕ1 is the difference of the two latitudes. The computed and corrected speed values help us to identify the activities, as the activities can be well separated using these values.

The second algorithm of this group is based on the similarity of the 3-dimension accelerometer time series. Time series data points can be paired to each other and – as acceleration is measured in 3-dimensions – Euclidean distance can be calculated in 3 dimensions (2). Using the distances, activities can be categorized by comparing the differences to the reference data.

$$d(\mathbf{x}, \mathbf{y}) = \sqrt{\sum_{i=1}^{n}(x_i - y_i)^2} = \sqrt{(x_1 - y_1)^2 + (x_2 - y_2)^2 + (x_3 - y_3)^2} \qquad (2)$$

The third algorithm of this group is based on the amplitude of the time series. This algorithm is the best in recognizing the real macro activities. This is able to recognize the stops of a public transport vehicle and this way it can identify the travelling on public transportation as a process. The highest and lowest amplitude sections' patterns are unique by the activities.

The last algorithm in this group is based on a matrix model. In this model, we create unique "masks" of each reference activities. Activities differ in the sense how data points follow each other so if we use a matrix to identify the sequentiality of the data points, we can create matrices that can be used to identify the unknown activities. These "masks" create a similarity measure that varies between -1 and 1, and where 1 means the identity.

C. Real life examples

Table 2 and 3 each show the similarities/differences between the activities for one algorithm. The figures indicate the similarity of the selected basic actions.

Table 2: Similarities of basic actions as indicated by sensor data

	bus	stairs	lift	metro	immobility	walking	phone usage	tram	train
bus	0,91	0,67	0,25	0,70	0,31	0,66	0,71	0,54	0,65
stairs	0,67	0,95	0,36	0,46	0,51	0,95	0,43	0,39	0,46
lift	0,25	0,36	0,89	0,45	0,77	0,40	0,06	0,62	0,43
metro	0,70	0,46	0,45	0,77	0,44	0,47	0,58	0,66	0,70
immobility	0,31	0,51	0,77	0,44	0,82	0,55	0,12	0,56	0,43
walking	0,66	0,95	0,40	0,47	0,55	0,97	0,43	0,41	0,48
phone usage	0,71	0,43	0,06	0,58	0,12	0,43	0,80	0,38	0,59
tram	0,54	0,39	0,62	0,66	0,56	0,41	0,38	0,70	0,63
train	0,65	0,46	0,43	0,70	0,43	0,48	0,59	0,63	0,82

Table 2 contains the similarity measures of the matrices that we have created in the matrix model. The theoretical minimum value is -1, while the theoretical maximum is 1. The main diagonal indicates the average similarity between identical actions, while the rest shows the distance between the actions. Note the strong similarity between walking and stair climbing, which is caused by the fact that stair climbing consists of steps, too. Interestingly, the differences between users' movements are so big that the two actions can hardly be distinguished. The only difference between them is in their amplitude, which this algorithm is less suitable to detect. Note the similarity of moving in an elevator and being immobile; the reason is that a lift usually moves in a straight line at a steady pace apart from the initial acceleration and the concluding deceleration, i.e. an accelerometer indicates the same data as in the case of immobility. Acceleration is zero both in case of immobility and during straight movement at a steady speed.

Table 3 shows Euclidean distances in the 3-dimension accelerometer time series measured in m/s^2 between reference activities – hence the zeros in the main diagonal. Some activities constitute groups of stronger similarity; and walking is similar to climbing stairs in this case as well. Also, the data measured in vehicles are similar. But this algorithm seems to better distinguish riding an elevator from a state of immobility, so it is important to combine several algorithms.

Table 3: Distances between reference actions

	immobility	walking	stairs (down)	stairs (up)	bus	metro	tram	lift	escalator	train
immobility	0,00	4,42	4,22	4,49	0,39	0,49	0,60	0,38	0,25	0,22
walking	4,42	0,00	2,46	2,80	4,10	4,02	4,43	4,18	4,28	4,23
stairs (down)	4,22	2,46	0,00	2,50	3,94	3,82	3,88	3,97	4,07	4,05
stairs (up)	4,49	2,80	2,50	0,00	4,20	4,10	4,40	4,28	4,63	4,32
bus	0,39	4,10	3,94	4,20	0,00	0,27	0,35	0,33	0,24	0,24
metro	0,49	4,02	3,82	4,10	0,27	0,00	0,38	0,32	0,33	0,32
tram	0,60	4,43	3,88	4,40	0,35	0,38	0,00	0,47	0,45	0,44
lift	0,38	4,18	3,97	4,28	0,33	0,32	0,47	0,00	0,30	0,30
escalator	0,25	4,28	4,07	4,63	0,24	0,33	0,45	0,30	0,00	0,15
train	0,22	4,23	4,05	4,32	0,24	0,32	0,44	0,30	0,15	0,00

For robustness, development was supported by large-sample data collection in order to map differences between users and to avoid "over teaching", a frequent error when developing models. In this modeling case, it would be rather problematic if the algorithm learned too much from a single user's movement and used that to draw conclusions about the movement of other people who walk slower or use a different means of public transport the patterns of which cannot be observed in the first user's case, so the algorithm could not recognize them. We involved 80-100 users (in two phases) to help test the system under everyday circumstances.

The system can provide the final guesses concerning the activities performed in 2-4 minutes from receiving the data files, and is scalable depending on the number of users. Based on our back-testing (consisting of a 90-minute combination of city travel and actions by a small sample of users), the accuracy of the guesses was 67%.

5. Conclusion

The system presented in this article is suitable for detecting motion in its most natural form, with devices that are the most widely available. The conclusion of the research is that it is worthwhile to define several algorithms that complement each other, in order to improve the accuracy of action detection.

It is hard to compare the results to the results of other researches in the literature, because the goal of the research was pretty different from the previous solutions. In this paper our goal was to recognize the motions in their

natural way, and not in a laboratory environment. We have used a wide range of devices, more people, and several device positions. We also tried to recognize complex activities (such as traveling on tram). The statistical performance of this system is naturally lower than in a laboratory environment, and also compared to simpler models aimed at identifying just a few actions. As we could also see in the literature, accuracy decreases as we allow more and more flexibility in the system; however, this system, too, becomes more accurate as actions are aggregated or flexibility is reduced. The closest research found in the literature is [12], and compared to the 50% accuracy, our 67% is satisfactory. The results of this research provide satisfactory answers to the challenges explained at the beginning of the article. A balance has been created between the need for accuracy and the objective of universality. A system has been worked out that can be used conveniently by anyone and offers sufficiently fast response times. Further fine-tuning is underway.

References

[1] Freeman, C., Louca, F., "The Emergence of a New Techno-Economic Paradigm: The Age of Information and Communication Technology (ICT)", *As Time Goes By – From the Industrial Revolutions to the Information Revolution*, New York, 2002, pp. 318–324.

[2] Naqvi, N. Z., Kumar, A., Chauhan, A., Sahni, K.,"Step Counting Using Smartphone-Based Accelerometer", *International Journal on Computer Science and Engineering*, 4(3), pp. 675–681,May. 2012.

[3] Jayalath, S., Abhayasinghe, N., Murray, I.,"A Gyroscope Based Accurate Pedometer Algorithm", Presented at *International Conference on Indoor Positioning and Indoor Navigation*, Oct. 2013.

[4] Cruz-Silva, N., Mendes-Moreira, J., Menezes, P., "Features Selection for Human Activity Recognition with iPhone Inertial Sensors", Presented at *Portuguese Conference on Artificial Intelligence*, 2013.

[5] Das, S., Green, L., Perez, B., Murphy, M., "Detecting User Activities using the Accelerometer on Android Smartphones", Jul. 2010.

[6] Cerqueira da Silva, J. R., "Smartphone Based Human Activity Prediction", *M.S. thesis, Faculdade de Engenhariau., Universidade do Porto.*, 2013.

[7] Siirtola, P., Röning, J., "Recognizing Human Activities Userindependently on Smartphones Based on Accelerometer Data", *International Journal of Interactive Multimedia and Artificial Intelligence*, 1(5), pp. 38–45, Jun 2012.

[8] Khan, A. M., Lee, Y.-K., Lee, S. Y., Kim, T.-S., "Human Activity Recognition via An Accelerometer-Enabled-Smartphone Using Kernel Discriminant Analysis". Presented at*Future Information Technology, 2010 5th International Conference*, May. 2010.

[9] Tomlein, M., Bielik, P., Krátky, P., Mitrík, S., Barla, M., Bieliková, M.,"Advanced Pedometer for Smartphone-based Activity Tracking", Presented at *Proceedings of the International Conference on Health Informatics*, 2012.

[10] Shin, J., Shin, D., Shin, D., Her, S., Kim, S., Lee, M., "Human Movement Detection Algorithm Using 3-Axis Accelerometer Sensor Based on Low-Power Management Scheme for Mobile Health Care System", Presented at *GPC'10 Proceedings of the 5th international conference on Advances in Grid and Pervasive Computing*, May. 2010.

[11] Kwapisz, J. R., Weiss, G. M., Moore, S. A., "Activity Recognition using Cell Phone Accelerometers", *ACM SIGKDD Explorations Newsletter*, Dec. 2010.

[12] Dernbach, S., Das, B., Krishnan, N. C., "Simple and Complex Activity Recognition Through Smart Phones", Presented at *Intelligent Environments (IE), 2012 8th International Conference*, Jun. 2012.

[13] Yang, C.C., Hsu, Y.L., "Development of a wearable motion detector for telemonitoring and real-time identification of physical activity", *Telemed. J. E. Health*, no. 15, 2009, pp. 62–72.

[14] Karantonis, D.M., Narayanan, M.R., Mathie, M., Lovell, N.H., Celler, B.G., "Implementation of a real-time human movement classifier using a triaxial accelerometer for ambulatory monitoring", *IEEE. Trans. Inf. Technol. Biomed.* no. 10, 2006, pp. 156–167.

[15] Ozdemir, A. T. and Barshan, B., "Detecting Falls with Wearable Sensors Using Machine Learning Techniques" *Sensors*, vol. 14, MDPI, pp. 10691–10708, 2014.

Acta Universitatis Sapientiae
Electrical and Mechanical Engineering, 8 (2016) 42-61

Evaluation of Different Thin-Client Based Virtual Classroom Architectures

Örs DARABONT, Konrád József KISS, József DOMOKOS

[1] Department of Electrical Engineering, Faculty of Technical and Human Sciences,
Sapientia Hungarian University of Transylvania, Tg. Mureş,
e-mail: darabont_ors@yahoo.com, konrad@ms.sapientia.ro, domi@ms.sapientia.ro

Manuscript received November 20, 2016; revised February 26, 2017

Abstract: This paper presents an evaluation of different methods used to deliver virtual machines capable of being accessed remotely by thin-clients. The objective of the research was to provide a recommendation for building a cost-effective computer infrastructure for use in two scenarios: as a programming lab, and as an office infrastructure.

We have found that different thin-client solutions based on single board computers are reliable solutions for commercially available thin client replacement, because they can run free Linux-based operating systems, can handle Remote Desktop Protocol, have lower acquisition costs, lower power consumption and offer almost the same computing performance.

For providing remote desktops, there are several methods and virtualization platforms available. We benchmarked some of these platforms in order to choose the one best-suited for implementation. Our conclusion is that Microsoft Remote Desktop Services outperforms the virtualization based solutions, but it entails high license fees. Of the virtualization solutions tested, the VMW are ESXi based one is the most reliable choice.

Keywords: platform virtualization, virtual machine, virtual classroom, remote desktop, thin-client/server architecture.

1. Introduction

The fast spread of computer networks with broadband Internet access, along with the rapid development of different operating systems, virtualization technologies and platforms, makes it possible to use different computing solutions in many domains of interest [1]. The release and spread of virtualization platforms has allowed the development of cost-effective information systems capable of providing dynamic resource management and simplified system administration [2], [3]. Thus, the educational process in the fields of informatics, computer science, information technologies and

communications or other related domains can be optimized to minimize the cost of educational lab resources [4], [5]. Universities often use virtualization technologies to reduce the total cost of IT equipment and management, and also to simplify management of resources and technologies [6], [7]. These observations apply also in the case of the office infrastructure.

An infrastructure introduced by VMware (http://www.vmware.com/) for supporting desktop virtualization for research and educational user communities is presented in paper [8]. The software infrastructure is called VMLab. These software resources were used to develop also the Virtual Desktop benchmarking toolkit for thin-clients called VDBench [9].This toolkit can perform CPU usage, memory usage and network bandwidth measurements.

Paper [4] presents an original educational infrastructure using virtualization technologies implemented in a training process at the Kaunas University of Technology. This infrastructure allows university students and staff to access virtual desktops and applications online, both in computer classes at the University, and for self-working at home.

In paper [5] the authors underline how recent developments in network and hardware technologies have made thin-client/server architecture more efficient regarding the total cost of ownership, administration, maintenance, security and power consumption, and introduce such architecture in a Turkish Public College.

