A Distributed Power Consumption Measurement System and its Applications

Tamás Kovácsházy, Gábor Fodor
Department of Measurement and Information Systems
Budapest University of Technology and Economics
Budapest, Hungary
khazy@mit.bme.hu, fodorg@mit.bme.hu

Csaba Bernát Seres
Biomedical Engineering Knowledge Centre
Budapest University of Technology and Economics
Budapest, Hungary
seres@emt.bme.hu

Abstract — **The paper presents a distributed power consumption measurement system using ZigBee based wireless point of load power meters. The software architecture of the system is based the D-Bus inter-process communication solution. It is shown in the paper that our D-Bus based solution is better from the point of view of reliability and programming language support than the widely used JAVA based OSGi framework. The system consists data analyses components also. Our example component applies hybrid model based detection algorithms to identify the operating state of power consuming devices We also introduce our initial results from the operation of the system, for example wireless communication problems and the consequences of them are presented. The primary application area of the developed solution is activity detection in ambient assisted living solutions and energy aware IT system management.**

Keywords - smart grid, point of load power meters, wireless sensor network, inter-process communication, D-Bus, activity detection, ambient assisted living

I. Introduction

The smart grid, i.e., a ubiquitous electrical power measurement and optimization system, is a reality today. It provides power usage measurement and analysis capabilities at the user's premises, or in industrial environments to certain devices of the user with higher consumption. However, in homes and in some other industries, like IT and facility management, these solutions do not provide the necessary details from the user's perspective for the following reasons:

1. Most cases users are interested in detailed knowledge about the power consumption of devices. However, it is a complex, error prone, and in some applications practically impossible process to identify the operation of certain devices based on the power meter installed by the utility company. The primary cause of this problem is that these meters observer the aggregated power usage of a large number of devices. In addition, these meters are designed for and operated on long sampling intervals (1 to 15 minutes) compared to the dynamics of power consumption.
2. Utility companies may not allow real-time user access to the data provided by these smart meters [1]. There is an open legal debate if utility companies are required to allow user access to smart meter collected data real-time (and they are also required to build and operate the infrastructure for that) or they are only required to communicate just billing information to the users.

Point of load power meters offer an alternative solutions for such applications. They can be plugged into the wall socket, or connected using shunts or current transformers to high power devices, and they can measure power consumption and other electrical parameters of the connected devices with very low (down to the 1 second) sampling interval. They may be built into devices like home appliances or computers in the coming years with additional functionalities such us operational environment sensing (temperature, humidity, etc.), operational status reporting (e.g. operational state, wear levels, malfunctions) and remote operation (set points). These point of load power meters communicate using wireless sensor network technology such as ZigBee or 6LoWPAN, typically. In addition, they may offer remote switch on/off functionalities on command (initiated from the sensor network) and on time (scheduled on/off).

Unfortunately, point of load power meters are only sensors components of the whole distributed power consumption measurement system, and their capabilities cannot be utilized without integrating them into a complex distributed power measurement system, i.e., the other system components and functionalities of the system must be developed. This paper details the complete system architecture, functionalities and operation of our distributed power measurement system, and shows our initial results and practical experiences obtained by operating the system for 4 months.

The primary application areas of the proposed solution are human activity detection and energy usage optimization in homes and for IT systems. In the activity detection application scenario we detect human activities based on power consumption of home appliances and other devices in the home. For example, if a user opens the fridge door around noon, and then a little bit later he or she uses the microwave oven or the electric stove we can assume that he or she consumed his or her lunch. In the energy usage optimization application scenario the energy consumption of devices are measured, and based on the measured data

978-1-61284-361-2/11/$26.00 ©2011 IEEE

improvements are made on the system to decrease the energy consumption.

The organization of the paper is the following. Section II introduces the reader to the properties of point of load power meters. The system architecture is presented in Section III. Section IV details our approach to data analyses. The results are listed in Section V.

