

# Developing a RESTful Communication Protocol and an Energy Optimization Algorithm for a Connected Sustainable Home

Sotirios D. Kotsopoulos  
Federico Casalegno  
Massachusetts Institute of Technology  
Mobile Experience Laboratory  
(001) 617 324 2269  
skots@mit.edu  
casalegno@mit.edu

Adria Recasens  
Wesley Graybill  
Massachusetts Institute of Technology  
Electrical Eng. & Computer Sc.  
(001) 857 284 9598  
recasens@mit.edu  
wdgraybill@gmail.com

## ABSTRACT

Improving the energy efficiency of residential buildings can lead in reducing the energy consumption of cities. It is presented a prototype house that uses real time connectivity to enable effective energy management through the deployment of an optimization algorithm. Following the principles of the Web of Things, the devices of the house are mapped into a RESTful structure where the HTTP protocol is used for communication, and a risk-sensitive algorithm p-Sulu, based on Iterative Risk Allocation, is used for long-term energy optimization. The RESTful protocol makes the system compatible and scalable, able to expand at a neighborhood or a community scale.

## Categories and Subject Descriptors

C.2.2 [Computer Communication Networks]: Network Protocols – protocol architecture, protocol verification.

## General Terms

Algorithms, Design, Experimentation.

## Keywords

Web of Things, Home Automation, Optimization.

## 1. INTRODUCTION

The number of programmable devices aiming to improve the efficiency and enhance the quality of life in cities has been drastically increasing during the past ten years. For example, by deploying networks of sensors, gather and monitor data on air quality and noise levels, and making it available to residents and other parties, appropriate changes in behavior and policy can be initiated. The prospective of deploying wide-ranging networks of devices communicating through the Internet, jointly with the flexibility of the web communication protocols, had led to the growth of the Internet of Things (IoT) [1].

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

Urb-IoT 2014, October 27-28, Rome, Italy  
Copyright © 2014 ICST 978-1-63190-037-2  
DOI 10.4108/icst.urb-iot.2014.257303

It is envisaged that the evolution of these networks will lead to the development of connected homes, cars, machines and consumer devices along with the infrastructures that go with them, unleashing a wave of new possibilities for data gathering, predictive analytics, and automation. Hence, a world of devices is evolving, where every individual device has Internet access. The management and control of this world poses a critical technical challenge, namely: *How to interconnect a variety of diverse individual devices into a flexible and effective network?* Numerous proposals have been made to solve this problem. Some IoT systems use explicit protocols for each device and an exchange platform enabling device-to-device communication based on the best protocol within a given list. This mode allows the communication to be optimized depending on the properties of the communication entities. But, the lack of a common protocol makes the logistics of device-to-device communication harder and increases complexity. An alternative approach is the Web of Things (WoT) paradigm, where devices use HTTP as communication protocol [2]. It is presented an approach for managing a network of home devices aiming to deliver home automation and energy optimization at a residential and urban scale. It is presented a web-based protocol built upon a Representational State Transfer (REST) structure [3] for a prototype Connected Sustainable Home (CSH), which was just completed in Trento, N. Italy (Fig. 1). Following the principles of the Web of Things (WoT) paradigm, the devices servicing the CSH prototype are mapped into a web-based structure that uses an HTTP protocol for communication. A model-based control (p-Sulu) based on the Iterative Risk Allocation algorithm [4] is built upon this protocol to manage the house devices and to optimize their performance. Motivation for this project was that residential buildings consumed in 2012 nearly 40% of total energy usage in the U.S., or about 40 quadrillion Btu of energy, with heating and cooling being accounted for the 48% of this energy consumption [5]. These figures reveal that even small improvements in the energy efficiency of the residential sector can impact significantly the reduction of energy consumption in cities. By establishing connectivity among the house devices the presented approach enables autonomous energy management, improves comfort, and contributes to the sustainability of energy supply. A precise assessment of the proposed apparatus will be subject of future research, after quantifiable data from the prototype become available. This presentation considers the anticipated performance of the system, resulting from simulation. Simulations show energy savings as high as 42.8% in the winter, 15.3% in the spring, 16.8% in the summer. and 4.4% in autumn. After a brief reference to related work, the principles governing the RESTful protocol are explained and the energy optimization apparatus p-Sulu is discussed. The paper ends with a summary of the contributions, and with plans for future work.