The novelty of this paper lies in the description and evaluation of different virtualization techniques and the delivery of virtual machines (VMs) or virtual desktops (VDs) that can be accessed remotely by using thin clients. The aim of our research was to find an optimal solution; one that helps reduce energy consumption and which allows us to create a cost-effective network infrastructure in terms of the acquisition cost of hardware and software licenses as well as long-term operating and maintenance costs. The objective of the project was also to develop a cost-effective computer infrastructure considering desktop virtualization technologies and thin clients for use in two scenarios: as programming labs and as office infrastructure if possible [1], [6], [7].

In the first stage of the project we studied different thin client solutions, from commercially available dedicated thin clients to alternative thin client solutions. The findings of this stage of the project are briefly presented in Section 2 and also in paper [6].

For providing remote desktops (RDs)or VMs, there are several virtualization software tools available [10], [12], [14] such as VMW are ESXi, Citrix Xen Server, Oracle VM VirtualBox or Microsoft Hyper-V, and other alternatives such as Microsoft Remote Desktop Services [7]. Our aim was to benchmark these solutions in order to choose the one best-suited for the implementation of

a virtual classroom and an office. Section 3 of the paper introduces the benchmarked system architectures and describes the test environments.

Section 4 contains the performed evaluation tests with detailed discussion on the results obtained.

Section 5 summarizes the main conclusions and the paper concludes with acknowledgements and references.

2. Thin-client solutions

Thin-clients are typically small form factor computing devices which are able to act as interfaces between the user and a server computer [15], [16], [17], [18]. These thin-clients include peripherals such as keyboard and mouse connectors (PS/2 or USB) for registering input signals from the user, video output (HDMI, VGA, DVI, etc.) and network interfaces for communication with the server computer [6], [10].

Because the thin-client acts only as an interface – the actual computing being performed at the server side – it requires no powerful internal components, and hence requires low electrical power consumption. This also reduces the cost of manufacturing and thus the acquisition cost of a thin-client. Additionally, as most thin-clients do not require active cooling, noise pollution is reduced.

In a complex computing infrastructure it is common practice to have most of the servers in one or more datacenters, the users being located in different buildings or different geographical locations. In this way the servers are easier to maintain and service [2].

From a functional perspective three categories of thin-client device can be distinguished [7], [10], [15].

The **zero clients** (ultra-thin-clients) are very small devices that don't feature a standalone operating system. These devices are based on proprietary PC-over-IP protocols (PCoIP) and have hardware decoding (through dedicated chips) that don't even require a local CPU to function.

Typical thin-clients are embedded computer systems that feature most of the components of a PC. They have a CPU, RAM, persistent storage (flash memory or HDD), peripheral connections (USB) and they run a fully functional operating system (generally Linux-based). The user can also run applications locally if necessary.

Table 1: Comparison of tested thin-clients

Thin-client device	CPU	Memory	Video output	Audio Input-Outputs	USB	Operating system	Power	Price
HP T5545	VIA Eden x86, 1 GHz, single core	512 MB	VGA, DVI	Microphone input, headphones output	6 x USB 2.0	Windows CE 6, HP ThinOS (Linux-based)	50W	$299
NComputing N500	Numo 3 SoC ARM Cortex-A9, 1 GHz, dual core	1 GB	DVI-I	Microphone input, headphones output	4 x USB 2.0	None	5W	$200
Raspberry Pi Model B	ARMv6, 700 MHz, single-core	512 MB	HDMI	Headphones output	2 x USB 2.0	Raspbian (and other Linux distributions)	3.5 W	$35
Raspberry Pi Model B+	ARMv6, 700 MHz, single-core	512 MB	HDMI	Headphones output	4 x USB 2.0	Raspbian	3 W	$35
ODroid U3	ARMv7, 1.7 GHz, quad-core	2 GB	uHDMI	Microphone input, headphones output	3 x USB 2.0	XUbuntu (and other Linux distributions), Android	5W	$65
Cubie board 2	ARM Cortex-A7, 1 GHz, dual-core	1 GB	HDMI	Line input and output	2 x USB 2.0	Ubuntu, Android 4.2, other Linux distributions	3W	$65
Asus Google Nexus 7	ARM Cortex-A9, 1.2 GHz, quad-core	1 GB	7” touch screen	Built in microphone and speaker	1 x uUSB	Android 4.4.4 Kitkat	2.25W	$230
HTC Desire C	ARM Cortex-A5, 600 MHz, single-core	512 MB	3.5” touch screen	Built in microphone and speaker	1 x uUSB	Android 4.0.3	~1W	$15

The third category of thin-client includes **Personal Computers** (PCs). PCs can be used as standalone computing devices and also as thin-clients. From the

perspective of functionality in thin-client mode they act like a dedicated thin-client, possibly featuring higher performance than dedicated thin-clients.

From the hardware perspective we can differentiate two categories of thin-client [6], [7]. The first category consists of purpose-built, commercially available thin-clients (these include zero clients and thin-clients). These are products that were built to function specifically as thin-clients. The second category features multi-purpose embedded computers, single board computers and development boards that can be adapted and configured to function as thin-clients. While commercially available thin-clients tend to have a high acquisition price, single board computers are becoming faster and cheaper at a rapid pace [7]. See table 1 for acquisition cost details.

Because of the great potential of single board computers regarding performance and their low acquisition cost, we conducted experiments to find out if these devices are a viable alternative to commercial thin-clients or not. We also wanted to pinpoint any technical limitations and address them.

Table 1 presents a comparison between the tested thin-client equipment. We studied different thin-client solutions: commercially available dedicated thin-clients and alternative thin-client solutions such as Android based smart devices and single board computers such as the Raspberry Pi, Odroid U3 and Cubie Board. We found that alternative thin-clients are a reliable solution for commercially available thin-client replacement. These single board computers are flexible, they can run Linux-based operating systems, can handle Remote Desktop Protocol (RDP) using the freeware FreeRDP software application [13] for connecting to a server, the acquisition costs are an order of magnitude cheaper than that of commercially available thin-clients and they also have lower power consumption [7]. Furthermore, all the operating systems, software applications and tools used at the thin-client side are freely available, therefore in cases where a large number of client stations are required, we can achieve large savings in license cost.

3. Benchmarked architectures

The main purpose of this paper is to present a comparative performance study of the different virtualization techniques and delivery of VMs and VDs. For remote desktop virtualization there are several virtualization software tools including VMWare ESXi, Citrix Xen Server, Oracle VM VirtualBox or Microsoft Hyper-V, and other alternatives such as Microsoft Remote Desktop Services. We have tried to evaluate some of these solutions. The benchmarked systems and test infrastructure are presented here.

All measurements were made on the same hardware configuration, an HP ProLiant DL380p Gen 8 server station with an Intel C600 Series chipset,

equipped with one Intel Xeon E5-2609 v2 @ 2.5 GHz processor having 4 cores, 8×4GB DDR3 memory, Smart Array P420i/ZM storage controller and 3×500 GB 7200 RPM SAS HDD, connected to the network through a 1Gb 4 port 331FLR Ethernet adapter.

With the complete test environment in place, performance tests were carried out to evaluate memory, CPU and storage management using the built-in benchmark tools.

3.1. Microsoft Hyper-V and Remote Desktop Services

The first test environment included the two Microsoft tools for Remote Desktop Virtualization: Hyper-V hypervisor and Remote Desktop Services.

Fig. 1 presents the test environment for both Remote Desktop Services (RDS) and Hyper-V based Desktop Virtualization scenarios. We used the Windows 2012 Server R2 operating system with both Hyper-V role and RDS installed and configured on an HP ProLiant DL380p Gen 8 server station. On the left part of the figure are the RDS components, marked in green, and on the right the Hyper-V with the guest VMs and their clients marked in red. Hyper-V Manager was used to create and manage VMs.

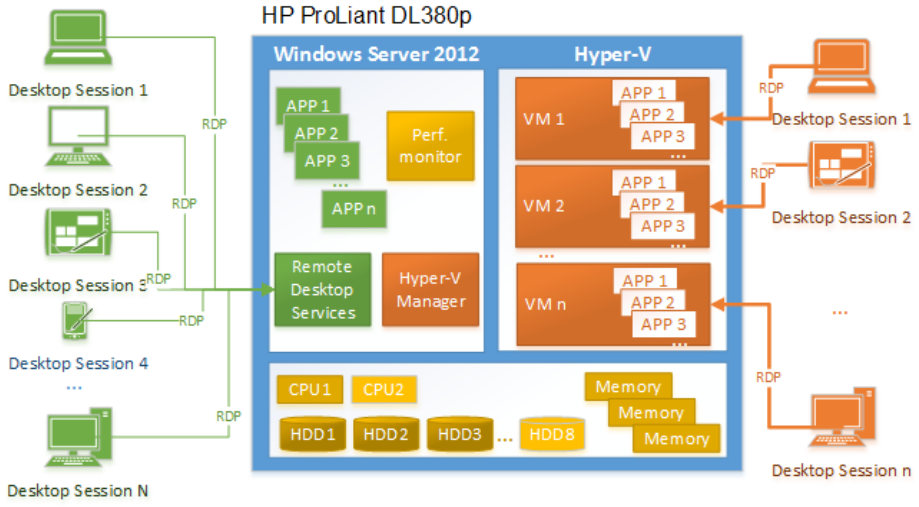

Figure 1: Test environment for Microsoft Hyper-V and Remote Desktop Services

A. Microsoft Hyper-V

Hyper-V is a proprietary virtualization environment created by Microsoft and included with the Windows Server edition operating systems. It is a Type 1 hypervisor that runs directly on the system hardware offering efficient

virtualization. The Hyper-V hypervisor is responsible for server hardware resource sharing between the created VMs. Several types of client presented in Section 2 can be connected to the VMs or directly to RDS such as personal computers, laptops, dedicated thin-clients or thin-clients based on single board computers, and even Android based smartphones or tablets. The connections are made using Remote Desktop Protocol (RDP).

In order to test the performance of both our Hyper-V and Remote Desktop Services implementations we used the built-in Performance Monitor tool, marked in yellow in Fig. 1.

For our tests in the Hyper-V scenario, we created 12 second-generation virtual machines, each with one quad-core virtual processor, 2 GB of RAM, IDE attached VHDx virtual disks and a virtual network adapter. The virtual machines (guests) were running Windows 7 Enterprise Edition operating systems. Fig. 2 shows the configuration of the virtual machines in the Hyper-V Manager application used for creating and managing the virtual machines. In order to improve storage I/O throughput, we disabled the page file and the System Restore service in the guest operating systems.

B. Remote Desktop Services

Microsoft Remote Desktop Services is an alternative means of offering virtual desktops to users. RDS is also included in Windows Server editions (2012 and up), but has a separate licensing strategy. The test environment is presented on the left side of Fig. 1. RDS makes multi-user connections to a Windows server station possible. Connected clients use RDP to transfer commands to the server and to receive the graphical user interface of the remote desktop. Connected users are granted access to all the software applications installed on the server station. In this case the received remote desktop is a Windows Server 2012 R2 desktop environment.

3.2. VMWare ESXi

The test environment used was based on an ESXi vSphere hypervisor core and the vCenter Server software management package. The two modules were deployed on two different machines: the vSphere on the HP ProLiant DL380p Gen 8 server station and the vCenter Server on a management server station. Reaching the management server was possible through HTTP protocol using the vCenterServer Web Client application.

The standalone management server improves system performance and capacity because the management software application doesn’t require an extra load for the server running VMs. However, this makes the configuration and administration of the whole system a little bit more difficult.

When the ESXi hypervisor is freeware, the vCenter Server management application can be purchased only with a license.

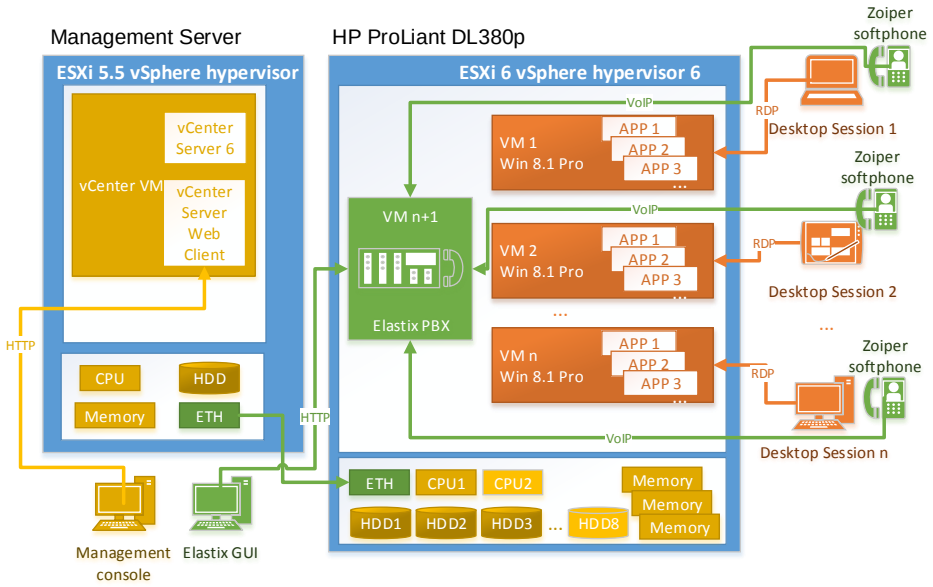

Figure 2: Test environment built using VMWare ESXi

3.3. XenServer

The XenServer is an open source virtualization platform previously developed by Citrix which is freely available to be used for any educational, research or business applications.