II. Properties of Point of Load Power Meters

Low prices (20-40 EUR) point of load power meters are available on the market on for a long time with LCD display to read the measurement results. However, only a very limited number of manufacturers offer such devices with ZigBee or other (e.g. 6LoWPAN or Bluetooth) wireless or wired communication protocols for home and office use. In addition, current prices are too high (approximately 100-150 EUR) for wide scale user acceptance. We use PloggZGB electricity meters (*www.**plogg**international.com/*) in our system from Energy Optimizers Limited, UK. They are typical point of load power meters with the following functions:

- Electricity consumption measurement on one phase using a plug-in type device (both consumed and generated energy are measured),
- No licensed technician required for installation,
- Momentary real and reactive power consumption, phase angle, frequency, RMS voltage and RMS current measurements are also provided,
- The devices offer on device storage for samples (in non-volatile memory) and can be set up for 1 second or longer sampling interval,
- Remote switch on/off function with or without timer utilizing a built in relay,
- CEE 7/7 (Continental Europe) or BS 1363 (British) socket plug-in device (available with other socket also),
- High current electricity meter option (no switch on/off function, licensed technician may be required for installation) using clip on current transformer,
- ZigBee proprietary application layer communication protocol or ZigBee Smart Energy Profile compatibility,
- All PloggZGB devices act as ZigBee routers extending the range of ZigBee network,
- They are powered from the measured socket, so no battery change is required.

Other devices may differ in functionality or communication interface, so any distributed power management system must be architected in a layered, flexible way that makes possible integration of point of load power meters from other manufacturers.

Even if the distributed power consumption measurement system uses point of load power meters, they measure the aggregated power consumption typically, i.e., multiple devices are connected to them due to previously mentioned financial reasons. The operation of the connected devices is independent most cases; however, human interaction, a higher level supervisory controller, or interrelated higher level operation (in case of WEB and database server, for example) may cause some coupling between them. Most devices are also compound devices, they consist of multiple power consuming subsystems, for example, a refrigerator have a compressor for cooling and a internal light switched on and off by opening or closing the door of the refrigerator. Therefore, even in this setup the system must consist intelligent components that are capable of identifying the operation of subsystems based on aggregated power consumption, and in some cases using additional not power related information. However, this task is less complicated than using the data available from the power meters of the utility company, because more detailed information is available.

III. System Architecture for Distributed Power Consumption Measurement

Distributed power consumption measurement systems consists one or more sensor networks to transfer information from point of load power meters to data collectors, and in addition, a data collection, processing and presentation framework. In essence, the system architecture is a generic distributed measurement and control system utilizing sensor networks.

A. Hardware and network architecture

The hardware and network architecture of our system is depicted on Figure 1. The architecture can be operated in one location (central site) with multiple sensor networks and it may have also remote sites. Remote sites include minimal infrastructure, and they connect to the central site through the Internet. Security is a paramount issue in such environments; therefore, our system uses Virtual Private Networking (VPN) technology and firewalls. The central and remote sites are guarded by firewalls, and the remote sites connect to the central site using the OpenVPN (***openvpn**.net/*) open source VPN solution, which provide strong security. Users access the system remotely using a WEB based interface supporting any devices including tables, smart phones, and notebook and desktop computers.

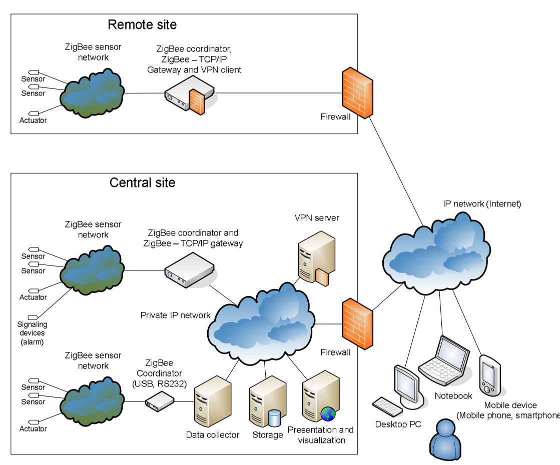

Figure 1. Hardware and network architecture.

ZigBee sensor networks connect to computers executing data collection tasks through USB (directly to the data collector computer) or by TCP connections (indirectly) through ZigBee-TCP/IP gateways. Gateways utilized on the central sites are simple serial-TCP/IP gateways, but gateways employed on the remote sites also implement the VPN and firewall functionality; therefore, they are complete embedded x86 or ARM architecture computers running the GNU Linux operating system. In experimental setups it is reasonable to use standard x86 PCs for the ease of development; however, for price, power consumption, and size reasons it is advised to use embedded devices based on the ARM architecture in the real applications.