**Figure 1. The Connected Sustainable Home in Trento.**

## 2. BACKGROUND

The concept of connected sustainability is based on the hypothesis that energy efficiency can be maximized if it is managed collectively, and in real time. Houses connected into a network can better achieve sustainability. A cognitive control system in every home can enable real time exchange of information among houses, to balance energy loads and to assure timely waste and resource management. Connected homes can have at their disposal a wide variety of energy production and storage capabilities (solar cells, wind turbines, cogeneration plants etc..) while symmetric connections can enable dynamic sharing of these resources. We addressed these challenges in three ways: 1) by proposing innovative physical architecture for homes and communities; 2) by implementing autonomous management of energy resources that is robust to uncertainty and user comfort levels; and 3) by enabling intelligent real time interaction between devices and between devices and occupants.

First, the physical architecture of the CSH integrates a variety of passive (high thermal mass), and electroactive (programmable) materials, such as wood, electrochromic glass etc. For example, the adjustment of the tinting levels of a programmable electrochromic façade, oriented towards south, enables the management of solar light and heat at the house interior based on the seasonal conditions.

Second, the problem of instability of energy supply and demand – which is exacerbated by the use of renewable resources – is addressed through the cognitive controller p-Sulu that employs resources as efficiently as possible. Based on the Iterative Risk Allocation algorithm [6] p-Sulu ensures that the risk of failing to meet demand is always kept within user specified bounds. We are also currently studying a distributed, market-based approach to risk allocation using the symmetric connectivity of the community as an insurance mechanism, to allocate and distribute risk in the network.

Third, connectivity between house devices and control, and between devices and residents, is used to support the decision-making layer, and to enable the execution of performance plans. Following the principles of the WoT paradigm, a RESTful protocol is used, and an HTTP server is employed to place the devices and the information provided by them into a path structure, where they become available to the control system and to residents of the house.

Ultimate vision is to enable accurate prediction of use patterns and cater to the needs of the occupants without explicit user input. Furthermore, in the future homes could act as personal trainers, actively shaping their user's behavior towards sustainable practices. For example, intelligent sockets would detect the appliances being used, and suggest alternative times to perform the associated activity at a lower cost. In the prototype CSH the individual devices are organized into a network and managed by a risk-sensitive optimization algorithm (p-Sulu). An alternative approach to IoT, the WoT paradigm [2] is adopted and a RESTful protocol [3] is implemented. The RESTful apparatus uses HTTP protocol for device-to-device communication, it simplifies the system, and it

enables the development of Web Mashups for Embedded Devices [7]. Web Mashups are used to collect information from web locations. In the CSH they are used to collect information from the devices of the network. The novelty of the CSH is that while a Mashup typically involves a simple service build upon the communication layer, the applications developed for the CSH are multifaceted. The RESTful protocol enables interaction between the optimization algorithm and the house devices: the sensors, the electrochromic façade, the artificial lighting and the HVAC system. The distinction between optimization and actuation apparatus permits the house devices to run in automatic or manual mode, and it creates a firewall in the case that the automatic mode fails.

Most of the existing protocols do not provide compatibility with devices without adding anything to these devices. The adopted solution, based on the WoT paradigm, enables HTTP communication. Since the HTTP protocol is part of all browsers, this enables the users to browse the devices in the network through a common browser. Hence, compatibility was a strong reason for selecting to follow the WoT paradigm. On the downside, the system can be slower and less efficient than other existing solutions.

In the context of home automation, research relevant to the WoT paradigm was advanced in [8] and [9]. Specifically, [8] proposes a structure for the implementation of physical mash ups in homes, and [9] introduces a new framework for the use of the WoT in home automation. Based on these two paradigms, the presented prototype is probably the first fully realized application of the WoT paradigm in home automation. Another relevant paradigm of an integrated approach is the TRON (The Real-time Operating System Nucleus) project [10] that aims to provide a framework architecture in which a wide variety of devices (or intelligent objects as referred to in the TRON Projects) can be easily integrated by developing a distributed network of TRON based designs. However, the communication specification of the TRON does not rely on a widely used protocol like HTTP. While the TRON aims to build a full stack of specifications and design guidelines to determine the hosted systems, the RESTful protocol of the CSH simplifies the communication layer without entering in other levels of computation.