The platform contains a hypervisor core, namely the XenServer and a management application. The management application is built into the XenServer and can be reached via HTTP protocol using the Citrix XenCenter application.

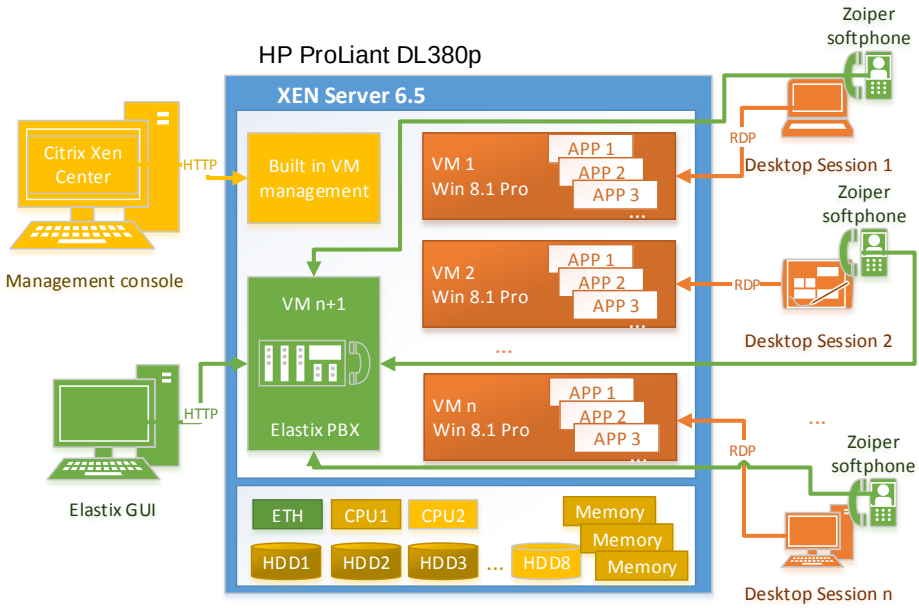

Figure 3: Test environment built using XenServer

In all virtualization test scenarios we used a virtual machine configured as a VoIP PBX. The software PBX used was Elastix.The PBX can be reached and configured through HTTP protocol using the Elastix GUI application. On the client side we installed Zoiper software IP phones. The advantage of this softphone, besides incurring no additional costs, is that it runs on different operating systems (Windows, Linux and Android), and its low resource requirement means it can be easily run on thin-client devices.

4. Measurements and test results

For the performance tests we configured 12 second-generation virtual machines, each featuring 4 CPU cores, 1 GB of RAM, SCSI virtual hard drive controllers and virtual network interface cards. The virtual machines ran Windows 8.1 Pro operating systems.

For testing the Microsoft Hyper-V and Remote Desktop Services scenarios we used the built in Performance Monitor application; an application included in the Windows Server 2012 operating system. Performance Monitor runs on the hypervisor host machine and is able to monitor the parameters of a local or remote computer in real time. It can handle more than 350 counters. The monitored data can be saved in files or can be graphically displayed.

For test measurements we initialized a data collector set which included the following counters:

- Hyper-V Hypervisor Logical Processor – % Total Run Time, - representing the total processor usage including the CPU load of the host operating system and also the CPU load of VMs,
- Memory – Committed Bytes – representing the total amount of memory used both by the host operating system and the VMs, in bytes,
- LogicalDisk – Average Disk Queue Length – representing the HDD queue in units.

These parameters were saved in the test measurement log files with a period of 1 second.

The same parameters were monitored in the VMW are ESXi vSphere scenario, but here they have different names:

- *CPU utilization (%),*
- *RAM allocation (bytes),*
- *Logical disk latency (ms) – instead of disk queue length.*

The XenServer offers a limited possibility of monitoring in comparison to Microsoft Hyper-V and VMWare vSphere. In this scenario we could monitor only 2 parameters, the:

- Average CPU *(%)*
- Used memory *(bytes)*.

CPU utilization test measurements using the Microsoft products are reported in conference paper [1]. The load for these tests was produced artificially on each VM using the CPU Stress application [11].

CPU utilization tests with realistic data are presented here. Two realistic CPU load tests were performed: virtual machine boot test and Matlab application start test. The tests were repeated in all test environments: Hyper-V, ESXi and XenServer based virtualization and also RDS.

4.1. Microsoft hyper-V and RDS platform test results

Fig. 5 shows the results of the Virtual Machine Boot test in the Hyper-V based scenario. The test began with the startup of the virtual machines (VMs) and ended when all of them had finished booting and remote desktop connections were engaged. We started all the VMs at once. The server had 3 hard drives installed, so we made a RAID 0 array for best performance. All 3 parameters logged are presented: CPU usage marked in red, RAM allocation in blue and disk queue highlighted in black.

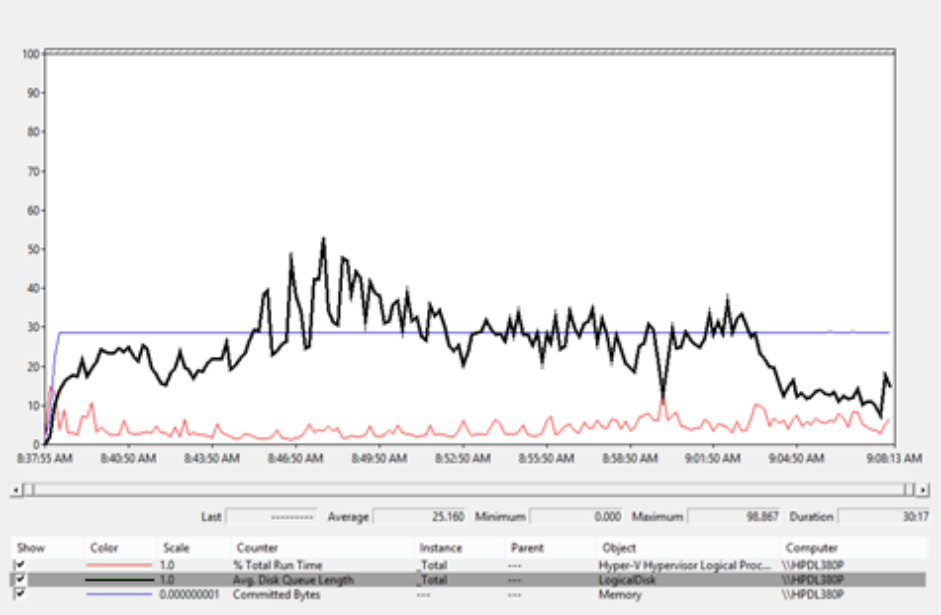

Figure 4: CPU usage, RAM allocation and disk queue variations measured during Virtual Machine Boot test in the Hyper-V scenario

The most important parameter that we can observe on the above chart is the Average Disk Queue Length (highlighted in black). This parameter shows how many disk commands were waiting in queue, to be executed. The average value is around 25-30. The maximum value should be the number of hard drives in the server, which was 3 in our case. This means that during the test, there was a severe performance bottleneck which slowed down the booting of the virtual machines. The performance bottleneck arose from the poor performance of the storage subsystem. The resulting performance bottleneck made the whole infrastructure almost unusable.

The committed bytes (RAM) performance counter(blue) shows an abrupt increase shortly after the start of the test, resulting from the allocation of RAM for every virtual machine at startup. The virtual machines featured static RAM allocation.

The CPU utilization performance counter shows a low average value.

This also confirms the storage bottleneck, because the CPU has to wait until the storage subsystem returns the desired information.

In the second test we measured the time taken for the Matlab software platform to start. We have also monitored the performance counters mentioned above. Measured parameters are presented in Fig. 6. The test started with the first virtual machine giving Matlab the start command, followed by the same

thing replicated on all 12 VMs. The test finished when all the Matlab programs were running on all 12 VMs and the "Ready" message was displayed in the status bar.

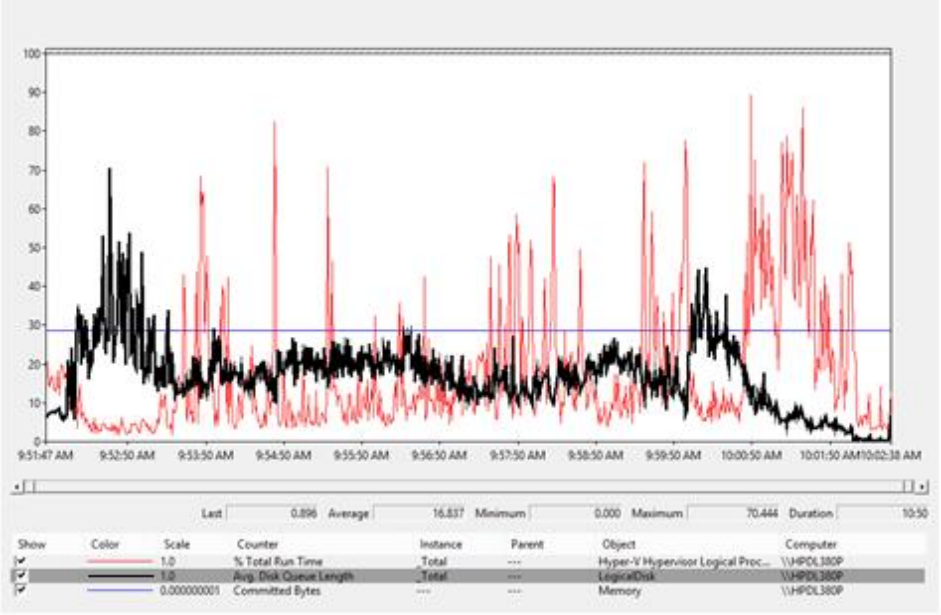

Figure 5: CPU usage, RAM allocation and disk queue variations measured during the Matlab startup test in Hyper-V scenario

Because the Matlab platform start time is shorter than the boot time of the VMs, we can observe more aggressive changes on the chart above. The Average Disk Queue Length counter (highlighted in black)still returned high values, while CPU utilization remained relatively low, indicating a severe performance bottleneck.

The same tests, repeated in the RDS scenario, reveal better RDS performances than those of the virtualization-based WM delivery solution.

The main difference is that there was no more virtualization and just one operating system running on the server. Only the applications started by the remote desktop users are multiplied. That is why there is a significantly lower memory amount in use. In comparison to the almost 20 GB RAM used for the host operating system and the 12 virtual machines in the hyper-V scenario, we see just around 10 GB RAM used in the RDS based test.

Fig. 6 shows much lower disk queue length values: the average is $0.464 < 3$. Additionally, much more processor occupancy is reached because the logic

drive bottleneck doesn't appear. High CPU utilization and low Average Disk Queue Length numbers indicate no logical disk performance bottleneck.

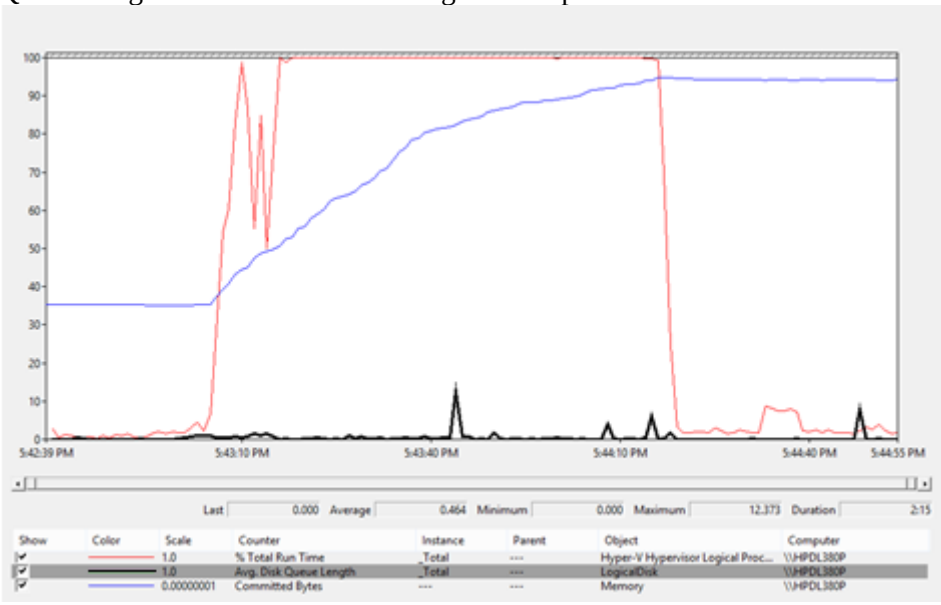

Figure 6: CPU usage, RAM allocation and disk queue values during Matlab startup test in RDS based scenario

4.2. VMWare ESXi platform test results

During testing of the ESXi virtualization environment, there were no performance counters monitoring the server, because Windows Server 2012 R2 wasn't installed. Instead, the virtual machines ran directly in the virtualization environment so only the built-in ESXi performance metrics were available.