All the other server components, including the data collector, i.e. the classic Information Technology (IT) components, are standard Linux PCs or servers running GNU Linux. Scalability and reliability issues of the IT infrastructure can be solved by regular IT solutions such as load balancers, failover techniques, and redundancy offered by the hardware, the operating system, and software components.

B. *Software architecture*

From point of view of software only the implementation of data collector and data analyses (processing) functionalities poses real architectural issues because similar data storage, presentation functionalities are widely implemented functionalities in some open source IT system management software packages CACTI (*www.**cacti**.net/)* and Nagios (*www.**nagios**.org/)*. These architectural issues of data collection and analyses arise because the following requirements can be formulated:

- The system must be flexible in supporting multiple sensors networks connecting from local and remote sites,
- New sensor networks and sensors should be allowed to connect and disconnect in run time, without stopping the system,
- Sensor network technology is advancing fast; therefore, the system should be adaptable to new technologies,
- Data analysis functionalities should be able to use any data available in the system, including outputs of other analyses functionalities,
- Analyses functionalities should be able to started up or stopped run-time,
- The system must be able to use the multiple execution units of modern processors,
- The system should be ready to use the cloud computing infrastructure if local resources are not available,
- The system should be able to use multiple independent storage solutions, including database servers, files using the TAB separated format, etc.,
- The components of the system should not be required to be written in one programming language, i.e., the employed framework integrating the components should be programming language independent to allow specialists writing the components in their language of choice,
- The architecture must be able to separate the components/functionalities strongly, i.e., it should be easy to identify misbehaving components, and a misbehaving component should not be allowed to take control of the system.

The JAVA based OSGi framework [2] is extensively used in research projects to provide most of these functionalities except the last two (language independence and component separation). However, based on our experience from previous similar projects [3] we decided not to use OSGi in our system because:

- Some of the components had been written before the project started in other languages than JAVA (e.g. PHP, python). These components were found to be solid, good quality software, i.e., rewriting them in JAVA would consume time and introduce new errors only.
- The components of the system are to be written by specialist, who have their specific language of domain. For example, analyses system components may be created in Matlab and tested using Matlab, than in a later development phase they may be translated to C automatically using a Matlab to C compiler for fast execution. Some other languages may be used in components running decision making and machine learning algorithms, which research fields have their own specific languages. Rewriting all of these components in JAVA requires lot of resources, and after that these specialists cannot take part in the software development process anymore, they can only test the software as algorithms specialists.
- OSGi and Java run components in a JAVA virtual machine, i.e., all components (bundles in OSGi terminology) share the same memory area of the JAVA virtual machine process in Linux. In case of a misbehaving bundle it means that it is practically impossible to defend other bundles from the effect of the misbehaving bundle, in other words, the lowest quality bundle determines the quality of the entire system. In addition, these bundles are only partially visible from the operating system for monitoring purposes [4].

Therefore we needed to find a high-level Inter Process Communication (IPC) solution on Linux which is language independent and also high performance. If this IPC solution is found, then it is applied to connect the components of the system that are implemented and running as separate Linux processes (programs) managed by the widely known standard Linux tools. This solution has some other advantages also:

- The standard Linux package management systems (such as apt, yum) can be used to maintain software upgrades in larger installations by using our own package repository. These package management

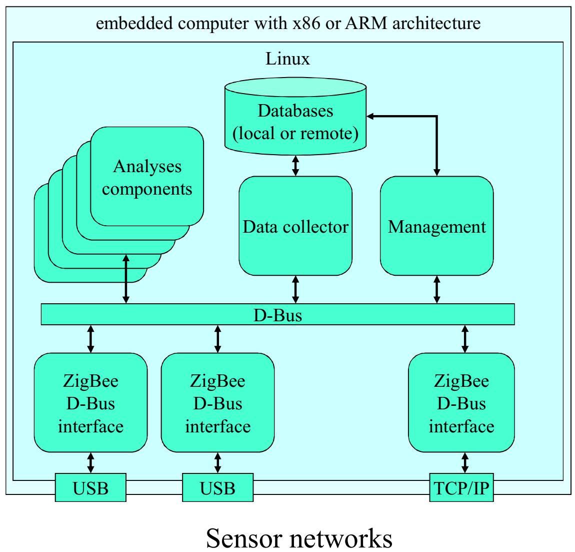

Figure 2. Software architecture of the system.

systems can also solve the issue of software package dependency.

- Standard Linux startup and shutdown script framework (init scripts) can be used to start and stop the components.