A risk-sensitive optimization algorithm, p-Sulu, manages the network of the CSH prototype. The application of AI methods in building control has been pursued by computational sustainability. For example, [11] employs the stochastic model-predictive control (SMPC) approach to reduce building energy consumption with a stochastic occupancy model, while [12] models energy consumption in residential and commercial buildings. One of the most extensive efforts employing SMPC has been the OptiControl project at ETH Zurich [13]. Although p-Sulu is similar to the OptiControl in that it is also built upon SMPC, p-Sulu employs a different problem formulation that it is goal-directed with chance-constraints. A key innovation behind p-Sulu is that it is able to leverage flexibility in a resident's schedule to achieve further reduction in energy consumption. The CSH controller allows the residents to specify desired ranges of room conditions, minimizes the use of non-renewable energy consumption, and limits uncertainty risk to user specified levels [4]. Earlier research [14] presents a risk-sensitive planner called Sulu, which has the above three capabilities. However, Sulu is an off-line planner: it pre-plans the control sequence for the entire plan duration. A planner of this type cannot execute effectively for the CSH, because: a) the home is operated constantly, and b) the house control requires frequent re-planning every few seconds. To overcome the first challenge, the newly developed planner p-Sulu uses a receding horizon approach: at each planning cycle, a planning problem is solved with a finite duration, which is called a horizon. In the next planning cycle, the planning problem is solved again over a

planning horizon with the same duration starting from the current time, by considering the latest sensory feedback of uncertain parameters. This re-planning process is repeated with a fixed time interval. The second challenge is overcome by building upon an anytime algorithm for chance-constrained programming (IRA).

The REST data structure is self-explanatory allowing an external device or application to access the network without storing information about the network itself. Hence, the network can be expanded freely without upsetting the accessibility of its nodes and it can scale to manage the systems and data of a neighborhood. Moreover, the HTTP server is ready to handle authentication and secure connections, and it can supply secure communications among the nodes of an urban scale network. Finally, p-Sulu can potentially manage the energy consumption of multiple CSH nodes, by allocating and distributing risk at a community level. The development of such a city energy optimizer is subject of ongoing research. And although cooperative energy management is already discussed in the scientific literature [3], [15] a higher level of social awareness will be required to enable such optimization policies to be adopted in practice.

### 3. RESTFUL PROTOCOL

The Representational State Transfer (REST) [2] is a principle commonly used in www. The REST architecture for distributed systems relies on stateless communication employing self-explanatory data and a client-server structure. Its primary advantages are scalability and compatibility. In an early version of the system for the prototype CSH, a serial protocol was implemented. This early version was abandoned for scalability and speed problems. Although we did not implement other systems to compare, the REST architecture was selected because it is easily scalable and compatible with any browser. Four of the main premises of a REST architecture are: (i) scalability of the component interactions, (ii) generality of interfaces, (iii) independence in the deployment of components, and (iv) existence of intermediaries that make the system faster and more secure. RESTful systems are resource-centered, built around the resources that are represented with Unique Resource Identification. In the RESTful protocol for the CSH prototype, the house devices become the resources of the system. They are ordered into a hierarchy similar to that of a web site. The implementation satisfies all the premises of the REST architecture. The system is built around an HTTP server, which is the main interface, so that premise (ii) is satisfied. The HTTP server can access and modify the state of any home device and can provide state information to any devices request it, such as the p-Sulu controller. Building the system around the server enables independence in the deployment of devices, because the devices are simply connected to the server. This satisfies premise (iii). The server enables the management of any number of devices, because the adopted URI system allows adding any number of extra devices. Therefore premise (i) is satisfied. Finally, accessing the devices through the server makes the system faster – although not more secure – than having a chain of components connected in sequence. This satisfies premise (iv). Last, the HTTP protocol provides an intuitive and efficient parallelism between the mode of accessing a common web site and any device in the house network. Three components compose the particular RESTful protocol, namely the *path structure*, the protocol *commands* and the *data format*.