The Virtual Machine Boot test had the same implementation as in Hyper-V.

The test began with all the VMs started at once, and finished when all were responsive and engaged in RDP connections.

Surprisingly, while using Hyper-V, the boot time of the VMs was around 40 minutes: in the ESXi implementation it only took 3 minutes to fully boot up.

The ESXi management VCenter Server was running on a different server station. In this way the management server was not an extra load for the server running the VMs.

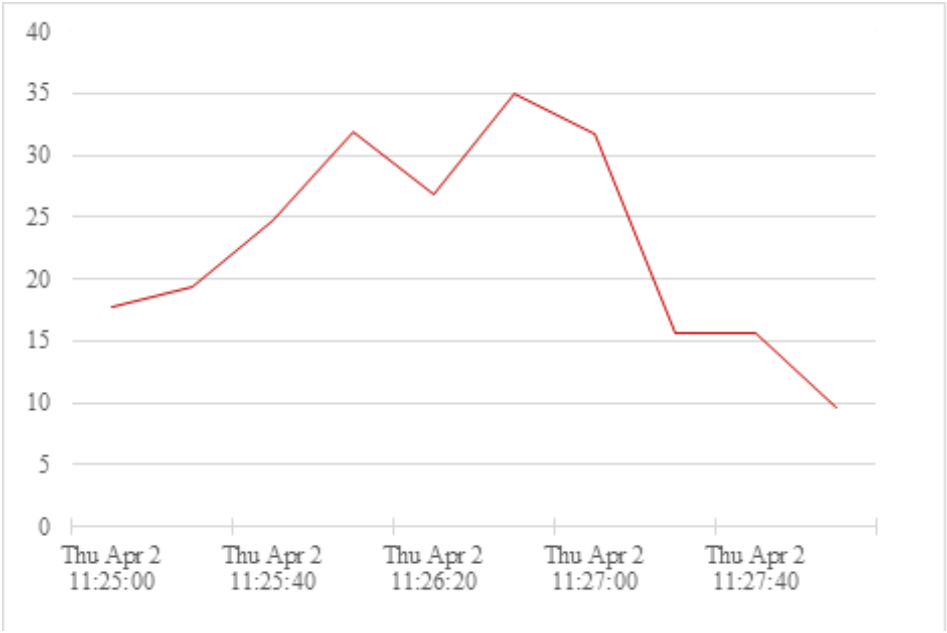

Figure 7: ESXi 6.0 Virtual machine boot test - CPU utilization (%)

Fig. 8 shows memory allocation test results and Fig. 9 shows logical disk latency test results for the ESXi VM boot test. In both tests ESXi slightly outperforms hyper-V.

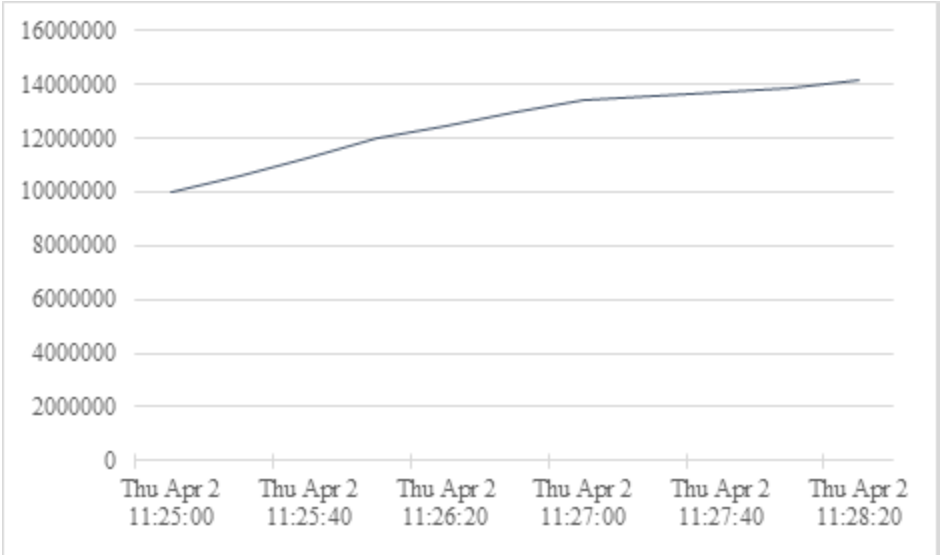

Figure 8: ESXi 6.0 Virtual machine boot test – RAM allocation (bytes)

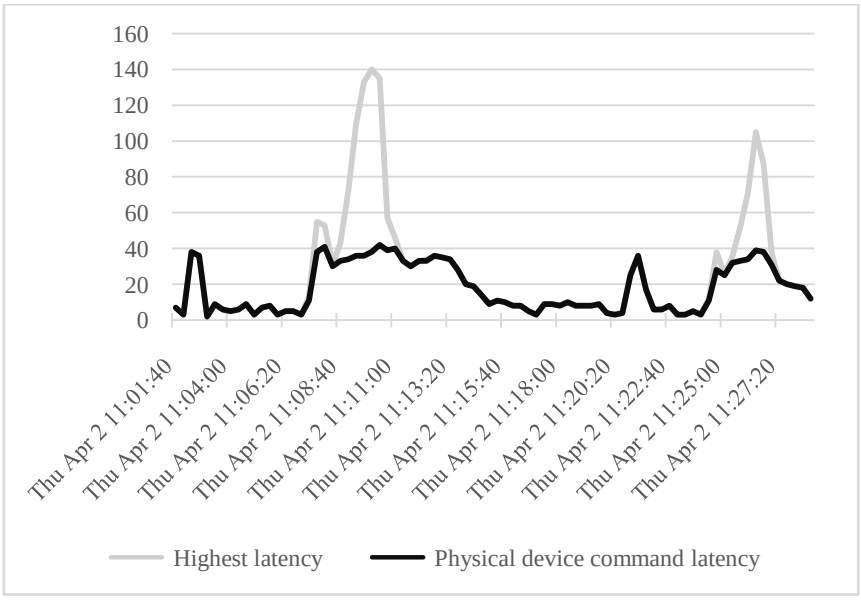

Figure 9: ESXi 6.0 Virtual machine boot test- Logical disk latency (ms)

In the following three figures: Fig. 10, Fig.11, and Fig.12are presented the CPU utilization, memory allocation and logical disk latency test results for the ESXi Matlab platform startup test.

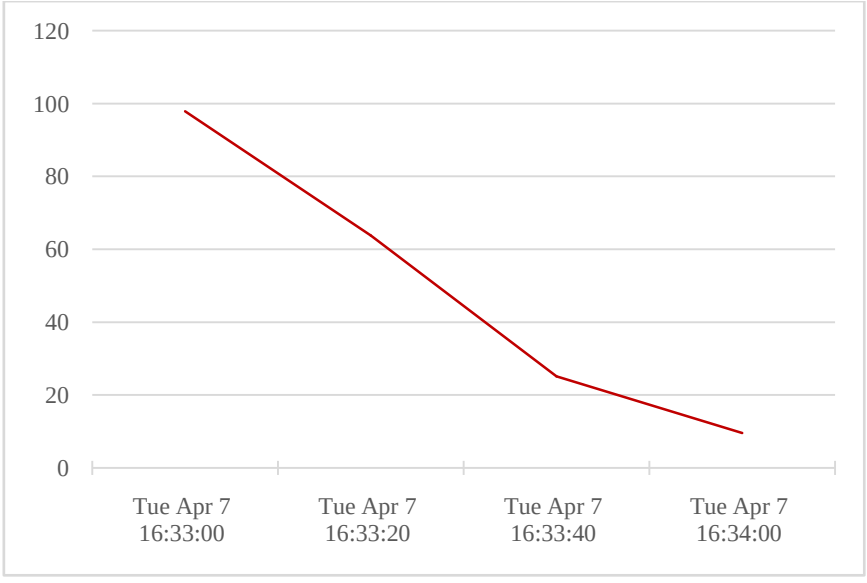

Figure 10: ESXi 6.0 Matlab startup test - CPU utilization (%)

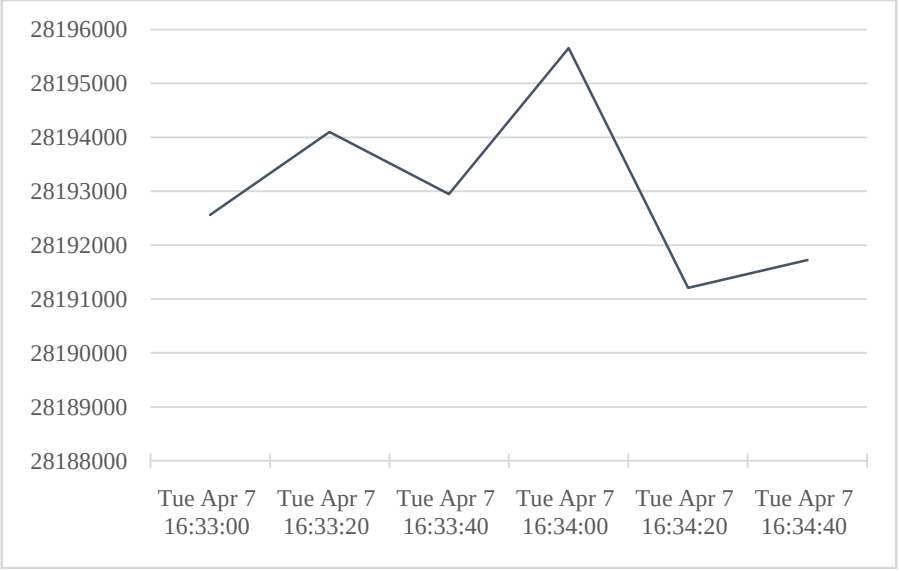

Figure 11: ESXi 6.0 Matlab startup test - RAM allocation (bytes)

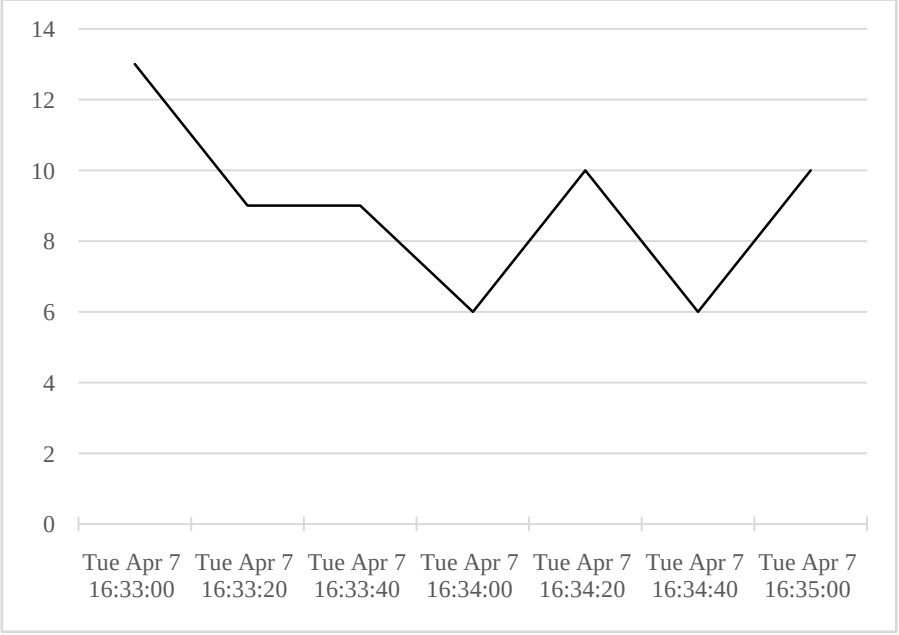

Figure 12: ESXi 6.0 Matlab startup test-Logical disk latency (ms)

4.3. XenServer platform test results

For the XenServer scenario, the procedure was then repeated the same way as for the Hyper-V and ESXi tests. The total run time for XenServer VM boot test was 4 minutes and 35 seconds. There were no disk latency or queue metrics provided in XenServer to measure logical disk performance. However, looking at the CPU utilization chart and the runtime of the test, we can deduce that a severe performance bottleneck, such as that seen with Hyper-V, is not present.