Finally, after experimenting with several solutions, the Desktop Bus (D-Bus) is selected as the IPC framework for our system. D-Bus is heavily used in Linux both in desktop and system applications, and it is also ported to other operating systems including Windows. D-Bus is a message bus, implementing the publish-subscribe architecture. There exists wrapper libraries or in other word bindings, for nearly all programming language including C, C++, JAVA, Python, PHP, etc., so it provides the required language independence. Furthermore, it is also possible to connect D-Buses present on different computers making distributed computing possible.

Figure 2. shows the system architecture constructed for the distributed power measurement system. It may consists multiple ZigBee D-Bus interfaces, analyses components, data collector components saving collected data and the results of the analyses to databases, and a management component capable of configuring the components of the system for autonomous operation. The detailed specification of the communication on D-Bus is out of the scope of this paper.

IV. Analyses Components

As previously mentioned in the paper, even if the distributed power consumption measurement system uses point of load power meters, they measure typically the aggregated power consumption of multiple devices. In addition, most of the devices are also compound devices, i.e., most of them consist of multiple power consuming subsystems. Therefore, the system must have intelligent components that are capable of identifying the operation of subsystems based on aggregated power consumption, and in some cases using additional not power related information. This is one of the essential tasks of the system that may be solved by analyses components. For this process we have to find a sufficient and generic mathematical model to describe the operation (power consumption) of devices and device subsystems.

A. Hybrid system model

The power consumption of the device in a certain state can be modeled using a linear or non-linear dynamical system, and the switch between state can be describe by some mapping between the states (transition). This system structure is clearly calls for a hybrid system model [5]. Based on this view, we have started developing a hybrid system modeling and identification scheme (see Figure 3) for determining the operation state and parameters of power consuming devices based on algorithms presented in [6]. The developed preliminary solution has been tested with refrigerators. It is capable of identifying the switched on/off state of the compressor in the refrigerator; however, it can only identify the event of opening or closing the door if the sampling time is significantly shorter than the typical time the door is kept opened. This is due to the fact that if the door is kept opened for shorter time than the sampling interval it is more likely that the power consumption caused by the internal lamp in the refrigerator is filtered out by the averaging nature of energy consumption measurement. Another phenomenon is also observed in our system, the compressor switches on or off quite frequently when the door is opened or closed, i.e., there is some correlation between these theoretically independent subsystems. The reason behind this behavior is that the mechanical temperature switch (bimetal) used to switch on and off the compressor can switch earlier if mechanical stress (vibration due to the opening/closing the door) introduced to the system.

B. Presentation of data for human analyses

Currently the data collected by the system is available through a WEB based user interface. Users can specify the

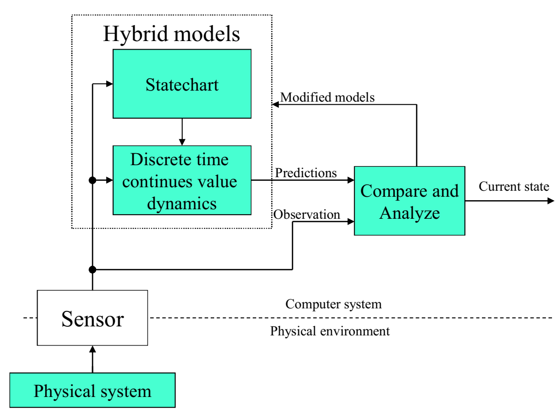

Figure 3. Hybrid identification scheme to determine the operational state of a hibrid system.

list of point of load power meters and a time range. The system generates figures using the Gnuplot (*www.**gnuplot**.info/*) graphing utility run-time using this specification and embeds them in dynamically generated WEB pages. Gnuplot is used because it can generate detailed high resolution images with a style very familiar to technical people. The server software component is coded in PHP, and runs on an Apache WEB server.

In addition, the system is capable exporting the data through the WEB interface. In this case users or programs invoking the WEB interface specify the sensor, the time range, and the format. Then the data is returned as XML or TAB separated document according to the format specification. This feature is very practical for data analyses algorithm development, because developers can get historical and real-time data from a central data repository in the live test environment. For example, Matlab and Excel are successfully programmed to analyze the data collected in our system.

Information about wireless sensor network topology and sensor node availability are also collected in our system. This data is visualized using the Graphviz (*www.**graphviz**.org/)* software package.