#### 3.1 Path Structure

Following the REST principles, the resources of the system are ordered in a hierarchy represented by a directory-based URI system. This makes the tree structure of the house devices scalable and allows accessing its leaves. Starting from the root of the system,

which represents the set of all house devices, it is possible to navigate through subsets of devices by selecting a path, until arriving at an individual device, at the end of the path. An example appears next:

*/home/*: Is the set containing all the physical nodes of the house network. It will provide the state of all the devices in the house.

*/home/window/*: Is the set containing all the windows. It will provide the state of all the windows of the house.

*home/window/c00*: Is the unique representation of the c00 window, where c00 is a unique identification for a given window. This URI enables the user to interact only with this particular window.

#### 3.2 Commands

The adopted HTTP protocol employs two commands, GET and PUT. This is the minimum set of commands necessary to implement this self-explanatory system. A similar system could possibly include more commands, but we were aiming at being minimal and simple.

GET enables to access the state of a single device, or aggregation of devices in a selected path. PUT permits to modify the state of a device or set of devices in a path.

#### 3.3 Data Format

One of the key features of the implemented protocol is to share a common format of data structure among the devices, independently of the nature of the exchanged data. A JSON format was used for this purpose. The JSON format allows implementing a system that enables any new added devices to understand the structure and adapt. The response to a GET command to the path */home/window/c00* would be:

```
{“id”:“c01”,“tint”:2,“privacy”:true,“open”:8,“url”:“/home/window/c00”,“previous”:“/home/window”}
```

This example shows that the data structure is self-explanatory, because it is indicating the upper bound of the hierarchy in the structure. This data format is standard in REST protocols and allows a device without previous knowledge to browse the structure tree.

#### 3.4 Implementation

The physical and logical infrastructure of the CSH network of devices includes five subcomponents: the *HTTP server*, the *p-Sulu controller*, the *Sage controller*, the *circuit board* and the *KNX network*. These are dedicated to specific tasks that are explained next.

*HTTP server*. The HTTP server provides a shared interface enabling the users to communicate with the devices of the house. It also performs the conversion between the various protocols used by individual devices and the HTTP protocol.

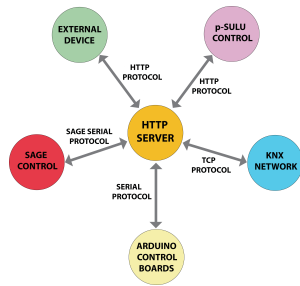
*p-Sulu controller*. The risk-sensitive plan executive of the house, p-Sulu, executes plans with time-evolved goals, which are specified as a sequence of state and temporal constraints. p-Sulu optimizes energy performance based on the weather conditions, so that the user specified constraints of comfort are satisfied.

*Sage controller*: The Sage controller commands the tinting process of the electrochromic layer on the windowpanes of the south façade.

*Circuit board*: The circuit board controls the degree of the window opening and the privacy mode. It commands the state of the poly-dispersed liquid crystal (PDLC) layer that is applied on the windows.

*KNX Network*: The KNX router controls the artificial lighting and the temperature sensors of the house.

A diagram of the implemented communication apparatus of the connected sustainable home is depicted next, in Fig. 2.



**Figure 2. Diagram of the implemented communication apparatus of the prototype connected sustainable home.**

The communication apparatus of the CSH is based on an HTTP server receiving petitions from external devices, and the p-Sulu controller, and returning the required information or initiating the desired change of state for any device in the network. The server uses the HTTP protocol to communicate with the devices that petition to access or modify the state of specific devices in the house. At the same time, the server is using a wide spectrum of protocols to communicate with the house network, as it needs to adapt to the different low-level protocols supported by the individual devices. When the HTTP server receives a petition to access or modify the state of a device in the network, it first loads it in the program. From there, the server decides which device must be communicated and it converges the information to the low-level protocol that is used by this device. Using the specific low-level protocol the server accesses the requested information and then generates an HTTP response that is sent to the device. Hence, the HTTP server translates the HTTP petition to a specific low-level protocol to access a device, and once it receives the requested data it retranslates it to an HTTP response. Three low-level protocols are used: the Sage, the Arduino and the KNX protocols.