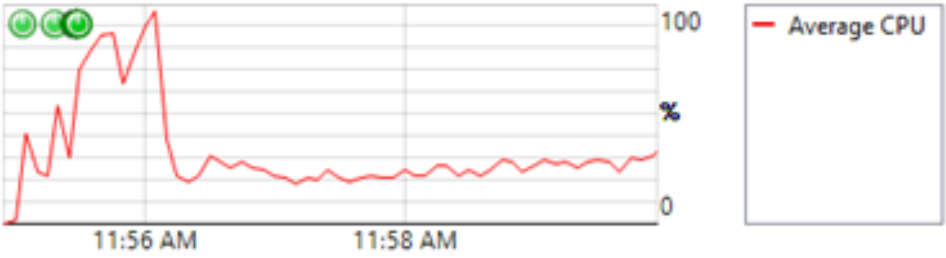

Figure 13: XenServer 6.5 Virtual Machine boot test - CPU utilization (%)

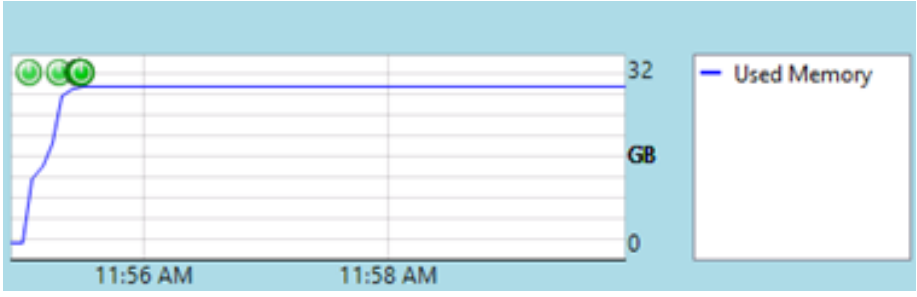

Figure 14: XenServer 6.5 Virtual Machine boot test - RAM allocation(GB)

The Matlab platform startup test was conducted identically to that of the Hyper-V and ESXi. Total runtime of the test was 2 minutes, 7 seconds. Test results are presented in Fig. 15 and Fig. 16.

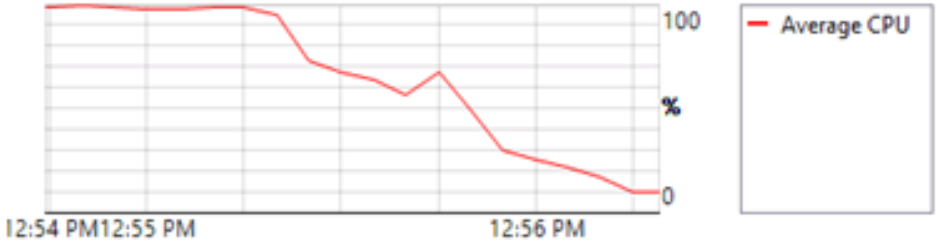

Figure 15: XenServer 6.5 Matlab startup test - CPU utilization (%)

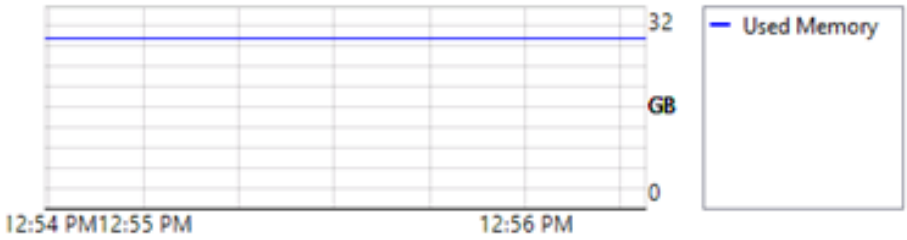

Figure 16: XenServer 6.5 Matlab startup test - RAM allocation (GB)

As static memory allocation was used for the VMs, the entire memory for VMs was allocated at startup.

5. Conclusions

In this paper we have presented experiments conducted in order to compare platforms that can be used for the implementation of a virtual classroom and an office IT infrastructure, and to identify the best solution among them.

We benchmarked different methods for delivering virtual machines capable of being accessed remotely by thin-clients and we performed tests with different thin-clients such as dedicated thin-clients, single board computers, smart phones and tablets. We have found that alternative thin-clients are a very good solution for commercially available thin-client replacement. These single board computers are flexible, they can run Linux-based operating systems, can handle Remote Desktop Protocol (RDP) for connecting to the server, the acquisition costs are an order of magnitude cheaper than those of the commercially available thin-clients and they also have lower power consumption.

There are several virtualization platforms available for providing remote desktops. We have benchmarked some of these platforms to see how they actually perform and have made a comparison between the acquisition and maintenance costs as well as relative performance.

We built three test environments for testing four scenarios. The first one was based on Microsoft products and tested hyper-V hypervisor and Remote Desktop Services, the second was based on VMW are ESXi and the third one was based on the Xen Server platform. For each platform we measured CPU usage, memory allocation and disk management parameters to evaluate the whole system performance. Tests were performed in a real network environment and using a real load.

We also studied the total cost of acquisition and the license fees for each technical solution.

Our main conclusion is that Microsoft Remote Desktop Services out-performs the virtualization based solutions, but incurs high license fees. From

the virtualization solutions tested, the VMW are ESXi based one seems to be the most reliable choice.

We highlight two problems encountered during our tests. The first is the high-level of memory usage in the case of virtualization based solutions compared to that of RDS. The second aspect is the bottleneck identified in the logical disk management. High-level disk queue was detected in the case of the hyper-V hypervisor based scenario. Both problems lead to a limitation of the number of delivered VMs.

Disk queue problems were not identified in the tests performed with the other two virtualization platforms. This can be explained by the fact that the management servers were running on a different server station, not on the one which delivers the VMs.

Acknowledgements

This research and its publication have been supported by the European Union and Hungary and co-financed by the European Social Fund through the project TAMOP- 4.2.2.C-11/1/KONV-2012-0004 - National Research Center for Development and Market Introduction of Advanced Information and Communication Technologies.

References

[1] Darabont Ö., Kiss K. J., Domokos J., "Performance Analysis of Remote Desktop Virtualization based on Hyper-V versus Remote Desktop Services", in *Proceedings of the 5th International Conference on Recent Acievements in Mechatronics, Automation, Computer Science and Robotics (MACRo 2015), Tîrgu Mureş, Romania*, 2015, pp. 125-135.

[2] Bari F., Boutaba R., Esteves R., Granville L. Z., Podlesny M., Rabbani G., Zhang Q., Zhani M. F., "Data Center Network Virtualization: A Survey", in *IEEE Communications Surveys & Tutorials*, Vol. 15, No. 2, pp. 909-918, 2013.

[3] Rodríguez-Haro F., Freitag F., Navarro L., Hernández-Sánchez E., Farías-Mendoza N., Guerrero-Ibanez J. A., González-Potes A., "A summary of virtualization techniques", in *Proceedings of the 2012 Iberoamerican Conference on Electronics Engineering and Computer Science*, Procedia Technology 3, 2012, pp. 267 – 272.

[4] Miseviciene R., Ambraziene D., Tuminauskas R., Pažereckas N., "Educational Infrastructure Using Virtualization Technologies: Experience at Kaunas University of Technology", in *Informatics in Education, Vol. 11, No. 2*, pp. 227-237, 2012.

[5] Cimen C., Kavurucu Y., Aydin H., "Usage Of Thin-client / Server Architecture In Computer Aided Education", *TOJET: The Turkish Online Journal of Educational Technology*, volume 13, issue 2, pp. 181-185, 2014.

[6] Darabont Ö, Kiss K. J., Domokos J., "Cost effective computer infrastructure" in *Proc. of the XXIVnd International Conference on Computers and Education (SZÁMOKT 2014), Székelyudvarhely, Romania,*, 2014, pp. 144-147.

[7] Domokos J., Kiss K. J., Darabont Ö., "Economic and technological analysis of the development of a virtual machine room", in *Proc. of the Annual Scientific Conference of*

the Hungarian National Coordinating Center for Infocommunications ASCONIKK, Veszprém, Hungary, 2014.

[8] Calyam P., Berryman A., Lai A.M., Honigford M., "VMLab: Infrastructure to Support Desktop Virtualization Experiments for Research and Education", in *VM-ware Technical Journal*, 2012.

[9] Berryman A., Calyam P., Honigford M., Lai A. M., "VDBench: A Benchmarking Toolkit for Thin-client based Virtual Desktop Environments", in *Proceedings of the 2nd IEEE International Conference on Cloud Computing Technology and Science*, 2010, pp. 480-487.

[10] Kaur D., Bhathal G. S., "Thin Client Cloud Computing: An Overview", *International Journal for Multi Disciplinary Engineering and Business Management*, Volume-2, Issue-1, pp. 8-10, 2014.

[11] MSDN blog Tools to simulate CPU / Memory / Disk Load, Windows sysinternals CPU Stress application: *http://blogs.msdn.com/b/vijaysk/archive/2012/10/27/tools-to-simulate-cpu-memory-diskload.aspx*.

[12] Pék G.,Buttyán L., Bencsáth B.,"A Survey of Security Issues in Hardware Virtualization", *ACM Computing Surveys (CSUR)*, Volume 45 Issue 3, pp 40:1-40:34, 2013.

[13] The FreeRDP software webpage: http://www.freerdp.com/

[14] Sahoo J., Mohapatra S., Lath R., "Virtualization: A Survey On Concepts, Taxonomy And Associated Security Issues", *in Second International Conference on Computer and Network Technology*, 2010, pp. 222–226.

[15] Nieh J., Yang S. J., Novik N.,"A Comparison of Thin-Client Computing Architectures", *Technical Report CUCS-022-00, Network Computing Laboratory, Columbia University*, 2000.

[16] Lai A. M., Nieh J., "On the Performance of Wide-Area Thin-Client Computing", *ACM Transactions on Computer Systems*, Vol. 24, No. 2, pp. 175–209, 2006.

[17] Baratto R. A., Kim L. N., Nieh J., "THINC: A Virtual Display Architecture for Thin Client Computing", *in Proceedings of the twentieth ACM symposium on Operating systems principles*, 2005, pp. 277-290.

[18] Jae Yang S., Nieh J., Naomi N., "Measuring Thin-Client Performance Using Slow-Motion Benchmarking", *in ACM Transactions on Computer Systems*, Volume 21, Issue 1, pp. 87-115, 2003.

Acta Universitatis Sapientiae
Electrical and Mechanical Engineering, 8 (2016) 62-78

Macroscopic Thin Film Deposition Model for the Two-Reactive-Gas Sputtering Process

András KELEMEN[1], Domokos BIRÓ[2], Albert-Zsombor FEKETE[1], László JAKAB-FARKAS[1], Róbert Rossi MADARÁSZ[1]

[1] Department of Electrical Engineering, Faculty of Technical and Human Sciences, Sapientia Hungarian University of Transylvania, Tg. Mureş, e-mail: kandras@ms.sapientia.ro
[2] Department of Mechanical Engineering, Faculty of Technical and Human Sciences, Sapientia Hungarian University of Transylvania, Tg. Mureş, e-mail: dbiro@ms.sapientia.ro

Manuscript received February 15, 2017; revised March 15, 2017.

Abstract: The presence of a second reactive gas in the magnetron sputtering chamber makes the process much more complicated, and the process control much more difficult than in the case of a single reactive gas. Macroscopic models have been developed in order to explain the complex phenomena and to provide support for the process control. These models are able to explain the nonlinearities of the process and the strong coupling between the control channels.

This paper introduces a model created with the intention to of gaining a good grasp of the process, especially regarding the conditions necessary to obtain the required stoichiometry of the film deposited on the substrate. For this purpose, we modelled the formation of the desired ternary compound both directly from the available particle fluxes and from intermediary compounds. The surface of the substrate is divided into eight dynamically variable regions, covered by different compounds, each exposed to the streams of five types of particles.

We present the analytical model and provide simulation results in order to demonstrate its capability toof describeing the nonlinear phenomena, which that characterisze the two-gas sputtering process.

Keywords: DC magnetron sputtering, thin film deposition, reactive sputtering, macroscopic modelling.

1. Introduction

DC magnetron sputtering and thin film deposition have been studied for decades and several attempts have been made for to modellling different aspects of these processes.

There exist several works dedicated to the study of the reactive DC magnetron sputtering process, which basically consists of sputtering particles from the surface of a metallic target, in the presence of reactive gases. The sputtering effect is produced by Ar+ ions accelerated by a DC electric field.

A successful approach for to macroscopic modeling was provided by Berg et al. [1], [2], which is based on the equilibrium equations of the reactive gas and metal particles on the target and substrate surfaces. This model has been successful in explaining the strong nonlinearity and hysteresis effects observed in the process, and is simple enough to be used for process control.

Since then, a number of extensions of this idea have been formulated, to handle the cases wheren compound targets or more reactive gases are being used ([2], [11], [12], [13], [14]). Dynamic versions of these models have been formulated for control purposes ([7], [8], [9], [10], [15]).

In the presence of two reactive gases, the simplified modelling approach generally used is to consider, instead of the complex crystallization phenomena, only binary compounds MG_1 and MG_2 (ex. $MG_1 \equiv TiO_2$, $MG_2 \equiv TiN$), the participating average metal and gas quantities reflecting the stoichiometry of the real process.

This paper attempts to provide a more detailed macroscopic model of the thin film growth process by highlighting its intermediary phases. Thus, we assumed that the surface of the substrate consisted of areas covered by metal atoms (M), areas covered by reactive gas atoms (G_1 and G_2), areas covered by binary metal-gas compounds (MG_1 and MG_2) and by ternary MG_1G_2 compounds (*Fig. 1*). The coverage fractions represented by these areas are changing vary due to the incident particle fluxes and due to the bonds formed between the particles adsorbed to the surface of the substrate.