V. Results

A. The status of implementation

Currently, from we have operational ZigBee D-Bus interface software, analyses components, and data collector components saving data into TAB separated files and into MySQL databases. The system is configured manually; the management component is in the early implementation phase. The implemented data analyses components are preliminary, but they show the applicability of our D-Bus based flexible publish-subscribe architecture for sensor data processing. However, further algorithm development and research are needed in this field.

The data presentation components provide the required functionality for expert. For example, Figure 4 shows the

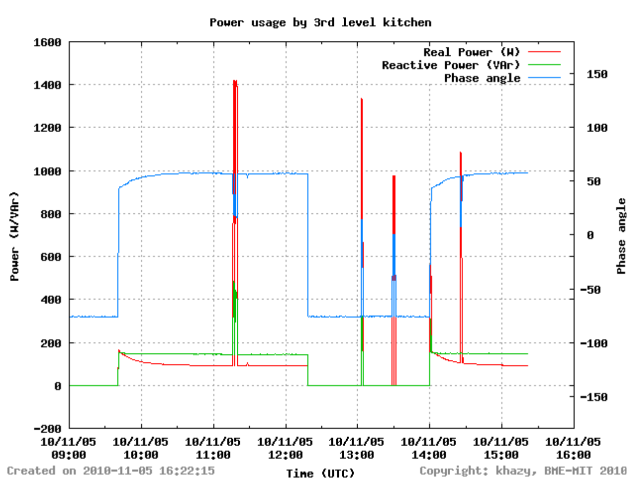

Figure 4. Power consumption of an outlet in a kitchen, which powers a refrigerator, a water heater, and a microwave oven

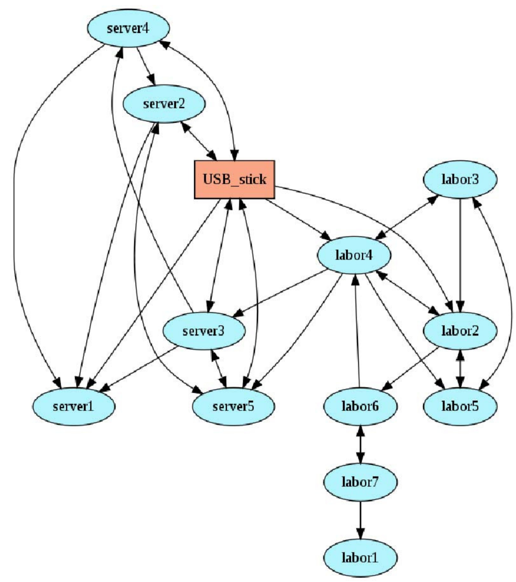

Figure 5. A typical network topology during normal operation

power consumption measured in a wall outlet powering a refrigerator, a microwave oven, and a water heater for 8 hours. Figure 5 and 6 present the wireless network topology in various operational states of the system.

The system has been running for more than 4 months without major software problems. We have identified the following system level issues:

- Reliability of wireless sensor networks is questionable, sensor locations (antenna placement) and other disturbances (WI-FI and Bluetooth) may have dramatic effects on the operation of the system,
- Data processing and presentation algorithms must take into account the non-continuous availability of data, a quite common phenomenon in wireless senor networks.

B. Wirelss Sensor Network reliability

We have observed several cases when a part of our wireless sensor network was not able to deliver data and lost full connectivity. All of these issues were later tracked back to clustering issues. The point of load power meters are scattered around in our building on two floors. Five meters are located in the server room and measure power consumption of servers and an uninterruptable power supply (UPS) powering network equipment. These devices have the name serverX, where X is a number between 1 and 5. One of these servers run the data collector software also; therefore, the ZigBee coordinator (USB stick) is also connected to it. Seven other power meters (laborX) are located in a laboratory and office environment farther away (20 meters) on the same floor (3^{rd}) and a floor over this floor (4^{th}). These two parts form two clusters of wireless nodes which have limited connectivity, which introduces clustering problems.