The Sage protocol is a serial, low-level protocol used to communicate information to the Sage electrochromic layer of the programmable windowpanes. This protocol is defined by Sage Electrochromics. The HTTP server does the conversion between the information in the HTTP request and its equivalent in the Sage language.

The Arduino protocol is the dedicated serial protocol that we developed to enable communication with the Arduino boards used in the prototype. A presentation of the technical details of this protocol is outside the scope of this paper.

The KNX protocol enables communication with the KNX products used in the house network. This protocol permits to access information related to the state of the sensors, as well as to modify this state when necessary.

A wireless network of sensors (WSN) serves the need for distributed monitoring of the interior and exterior environment in real time. The sensors support the decision-making layer and assist the management of the interior. The optimization core receives data from the sensors and makes use of the house devices, as it would do a house resident. The various types of sensors deployed in the prototype and their sampling rates are listed in Table 1.

**Table 1. Sensor types deployed in the prototype.**

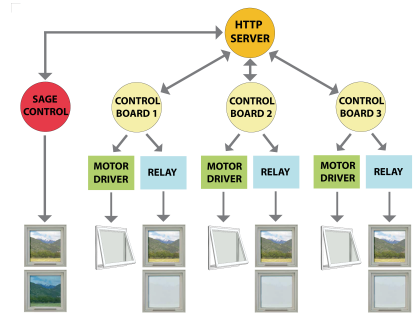
Sensor Type	Sampling Rate
Indoor temperature / humidity	On the order of 15 min – 1 h
Outdoor temperature	On the order of 1 h
Motion	High; "1 min/ motion trigger
Airflow	High; "1 min
Illuminance	High; "1 min

For example, the adjustment of the tinting levels of the 36 programmable windowpanes on the south façade permits the efficient management of solar exposure at the house interior and the exploitation of the high thermal mass house envelope, based on the seasonal conditions and the preferences of the inhabitants. Each windowpane incorporates two kinds of electro-active materials: The first, electrochromic coating (Sage) regulates solar penetration; The second, poly-dispersed liquid crystal film (PDLC) regulates visual privacy. Fig. 3 presents the tinting end states of the windowpanes.



**Figure 3. End-states of the windowpanes. a. PDLC layer is active and electrochromic is inactive; b. PDLC and electrochromic layers are inactive; c. PDLC and electrochromic layers are active; d. PDLC layer is inactive and electrochromic is active.**

Furthermore, 27 windows can be automatically opened by precise amounts so that the permeability to air flow can be adjusted. The states of the windows are managed by p-Sulu, but at a local level each window is driven by its own low-level control and electronics. Each window transmits and receives data through the network. The on/off controlling of individual electrochromic panes happens through the low-level Sage controller. The operation of the PDLC layer happens through a relay circuit. The opening of the windows is controlled through chain actuators (C20 TOPP). The communication of the low-level and the high-level control is achieved through serial ports. A distributed network including 3 circuit boards is used. Each board is able to drive up to 12 electronic chain actuators and 12 relays. Finally, each circuit board contains a communication module, which is connected to the server. The microprocessor on each board processes the incoming commands sent from the server and transmits the control signal to the corresponding actuator. The communication hierarchy of the house is presented in Fig. 4.



**Figure 4. The hierarchy of layers that are involved in the operation of the programmable windows.**

#### 4. OPTIMIZATION ALGORITHM

The risk-sensitive, model-based plan executive p-Sulu built upon the RESTful protocol, takes a threefold approach to minimize energy consumption: Firstly, it minimizes the use of artificial lighting, heating and cooling by adjusting the incoming natural sunlight and heat; Secondly, it exploits the high thermal capacity of the envelope to store heat; And thirdly, it helps the residents to interact with the house systems and to adopt energy-efficient behaviors.