To some extent, the process is can be thought of as if different coloureds of paints were applied on to a surface, using atomizers.

Regarding the phenomena on the surface of the target, it is assumed that the Ar+ ion flux can sputter both metal atoms (ex. Ti) and molecules formed from this metal and the atoms of the reactive gases (ex. N, O). We used a single layer approach, i.e. removingal of an oxide or nitride molecule results in to leaves behind a metallic surface.

The target fractions covered by different compounds are denoted by θ_{ti}, i.e.

θ_{tM} - the metallic fraction of the target surface;

θ_{tMG1}, θ_{tMG2} - the fractions of the target surface „poisoned" by MG_1 and MG_2 binary compounds

Formation of ternary compounds was not taken into account on the surface of the target.

In the case of the substrate, the fractions covered by different compounds were denoted by θ_{si}, where $i \in \{M, G_{1a}, G_{2a}, MG_{1a}, MG_{2a}, MG_{1p}, MG_{2p}, MG_1G_2\}$, i.e.

θ_{sM} - the metallic fraction of the substrate surface;

θ_{sMG1G2} - the fraction covered by the ternary compound MG_1G_2;

θ_{sMG1a}, θ_{sMG2a} - the fractions covered by the „active" binary compounds MG_1 and MG_2 (formed on the surface of the substrate from incident atomic components, ready available to react with G_2 or G_1 respectively, considered as intermediateas intermediate phases of the MG_1G_2 crystal growth);

θ_{sMG1p}, θ_{sMG2p} - the fractions covered by the „passive" binary compounds MG_1 and MG_2 (resulted fromby sputtering of the same compounds from the surface of the target, expected to segregate on the grain boundary);

θ_{sG1a}, θ_{sG2a} - the fractions covered by the „active" gas atoms G_1 and G_2 (formed from incident gas atoms, ready to participate in the crystal growth, forming directly MG_1G_2, or „active" MG_1 and MG_2 compounds).

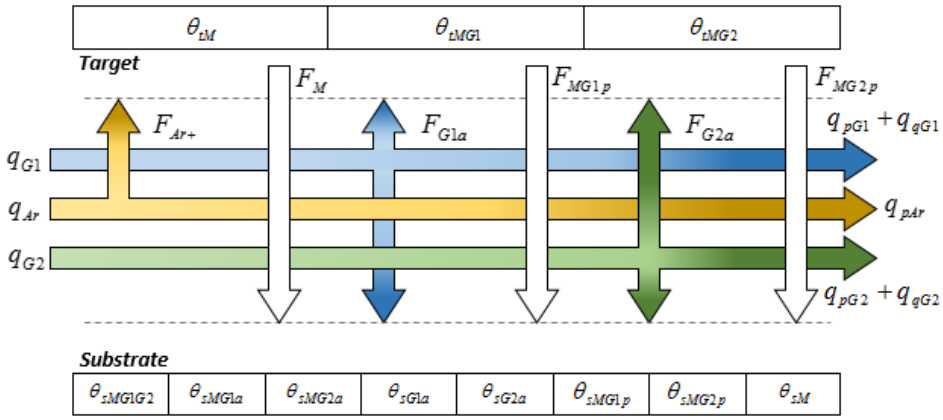

Figure 1: Particle fluxes in the sputtering chamber and coverage fractions on the target and substrate surfaces

The deposition of metal atoms on the metallic fraction does not influence the θ_{sM} fraction. In the same manner, deposition of G_i on θ_{sGia} does not change θ_{sGia}, deposition of MG_i on θ_{sMGip} does not change θ_{sMGip}. A reaction, which yields MG_i on the θ_{sMGia} fraction (using M and G_i deposited on the same fraction), does not modify θ_{sMGia}, and a reaction which yields MG_1G_2 on the

θ_{sMG1G2} fraction (using M, G_1 and G_2 deposited on θ_{sMG1G2}), does not modify θ_{sMG1G2}.

In this paper we assumed that different compounds are formed only on the surface of the substrate, and the formation of compounds in the plasma was neglecteddisregarded. We considered that the reactive gases reach the surface of the substrate in atomic form, after dissociation of the gas molecules in the plasma, in front of the substrate. These atoms were regarded as „active" and ready available to contribute to the crystal growth. We considered that the MG_1 and MG_2 molecules sputtered from the target, reach the substrate in the same form, without dissociation.

We assumed that the condensation surface is composed of the surface of the substrate and the surface fraction of the vacuum chamber situated in the range of the sputtered particles.

The surface of the chamber is present in the model as a getter pump with adsorption efficiencies α_{g1} and α_{g2} (1), corresponding to G_1 and G_2.

$$\alpha_{g1,2} = \frac{Nr\ of\ \ adsorbed\_gas\_molecules}{Nr\ incident\ \ gas\ molecules} \tag{1}$$

The following notation has been used in the article:

n_{MGi} - number of metal atoms in the molecule formed with the gas G_i (ex. $n_{MG1}=1$ in TiO_2, $n_{MG2}=1$ in TiN);

n_{GiM} - number of gas atoms in a binary molecule (ex. $n_{G1M}=2$ in TiO_2, $n_{G2M}=1$ in TiN);

n_{GiGi} - number of gas atoms in a gas molecule;

n_{MG1G2} – number of metal atoms in the MG_1G_2 compound (ex. $n_{MG1G2}=1$ in the TiO_2N molecule); in this model, it has been assumed that $n_{MG1G2}=n_{MG1}=n_{MG2}$;

x, y - number of G_1 and G_2 atoms in the MG_1G_2 compound;

$k_B=1.38e-23\left[\frac{J}{K}\right]$ - the Boltzmann constant;

$N_A=6.023e26\left[\frac{1}{kmol}\right]$ - the Avogadro constant;

$e=1.6e-19\left[C\right]$ - the electron charge;

$R=8310\left[\frac{J}{kmolK}\right]$ - the universal gas constant;

M_{G1} - mass of the G_1` gas molecule (O_2);

M_{G2} - mass of the G_2 gas molecule (N_2);

N_M - the surface density of the metal atoms (Ti);

T - the absolute temperature;

V - the volume of the vacuum chamber;

A_t - the target area;

A_c - the condensation area;

A_s - the substrate area;

A_g - the getter area;

S_{G1} - the pumping speed of the reactive gas G_1 (O_2);

S_{G2} - the pumping speed of the reactive gas G_2 (N_2);

η_M - sputtering efficiency of the metal (number of Ti atoms sputtered by an incident Ar^+ ion);

η_{MGi} - sputtering efficiency of the MG_i compound;

α_{tGi} - sticking coefficient of the G_i reactive gas atom to the surface of the target;

α_{gGi} - the gettering efficiency of the G_i reactive gas atom;

$\boldsymbol{\alpha}_s$ - the matrix of sticking coefficients on the surface of the substrate (the α_{sij} element of this matrix is the sticking coefficient of a particle type i to the area type j of the substrate- see *Table 1*);

$\mathbf{c}_s = \lfloor c_{MG1G2} \;\; c_{MG1a} \;\; c_{MG2a} \;\; c_{G1a} \;\; c_{G2a} \;\; c_{MG1p} \;\; c_{MG2p} \;\; c_M \rfloor$ - the elements of the $\mathbf{c}_s$ matrix represent the per unit area (the number of covered metal atoms) of the compounds corresponding to the subscripts.

Table 1: Interpretation of subscript i of the α_{sij} sticking coefficient

Row number (*i*)	1	2	3	4	5
Particle type	MG_1	MG_2	G_1	G_2	M

Table 2: Interpretation of subscript j of the α_{sij} sticking coefficient

Column number (j)	1	2	3	4	5	6	7	8
Substrate fraction	θ_{sMG1G2}	θ_{sMG1a}	θ_{sMG2a}	θ_{sG1a}	θ_{sG2a}	θ_{sMG1p}	θ_{sMG2p}	θ_{sM}

The block diagram shown in *Fig. 2* presents the main tasks performed by the model introduced in this paper.

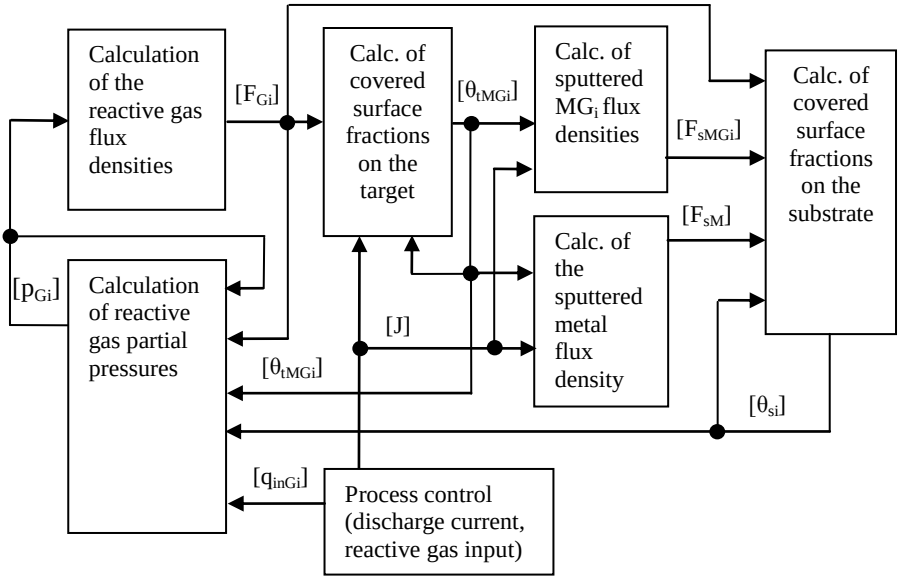

Figure 2: The main components of the two-reactive-gas sputtering model

2. Particle flux densities and the dynamics of the target coverage

The dynamics of the areas covered by different compounds on both the surface of the target and of the substrate is determined by the incident fluxes of particles, by the fluxes of particles removed via sputtering, and by the complex phenomena on these surfaces. These include the formation of chemical compounds, the crystal growth, and the segregation of some compounds on the grain boundary and the migration of particles.

On the surface of the target, the sputtering effect is produced by the incident flux of Ar^+ ions, while the fluxes of reactive gases result in chemical reactions,

i.e. target poisoning. It was assumed that metal atoms are sputtered from the surface of the target, and metal-gas compounds can be sputtered as well.

On the surface of the substrate, the sputtered particles produce incident fluxes, which participate in the process of thin film formation, along with the reactive gas fluxes resulting from the atmosphere created in the chamber.

The flux density J of the sputtering Ar^+ ions is calculated from the discharge current I, neglecting the secondary electron emission and the ion current of the reactive gases, according to (2).

$$J = \frac{I}{eA_t} \tag{2}$$

The atomic flux densities F_{Gia} of the reactive gases are calculated using results of the kinetic gas theory:

$$F_{Gia} = \frac{n_{GiGi} p_{Gi}}{\sqrt{2\pi k_B T M_{Gi}}}, \tag{3}$$

where $i = 1$ for oxygen and $i = 2$ for nitrogen.

On the substrate, the F_{sMGi} flux densities of the MG_i particles sputtered from the target are calculated using equation (4), assuming that these are uniformly distributed on the condensation area A_c.

$$F_{sMGi} = J * \eta_{MGi} * \theta_{tMGi} * \frac{A_t}{A_c} \tag{4}$$

The same approach stands for the F_{sM} flux density of the metal atoms sputtered from the target

$$F_{sM} = J * \eta_M * \theta_{tM} * \frac{A_t}{A_c} = J * \eta_M * \left(1 - \theta_{tMG1} - \theta_{tMG2}\right) * \frac{A_t}{A_c} \tag{5}$$

In the case of the target, it is assumed that the G_i gas atoms adhere only to the metallic fraction θ_{tM} of the surface, forming MGi compounds (poisoning the target).

The number of MGi molecules formed on the unit area of the target in a second (the rate of increment of the N_{tMGi} surface density of MGi molecules) is

$$\dot{N}^+_{tMGi} = \frac{n_{GiGi}}{n_{GiM}} F_{Gia} \alpha_{tGi} \left(1 - \sum_i \theta_{tMGi}\right). \tag{6}$$

The number of MGi molecules removed by sputtering from the unit area of the target (the rate of decrement of N_{tMGi}) is

$$\dot{N}^{-}_{tMGi} = A_s J \eta_{Gi} \theta_{tMGi} . \tag{7}$$

From the equilibrium of these processes, it results $\dot{N}^{+}_{tMGi} = \dot{N}^{-}_{tMGi}$, i.e.

$$\frac{n_{GiGi}}{n_{GiM}} F_{Gia} \alpha_{tGi} \left(1 - \sum_i \theta_{tMGi}\right) = J \eta_{MGi} \theta_{tMGi} . \tag{8}$$

Usually, $\eta_{MGi} << \eta_M$, i.e. the sputtering yield of the metal atoms is much bigger than that of the MG_i compounds.