In this setup the sensor labor4 plays a key role, because as the normal operational topology shows on Figure 4, it connects the two installation areas as a ZigBee router. In

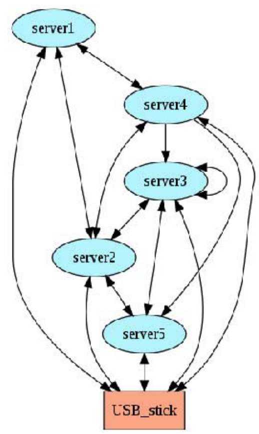

Figure 6. A typical network topology in a clusted state of operation

addition, labor2 has weak access to the other part of the system also. The operation of the system was very sensitive to the location of labor4. If labor4 was connected to the wall socket the system was partially operational. In working hours only the serverX nodes were accessible, see Figure 5 for an example topology in this case. Full operation was observed sometimes on weekends and nights. In these cases laborX nodes connected through labor2 or labor4. This time dependant clustering behavior was caused by heavy use of WI-FI by staff and students in working hours, as we were able to reproduce this phenomenon during a weekend by generating WI-FI traffic.

As a solution, the sensor labor4 was put on an extension cord allowing more flexible sensor (and antenna) placement. By moving labor4 50 centimeters away from its original location in the wall socket and rotating it with 90° vertically we have got 99.9% uptime independently of the WI-FI traffic. Therefore we can state that ZigBee wireless sensor networks (and likely all the others using the 2.4 GHz band) are sensitive to sensor location if there are gaps in ZigBee router placement, and they are sensitive to WI-FI traffic.

C. *Data processing issues*

The availability of sensor data is lower and sampling jitter is higher in wireless sensor networks than in classic wired sensor networks. In addition, point of load power meters send the consumed/generated energy and momentary power values augmented with a sensor timestamp, and also data collector timestamps can be attached to the data. Therefore, the system must be prepared for unavailable data and non-equidistant sampling. Therefore in our current solution we use the following rules to allow more detailed analyses of data:

- All data is stored with both sensor and data collector timestamp, in other words the algorithms may use both timestamps for operation.
- If a sensor do not answer a request, all data coming from the sensor is set to NaN (Not a Number) and these NaN values are written to the database, i.e., unsuccessful access to data is also noted.
- Both momentary and average power values (computed from consumed energy between sampling events) are available in the database for analyses.
- Sampling time must be set to short enough to detect human and other activities, but the utilization of the sensor network is directly proportional to sampling time and to the number of sensors.

Conclusions

We have been collecting data using commercial of the shelf (COTS) plug-in type point of load power meters for more than 4 months. The data collection and analyzes framework has been developed in house, and its operation and the used software components are detailed in this paper. Our software architecture is based on the D-Bus IPC framework, which is better solution based on our practical experience than the widely used OSGi based solutions. In our test environment devices measure the real and reactive power consumption of appliances, such us refrigerators, water heaters, and microwave ovens, and IT infrastructure components such as servers, client computers, printers, uninterruptible power supplies, etc. Based on the collected data we have started to build prototype applications to demonstrate the usefulness of such systems and to evaluate power consumption analyses and activity detection algorithms that may be used to reduce power consumption of IT systems and help the daily life of people.

Acknowledgment

This work was funded by the CCE - Connected Care for Elderly persons suffering from dementia project (aal-2008-1-101) under the AAL - Ambient Assisted Living Joint Programme of the EU.

This work is connected to the scientific program of the "Development of quality-oriented and cooperative R+D+I strategy and functional model at BME" project. This project is supported by the New Hungary Development Plan (Project ID: TÁMOP-4.2.1/B-09/1/KMR-2010-0002).

References

[1] Department of Energy, "Data Access and Privacy Issues Related to Smart Grid Technologies", USA, October 2010.

[2] OSGi R4.1 2007. OSGi Service Platform, Core Specification, Release 4, Version 4.1 OSG Alliance - 2007 – May

[3] T. Kovácsházy , Cs. Tóth, "System Management Approach for Assisted Independent Living Applications," INTERNATIONAL CARPATHIAN CONTROL CONFERENCE. Zakopane, Poland, May 2009, pp. 513-516.

[4] Y. Royon, S. Frenot, "Multiservice home gateways: Business model, execution environment, management infrastructure," IEEE Communications Magazine, Vol. 45., Num 10., pp. 122-128.

[5] Sun, Z. and Ge, S.S., "Switched linear systems: Control and design," Springer Verlag, 2005.

[6] S. Paoletti, A.L. Juloski, G. Ferrari-Trecate, G. R. Vidal, "Identification of hybrid systems: A tutorial," European Journal of Control, Vol. 13, Num. 2-3, pp. 242-260,