## 4.1 p-Sulu

The plan executive p-Sulu is built upon the iterative risk allocation (IRA) algorithm [4], [14]. p-Sulu explicitly takes into account a SMPC model, which specifies probabilistic state transitions in a continuous domain. It takes as input a linear model describing the system being controlled. The model is of the form,

$$x_{t+1} = A_t x_t + B_t u_t + w_t$$

where  $x_t$  is a continuous state vector at time  $t$ ,  $u_t$  is a continuous control vector at  $t$ , and  $w_t$  is a disturbance whose probability distribution is known. p-Sulu takes as an input a *chance-constrained qualitative state plan (CCQSP)*, which encodes flexible temporal constraints and chance constraints. Given a CCQSP, p-Sulu determines a schedule – an assignment of execution time to events – and a control sequence – an assignment to real-valued control variables. In the presence of uncertainties p-Sulu robustly operates within user-specified upper bound on the probability of constraint violations. p-Sulu sets a safety margin along the boundaries of the constraints, and plans a nominal state trajectory to remain outside of the margin. The width of the safety margin is determined so that the probability of constraint violation is below the risk allocated to each constraint [4], [6]. An intuitive explanation of the IRA-CCQSP algorithm appears in Fig. 5. It is assumed that a resident can specify one of three ranges: Home, Asleep and Away. The temperature must remain between 20 °C and 25 °C while the resident is at Home, between 18 °C and 22 °C while Asleep, and above 5 °C while Away, to ensure that the pipes will not freeze. Home and Asleep episodes are associated with a single chance constraint class, with risk bound 1%. This is the risk the resident is willing to take that the temperature may become uncomfortable. Away episodes are associated to a single chance constraint class with risk bound 0.01%. This is the acceptable risk that the pipes may freeze. The width of the safety (blue) margin is determined so that the probability of constraint violation is below the risk allocated to each time step. In the First Iteration (a), risk is uniformly allocated to all the time steps. In the Second Iteration (b), the IRA-CCQSP algorithm reduces the risk allocated to the inactive constraints and reallocates it to the active constraint. The new safety margin in (b) allows a plan that costs less than in (a).

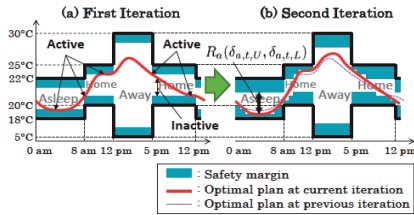


Figure 5. Intuitive explanation of IRA-CCQSP algorithm.

## 4.2 Verification

The previously described system is fully implemented, but there was not sufficient time for data collection and evaluation of the results to assess its performance. In the absence of sufficient building data coming from the prototype, characterizing the performance of its energy management system, we provide a series of simulations demonstrating the capability of p-Sulu for optimal plan execution with chance constraints. The final assessment of the proposed apparatus will be subject of future research, after quantifiable data from the prototype become available. In the simulations, p-Sulu was implemented in C++ and all trials were run on a machine with a Core i7 2.67 GHz processor and 8 GB of RAM. A disturbance  $w_t$  drawn from a Gaussian distribution with standard deviation  $\sigma \sin t = 0.5$  °C was also introduced. The simulation was based on a sample resident

CCQSP that spans a week. In all simulations, we use a planning horizon of 24 hours and an execution horizon of 12 hours. We let  $\Delta t = 1$  hour, so there are 168 total time steps over a week-long schedule. The performance of p-Sulu is evaluated based on energy savings and probability of constraint violations. Two additional models were considered, for comparison: (1) a PID (proportional-integral-derivative) controller that attempts to minimize error based on a given set-point, and (2) Sulu, a deterministic MPC (model predictive control) plan executive, which can handle flexible temporal constraints but can not consider chance constraints. First, p-Sulu is compared to the PID to compare the energy savings of an MPC-based control and a traditional control. By comparing p-Sulu with a traditional PID system we evaluate the level of performance of p-Sulu in association with the current baselines in energy saving. The set point of the PID controller was set 21 °C. Second, p-Sulu is compared to Sulu to show the robustness to failure that arises from risk-sensitive control. By comparing p-Sulu with Sulu we evaluate the benefit of introducing chance constraints. Fig. 6 illustrates the results for January 1 (Fig. 6a) and July 1 (Fig. 6b).

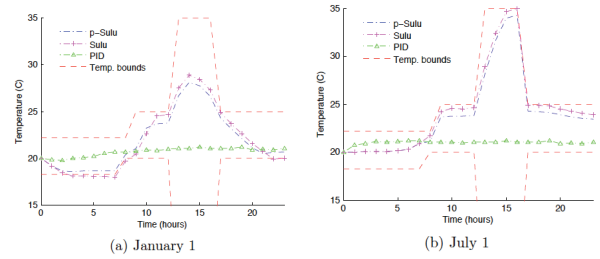


Figure 6. Results of the PID, Sulu, and p-Sulu controllers on January 1 (a) and July 1 (b). p-Sulu is in blue line.