The dynamics of the covered fraction θ_{tMGi} is described by the equations

$$\begin{cases} \dot{\theta}_{tMGi} = \dfrac{c_{MGip}}{N_M} \left[\dfrac{\alpha_{tGi} F_{Gia}}{n_{GiM}} \left(1 - \sum_i \theta_{tMGi}\right) - J \eta_{MGi} \theta_{tMGi} \right] \\ \theta_{tMGi} = \theta_{tMGi}(t_0) + \int_{t0}^{t} \dot{\theta}_{tMGi} dt \end{cases} . \tag{9}$$

3. Macroscopic modelling of the thin film deposition process

During the thin film deposition process, different fractions of the substrate surface are covered by atoms provided by the incident gas fluxes, by particles sputtered from the surface of the target and by different compounds formed during the crystal growth.

The aim is the deposition of MG_1G_2, and an optimal process control has to enassure atomic fluxes corresponding to the stoichiometry of this compound.

The model introduced in this paper considers the continuous emergence and disappearance of areas covered by $M, G_{1a}, G_{2a}, MG_{1a}, MG_{2a}, MG_{1p}, MG_{2p}$ and MG_1G_2, on each of the θ_{sM}, θ_{sMG1G2}, θ_{sMG1a}, θ_{sMG2a}, θ_{sMG1p}, θ_{sMG2p}, θ_{sG1a} and θ_{sG2a} fractions. Thus, the „local" processes taking place on each of the fractions contribute to the global evolution of the different fractions.

Let us use the following notation:

N^{j}_{si} - the surface density of the molecules of type *i*, emerging on the fraction θ_{sj} of the substrate, and thus contributing to the decrease of the fraction θ_{sj} and to the increase of the fraction θ_{si};

F_{si} - the flux density of the particles type *i* on the substrate (the same for every fraction);

R_{si}^{j} - the surface density of the molecules type *i*, emerging on the fraction θ_{sj} of the substrate by reaction between different incident particles or by reaction between incident particles and the molecules type *j* .

Thus, the time derivative of N_{si}^{j} results:

$$\dot{N}_{si}^{j} = F_{si} + \dot{R}_{si}^{j} \tag{10}$$

Due to these emerging molecules, the corresponding surface fraction is subject to transition to other types of fractions, characteriszed by the rate of growth $\dot{\theta}_{si}^{j}$ (i.e. time derivative of the surface fraction type *i* emerging on the surface type *j*):

$$\dot{\theta}_{si}^{j} = \begin{cases} \dfrac{c_i}{N_M} \dot{N}_{si}^{j} & if \quad i \neq j \\ 0 & if \quad i = j \end{cases} \tag{11}$$

The weighted sum (12) of the rates of growth of fractions type *i* yields the time derivative of θ_{si} .

$$\dot{\theta}_{si} = \sum_{j} \theta_{sj} \dot{\theta}_{si}^{j} \tag{12}$$

The evolution of the substrate surface fractions is found by integration (13), subject to the condition $\sum_{i} \theta_{si} = 1$.

$$\theta_{si} = \left\{ \theta_{si}(t_0) + \int_{t_0}^{t} \dot{\theta}_{si}\, dt \;\middle|\; \sum_{i} \theta_{si} = 1 \right\} \tag{13}$$

The number of particles available for the formation of different compounds depends on the incident fluxes and on the sticking coefficients.

It is assumed that the ternary compound MG_1G_2 is formed from incident atoms and "active" binary molecules present on the surface, or directly from incident atoms to the extent of their availability in quantities corresponding to the required stoichiometry $MG_{1x}G_{2y}$.

The particles in excess of this ternary stoichiometry form binary compounds, and the atoms still in excess of the possible binary stoichiometries are deposited as atomic layers.

In order to illustrate this reasoning, the relation (10) is detailed below (Eqs. (14), (15), (16), (17) and (18)) for the case of the fraction θ_{sMG1G2}. In the general notation N_{si}^{j} , we use the particular $j = MG_1G_2$ upper index. Thus, for

example, in the case of the molecules type MG_{1a} emerging on θ_{sMG1G2}, the lower index becomes $i = MG_{1a}$, and the surface density is denoted by N_{sMG1a}^{MG1G2}.

If $\alpha_{51}F_{sM} < \alpha_{31}\dfrac{F_{sG1a}}{x}$ and $\alpha_{51}F_{sM} < \alpha_{41}\dfrac{F_{sG2a}}{y}$

$$\begin{cases} \dot{N}_{sMG1G2}^{MG1G2} = \alpha_{51}F_{sM} \\ \dot{N}_{sG1a}^{MG1G2} = \alpha_{31}F_{sG1a} - x\alpha_{51}F_{sM} \\ \dot{N}_{sG2a}^{MG1G2} = \alpha_{41}F_{sG2a} - y\alpha_{51}F_{sM} \end{cases} \tag{14}$$

Else if $\alpha_{31}\dfrac{F_{sG1a}}{x} < \alpha_{51}F_{sM} < \alpha_{41}\dfrac{F_{sG2a}}{y}$

$$\begin{cases} \dot{N}_{sMG1G2}^{MG1G2} = \alpha_{31}\dfrac{F_{sG1a}}{x} \\ \dot{N}_{sMG2a}^{MG1G2} = \alpha_{51}F_{sM} - \alpha_{31}\dfrac{F_{sG1a}}{x} \\ \dot{N}_{sG2a}^{MG1G2} = \alpha_{41}F_{sG2a} - y\alpha_{51}F_{sM} \end{cases} \tag{15}$$

Else if $\alpha_{31}\dfrac{F_{sG1a}}{x} < \alpha_{41}\dfrac{F_{sG2a}}{y} < \alpha_{51}F_{sM}$

$$\begin{cases} \dot{N}_{sMG1G2}^{MG1G2} = \alpha_{31}\dfrac{F_{sG1a}}{x} \\ \dot{N}_{sMG2a}^{MG1G2} = \dfrac{1}{y}\alpha_{41}F_{sG2a} - \dfrac{1}{x}\alpha_{31}F_{sG1a} \\ \dot{N}_{sM}^{MG1G2} = \alpha_{51}F_{sM} - \alpha_{41}\dfrac{F_{sG2a}}{y} \end{cases} \tag{16}$$

Else if $\alpha_{41}\dfrac{F_{sG2a}}{y} < \alpha_{51}F_{sM} < \alpha_{31}\dfrac{F_{sG1a}}{x}$

$$\begin{cases} \dot{N}_{sMG1G2}^{MG1G2} = \alpha_{41} \dfrac{F_{sG2a}}{y} \\ \dot{N}_{sMG1a}^{MG1G2} = \alpha_{51} F_{sM} - \alpha_{41} \dfrac{F_{sG2a}}{y} \\ \dot{N}_{sG1a}^{MG1G2} = \alpha_{31} F_{sG1a} - x\alpha_{51} F_{sM} \end{cases} \tag{17}$$

Else if $\alpha_{41} \dfrac{F_{sG2a}}{y} < \alpha_{31} \dfrac{F_{sG1a}}{x} < \alpha_{51} F_{sM}$

$$\begin{cases} \dot{N}_{sMG1G2}^{MG1G2} = \alpha_{41} \dfrac{F_{sG2a}}{y} \\ \dot{N}_{sMG1a}^{MG1G2} = \dfrac{1}{x} \alpha_{31} F_{sG1a} - \dfrac{1}{y} \alpha_{41} F_{sG2a} \\ \dot{N}_{sM}^{MG1G2} = \alpha_{51} F_{sM} - \alpha_{31} \dfrac{F_{sG1a}}{x} \end{cases} \tag{18}$$

The steady state of the reactive gas quantities inside the sputtering chamber means relates to the equilibrium between the reactive gas admission, evacuation by pumping, release from and adsorption to different surfaces.

The dynamics of the partial pressures is related to the dynamics of the masses by means of the ideal gas law (19).

$$\begin{cases} \dot{p}_{Gi} = \dfrac{RT}{VN_A M_{Gi}} \dot{m}_{Gi} \\ p_{Gi} = p_{Gi}(t_0) + \int_{t0}^{t} \dot{p}_{Gi} dt \end{cases}, \tag{19}$$

where $\dot{m}_{Gi}$ is the time derivative of the mass of G_i (Eq. (24)).

The inlet mass flow of each gas is controlled and is denoted by q_{inGi}.

The pumping mass flow is proportional to the partial pressure of the gas

$$q_{pGi} = S_{Gi} p_{Gi} \tag{20}$$

The gases are adsorbed to the walls of the chamber (getter effect), deposited on the target (poisoning effect) and on the substrate (thin film growth).

The mass flow of the gettered gas is given by the equation

$$q_{gGi} = M_{Gi} \frac{F_{Gia}}{n_{GiGi}} \alpha_{gGi} A_g . \tag{21}$$

The mass flow of the gas adsorbed to the metallic fraction of the target surface is

$$q_{tGi} = M_{Gi} \frac{F_{Gia}}{n_{GiGi}} \alpha_{tGi} A_t \left(1 - \sum_j \theta_{tj} \right). \tag{22}$$

The mass flow of the gas adsorbed to different fractions of the substrate is

$$q_{sGi} = M_{Gi} \frac{F_{Gia}}{n_{GiGi}} A_s \sum_j \alpha_{sij} \theta_{sj} \,. \tag{23}$$

Thus,

$$\dot{m}_{Gi} = q_{inGi} - q_{pGi} - q_{gGi} - q_{tGi} - q_{sGi} \,. \tag{24}$$

Given the input quantities provided by the controller (i.e. the discharge current and the inlet gas flows), and the dynamic model of the sputtering process (Eqs. (9) to (13) and Eqs. (19) to (24) corresponding to the relationships shown in *Fig. 2*), it becomes possible to determine the evolution in time of the areas covered by different compounds both on the target and on the substrate. It is also possible to determine the evolution of the partial pressures.

4. Simulation results

In order to demonstrate that the model is able to describe the nonlinear behaviour of the sputtering system, we present simulation results based on hypothetical model parameters. *Figures 3* to *7* show the Matlab Simulink simulation results for the following parameters:

-The sputtered metal is Ti ($N_M = 1.62 \times 10^{19}\ 1/m^2$);

- G_1 is oxygen (O_2 , $n_{G1G1} = 2$, $M_{G1} = 53.12 \times 10^{-27}\ kg$);

- G_2 is nitrogen (N_2 , $n_{G2G2} = 2$, $M_{G2} = 46.48 \times 10^{-27}\ kg$);

-Binary compounds: TiO_2, TiN ($n_{MG1} = 1$, $n_{MG2} = 1$, $n_{G1M} = 2$, $n_{G2M} = 1$);

-Ternary compound: TiO_2N ($n_{MG1G2} = 1, x = 2, y = 1$);

- $T = 300\ K$;

- $V = 80 \times 10^{-3}\ m^3$;

- $A_t = 0.84 \times 10^{-2}\ m^2$, $A_c = 0.22\ m^2$, $A_s = 0.22\ m^2$, $A_g = 0.3\ m^2$;

- $S_{G1} = 92 \times 10^{-7}\ m \bullet s$, $S_{G2} = 72.9 \times 10^{-7}\ m \bullet s$;

- $\eta_M = 5$, $\eta_{MG1} = 0.2$, $\eta_{MG2} = 0.1$;

- $\alpha_{tG1} = 1$, $\alpha_{tG2} = 1$, $\alpha_{gG1} = 0$, $\alpha_{gG2} = 0$;

$$\begin{matrix} & \theta_{sMG_1G_2}, & \theta_{sMG_{1a}}, & \theta_{sMG_{2a}}, & \theta_{sG_{1a}}, & \theta_{sG_{2a}}, & \theta_{sMG_{1p}}, & \theta_{sMG_{2p}}, & \theta_{sM} & \\ \boldsymbol{\alpha_s} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0.2 & 0.2 & 0.3 & 0.2 & 0 & 0 & 0.4 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \end{bmatrix} & \begin{matrix} MG_1 \\ MG_2 \\ G_1 \\ G_2 \\ M \end{matrix} \end{matrix}$$

- $\mathbf{c_s} = [4\ 3\ 2\ 1\ 1\ 3\ 2\ 1]$.

The elements of the $\mathbf{c_s}$ matrix represent the per unit area (the number of covered metal atoms) of the compounds corresponding to the subscripts.