Sulu plans right to the edge of the constraints, often violating constraints when a disturbance is introduced, while p-Sulu respects a safety margin. Table 2 presents the results of the Monte-Carlo simulation on a weeklong scenario, averaged over 100 trials each with  $\Delta = 0.1$ . The table shows that the MPC-based approaches (i.e., p-Sulu and Sulu) can significantly reduce energy consumption. p-Sulu yields energy savings of 42.8% over the PID controller during winter. During spring, summer and autumn, p-Sulu yields 15.3%, 16.8% and 4.4% energy savings, respectively. Sulu also achieves 58%, 22% and 8% energy savings compared to the PID controller during winter, spring and autumn, respectively.

Table 2. Comparison of energy use and failure rate for a simulation of the PID, Sulu, and p-Sulu controllers over a week-long schedule in four seasons. Failure rate is the percentage of time steps with constraint violations.

	spring		autumn	
	energy(J)	fail	energy(J)	fail
p-Sulu	$3.37 \times 10^4$	0	$3.81 \times 10^4$	0
Sulu	$3.09 \times 10^4$	0.30	$3.67 \times 10^4$	0.34
PID	$3.98 \times 10^4$	0	$3.99 \times 10^4$	0

	winter		summer	
	energy(J)	fail	energy(J)	fail
p-Sulu	$1.93 \times 10^4$	0	$3.47 \times 10^4$	0
Sulu	$1.65 \times 10^4$	0.29	--	--
PID	$3.97 \times 10^4$	0	$4.17 \times 10^4$	0

The difference in energy savings between p-Sulu and Sulu is significantly less than that of p-Sulu and the PID controller. But, p-Sulu results in lower savings compared to Sulu because it requires operating within safety margins. However, note that Sulu did not complete the summer simulations due to infeasibility of the planning problem caused by exogenous disturbances. Hence, the summer consumption of Sulu is not included in Table 2. In contrast, p-Sulu is robust to such disturbances because it operates with sufficient safety margins to guarantee that the probability of failure remains always below the given risk bound. The probability of failure of p-Sulu is successfully limited by the given threshold  $\Delta$  in all four experiments. Using constraint violations as a measure of success, p-Sulu outperforms the deterministic approach of Sulu. Averaged across all trials, p-Sulu exhibits a difference of 30.88% in comfort improvement over Sulu. This demonstrates how risk-sensitive control is critical in guaranteeing comfort and encouraging the adoption of the technology: A control causing uncomfortable conditions 30% of the time would be abandoned by the users.

## 5. CONCLUSIONS

It was presented a RESTful protocol for a CSH, a prototype of which is fully realized in Trento, N. Italy. This protocol is probably the first fully implemented application of the WoT paradigm in home automation. The home appliances are mapped into a web-server based structure, where the HTTP protocol is used for communication. A plan executive, p-Sulu, developed upon the communication layer optimizes the performance of the appliances. Following the WoT paradigm, the potentially complex house network is reduced into a HTTP communication that makes the system easily compatible. The JSON data structure of the protocol makes the data-format integration self-explanatory, enabling the network to be scalable to incorporate more nodes and data, at a neighborhood or a community scale. The house controller p-Sulu takes a threefold approach to optimize energy consumption: Firstly, it minimizes the use of artificial lighting, heating and cooling by managing the incoming sunlight and heat; Secondly, it exploits the high thermal capacity of the envelope to store solar heat; And thirdly, it helps the residents to choose energy-efficient behaviors. A key innovation behind p-Sulu is that it is able to leverage flexibility in a resident's schedule to achieve reduction in energy consumption. The RESTful protocol supports the decision-making layer, and enables the communication and execution of performance plans. The HTTP server receives requests from the home appliances and from the control system and returns the requested information, or initiates the desired state change. A wireless network of sensors (*WSN*) serves the need for distributed monitoring of the interior and exterior environment in real time. The optimization core receives data from the sensors, re-calculates the optimization plan every hour and updates the states of the house devices. A wide spectrum of applications can be developed based on the presented RESTful protocol. Some proposals [15] point to the possibility of social integration of the WoT. Furthermore, model-learning may enable the catering of the occupants needs without explicit user input. Although the previously described system is fully implemented, there was not sufficient time for data collection and evaluation of the results to assess its actual performance. The precise assessment of performance will be subject of future research. Preliminary simulation results over a week-long scenario showed energy savings of 42.8% in the winter, 15.3% in the spring, 16.8% in the summer, and 4.4% in the autumn. The level of fidelity that is required in plan recognition may demand a more fine process of receiving and decomposing the home energy signal into its component appliances (energy disaggregation). For finer sensing granularity, one could consider applying the techniques at the circuit level, or even the outlet level.

The realization of the CSH prototype offers great potential that can force many ideas into experiment and practice. The RESTful protocol jointly with the variety of protocols and devices operating in the prototype make it an ideal test-bed for exploring and evaluating the future of the WoT ecosystems at a residential and community scale.

## 6. ACKNOWLEDGMENT

This research was conducted within the Green Home Alliance between the MIT Mobile Experience Lab and the Fondazione Bruno Kessler, in Trento, Italy.

## 7. REFERENCES

- [1] Architectural Reference Model for the IoT - (ARM) update <http://www.iot-a.eu/arm/d1.3/vie>
- [2] Guinard, D., Trifa, V., and Wilde, E. 2010. A Resource Oriented Architecture for the Web of Things. In *Proceedings of Internet of Things International Conference*, Tokyo, Japan, 1-8.
- [3] Fielding, R.T. 2000. Architectural Styles and the Design of Network-based Software Architectures. Ph.D. Thesis. University of California, Irvine.
- [4] Graybill, W. 2012. Robust, Goal-Directed Planning and Plan Recognition for the Sustainable Control of Homes. Master's Thesis, MIT.
- [5] U.S. Energy Information Administration, Independent Statistics & Analysis, <http://www.eia.gov/tools/faqs/faq.cfm?id=86&t=1>
- [6] Ono, M., and Williams, B.C. 2008. Iterative risk allocation: A new approach to robust model predictive control with a joint chance constraint. In *Proceedings of 47th IEEE Conference on Decision and Control*, Cancun, Mexico, 3427-3432
- [7] Guinard, D., and Trifa, V. 2009. Towards the Web of Things: Web Mashups for Embedded Devices. In *Proceedings of WWW 2009 ACM*.
- [8] Kovatsch, M., Weiss, M., and Guinard, D. 2010. Embedding Internet Technology for Home Automation. In *Proceedings of 2010 IEEE Conference on Emergent Technologies and Factory Automation*, Bilbao, Spain, 1-8.
- [9] Kamilaris A., and Pitsilidis, A. 2011. The Smart Home meets the Web of Things. *International Journal of Ad Hoc and Ubiquitous Computing*, Volume 7, Issue 3, May 2011, 145-154
- [10] Overview of the TRON project. 1995. In *Proceedings of the 12<sup>th</sup> TRON Project International Symposium*, IEEE, Tokyo, 100-104.
- [11] Mady, A.E.D, Provan, G. M., Ryan, C., and Brown, K. N. 2011. Stochastic model predictive controller for the integration of building use. In *Proceedings of Twenty-Fifth AAAI Conference on Artificial Intelligence, Special Track on Computational Sustainability and AI*.
- [12] Kolter, J. Z., and J. Ferreira, J. 2011. A large-scale study on predicting and contextualizing building energy usage. In *Proceedings of Twenty-Fifth AAAI Conference on Artificial Intelligence, Special Track on Computational Sustainability, AI*.
- [13] ETH Zurich. 2012. OptiControl. <http://www.opticontrol.ethz.ch/>
- [14] Ono, M. 2012. Robust, Goal-directed Plan Execution with Bounded Risk. Ph.D. Thesis, MIT.
- [15] Guinard, D., Fischer M., and Trifa, V. 2010. Sharing Using Social Networks in a Composable Web of Things. In *Proceedings of the First IEEE International Workshop on the Web of Things*, Mannheim, Germany.