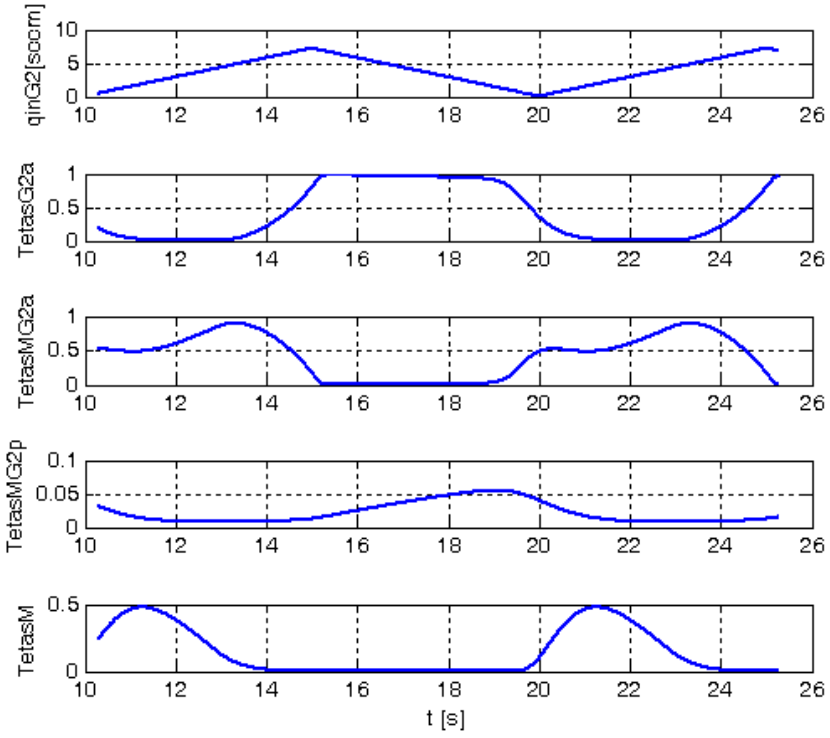

Figure 3: Evolution of the substrate fractions in the case of the admission of a single gas (G_2), with a periodic variation of its flow rate (uppermost diagram)

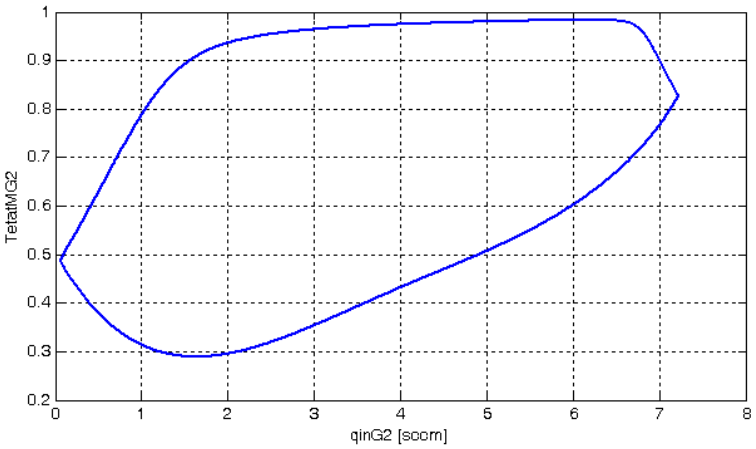

Figure 4: The dynamic hysteresis cycle of the target fraction covered by the MG_2 (TiN) compound, in the case of the admission of a single gas (G_2), with a flow rate variation period of $10s$

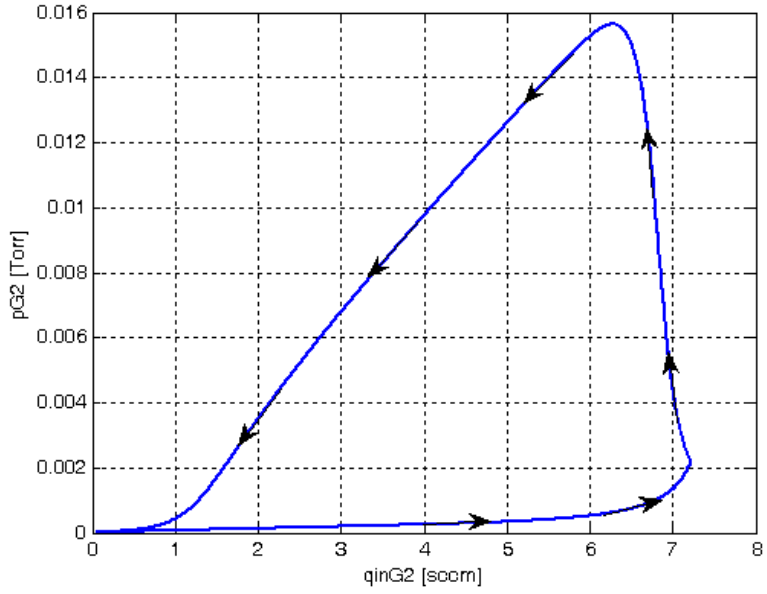

Figure 5: The dynamic hysteresis cycle of the G_2 gas pressure, in the case of the admission of a single gas (N_2), with a flow rate variation period of $10s$

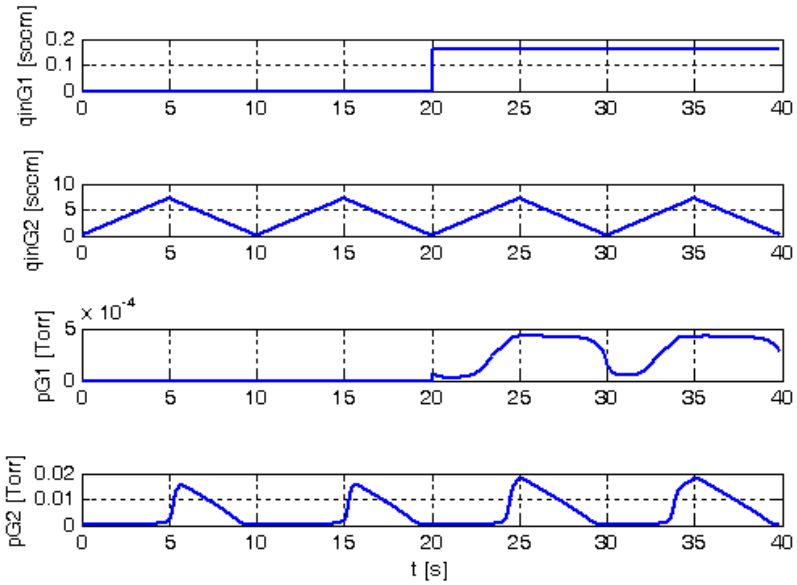

Figure 6: The admission of G_1 (O_2) has a strong influence on the partial pressure of G_2 (N_2)

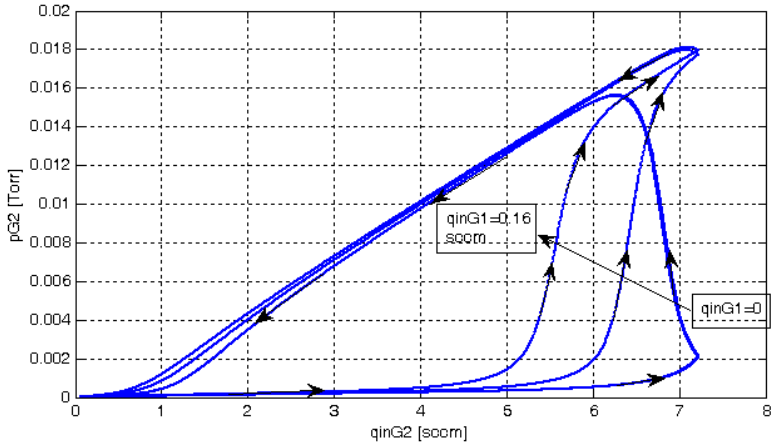

Figure 7: The influence of a small ($0.16\,[sccm]$) G_1 (O_2) gas flow step on the trajectory described in the $q_{inG2} - p_{G2}$ space (N_2 flow rate and partial pressure)

5. Conclusions

The dynamic model introduced in this paper has been developed to describe the essential macroscopic phenomena that accompany the DC magnetron sputtering and thin film deposition process in the presence of two or more reactive gases.

We defined state variables of the model, which are directly related to the stoichiometry of the compounds formed on the surface of the substrate. These state variables offer an insight into the satisfaction of the conditions required for the growth of crystals with the required stoichiometry.

In order to prepare the stoichiometry control of the deposition process, our next target is the identification of the model parameters and the practical validation of the model.

References

[1] Berg, S., Nyberg, T., Blom, H.-O., and Nender, C., "Computer modeling as a tool to predict deposition rate and film composition in the reactive sputtering process", *J. Vac. Sci. Technol. A 16(3).*, May/Jun 1998, pp. 1277-1285.

[2] Berg, S., Nender, C., „Modeling of Mass Transport and Gas Kinetics of the Reactive Sputtering Process", *Journal de Physique IV, Colloque C5, supplément au Journal de Physique II*, Volume 5, juin 1995, pp. 45-54.

[3] Berg, S., Katardijev, I. V., „Preferential sputtering effects in thin film processing", *J. Vac. Sci. Technol. A 17(4).*, Jul/Aug 1999, pp. 1916-1925.

[4] Jonsson, L. B., Nyberg, T., and Berg, S., "Target compound layer formation during reactive sputtering", *J. Vac. Sci. Technol. A 17(4)*, Jul/Aug 1999, pp. 1827-1831.

[5] Güttler, D., Abendroth, B., Grötzschel, R., Möller, W., and Depla, D., „Mechanisms of target poisoning during magnetron sputtering as investigated by real-time *in situ* analysis and collisional computer simulation", *Dept. Solid State Sciences, Univ. of Ghent, Annual Report IIM 2004, FZR-427*, pp. 34-37.

[6] Kubart, T., Polcar, T., Kappertz, O., Parreira, N., Nyberg, T., Berg, S., and Cavaleiro, A., „Modelling of magnetron sputtering of Tungsten Oxide with reactive gas pulsing", *Plasma Processes and Polymers, 2007, 4, WILEY-VCH Verlag GmbH & Co. KGaA, Weinheim*, pp. S522–S526, DOI: 10.1002/ppap.200731301.

[7] Görgy, K., „Cercetări privind dezvoltarea unor electrotehnologii pentru depunerea straturilor metalice subțiri", *Teză de doctorat, Universitatea Tehnică din Cluj-Napoca*, 2010.

[8] Christie D., J., „Making magnetron sputtering work: Modelling reactive sputtering dynamics, Part 1", *SVC Bulletin, Fall 2014*, pp. 24-27.

[9] Christie D., J., „Making magnetron sputtering work: Modelling reactive sputtering dynamics, Part 2", *SVC Bulletin, Spring 2015*, pp. 30-33.

[10] Christie D., J., „Making magnetron sputtering work: Modelling reactive sputtering dynamics, Part 3", *SVC Bulletin, Summer 2015*, pp. 38-41.

[11] Carlsson, P., Nender, C., Barankova, H., and Berg, S., „Reactive sputtering using two reactive gases, experiments and computer modeling", *J. Vac. Sci. Technol. A 11(4)*, Jul/Aug 1993, pp. 1534-1539.

[12] Sproul, W. D., Christie, D. J., Carter, D. C., Berg, S., and Nyberg, T., „Control of the reactive sputtering process using two reactive gases", *46'th Annual Technical Conference Proceedings, Society of Vacuum Coaters*, 2003, pp. 98-103.

[13] Carter, D. C., Sproul, W. D., and Christie, D. J., „Effective closed-loop control for reactive sputtering using two reactive gases", *47'th Annual Technical Conference Proceedings, Society of Vacuum Coaters*, 2004, pp. 37-43.

[14] Kubart, T., Trinh, D. H., Liljeholm, L., Hultman, L., Högberg, H., Nyberg, T., and Berg, S., „Experiments and modeling of dual reactive magnetron sputtering using two reactive gases", *J. Vac. Sci. Technol. A 26(4)*, Jul/Aug 2008, pp. 565-570.

[15] Papp, S., "Optimizarea controlului automat al pulverizării catodice în procesul de obţinere a straturilor subţiri", *Teză de doctorat, Universitatea Transilvania din Braşov*, 2012.

Acta Universitatis Sapientiae

The scientific journal of Sapientia University publishes original papers and surveys
in several areas of sciences written in English.
Information about each series can be found at
http://www.acta.sapientia.ro.

Acta Universitatis Sapientiae
Electrical and Mechanical Engineering

Sapientia University

Scientia Publishing House

ISSN 2065-5916
http://www.acta.sapientia.ro

Information for authors

Acta Universitatis Sapientiae, Electrical and Mechanical Engineering publishes only original papers and surveys in various fields of Electrical and Mechanical Engineering. All papers are peer-reviewed.

Papers published in current and previous volumes can be found in Portable Document Format (PDF) form at the address: `http://www.acta.sapientia.ro`.

The submitted papers must not be considered to be published by other journals. The corresponding author is responsible to obtain the permission for publication of co-authors and of the authorities of institutes, if needed. The Editorial Board is disclaiming any responsibility.

The paper must be submitted both in MSWord document and PDF format. The submitted PDF document is used as reference. The camera-ready journal is prepared in PDF format by the editors. In order to reduce subsequent changes of aspect to minimum, an accurate formatting is required. The paper should be prepared on A4 paper (210×297 mm) and it must contain an abstract of 200-250 words.

The language of the journal is English. The paper must be prepared in single-column format, not exceeding 12 pages including figures, tables and references.

The template file from http://www.acta.sapientia.ro/acta-emeng/emeng-main.htm may be used for details.

Submission must be made only by e-mail (`acta-emeng@acta.sapientia.ro`).

One issue is offered to each author free of charge. No reprints are available.

Contact address and subscription:
Acta Universitatis Sapientiae, Electrical and Mechanical Engineering
RO 400112 Cluj-Napoca
Str. Matei Corvin nr. 4.
E-mail: acta-emeng@acta.sapientia.ro

Printed by Digital Color Company
www.digitalcolorcompany.ro