**CZECH TECHNICAL UNIVERSITY IN PRAGUE**

**Faculty of electrical engineering**

# BACHELOR'S THESIS

**2013** **Oleg Prokopenko**

# CZECH TECHNICAL UNIVERSITY IN PRAGUE

## Faculty of electrical engineering

### Department of Cybernetics


# Simulation tool for local electricity distribution networks


Author: **Oleg Prokopenko**

Supervisor: **Ing. Pavel Vrba, Ph.D.**


Study programme: Software technologies and management

Field of study: Intelligent Systems

České vysoké učení technické v Praze
Fakulta elektrotechnická

Katedra kybernetiky

# ZADÁNÍ BAKALÁŘSKÉ PRÁCE

**Student:** Oleg Prokopenko

**Studijní program:** Softwarové technologie a management

**Obor:** Inteligentní systémy

**Název tématu:** Simulační nástroj pro lokální elektrické distribuční sítě

## Pokyny pro vypracování:

1. Provedení rešerše existujících simulátorů energetických systémů.
2. Návrh a implementace zjednodušených modelů chování vybraného vzorku komponent pro výrobu (kogenerační jednotka, vodní elektrárna, větrná elektrárna, solární panel), spotřebu (el. vytápění domu, bojler, ohřev bazénu, klimatizace, ostatní domácí spotřebiče, elektromobil) a uložení elektrické energie (baterie). Cílem je v závislosti na aktuálních podmínkách simulovat množství spotřebovávané a/nebo vyráběné el. energie.
3. Návrh a implementace jednoduchého modulu pro simulaci počasí (teplota, síla větru, sluneční svit) v závislosti na denní době a ročním období. Modul bude též poskytovat údaje o „předpovědi" počasí na zvolenou dobu dopředu.
4. Vytvoření simulačního nástroje, který pro zvolené období (např. 1 rok), zvolenou časovou granularitu (např. 1 hodina) a zvolené zrychlení oproti reálnému času odsimuluje chování microgridu.
5. Návrh a implementace komunikačního rozhraní pro obousměrnou interakci mezi simulátorem a multi-agentním řídicím systémem microgridu. Rozhraním se předávají údaje o aktuálním čase, povětrnostních podmínkách, aktuálně zapnutých el. spotřebičích apod.
6. Implementace jednoduchého modulu pro analýzu výsledků simulace.

**Seznam odborné literatury:** Dodá vedoucí práce.

**Vedoucí bakalářské práce:** Ing. Pavel Vrba, Ph.D.

**Platnost zadání:** do konce zimního semestru 2013/2014

ČESKÉ VYSOKÉ UČENÍ TECHNICKÉ V PRAZE
Fakulta elektrotechnická
L.S.
-7-

prof. Ing. Vladimír Mařík, DrSc.
**vedoucí katedry**

prof. Ing. Pavel Ripka, CSc.
**děkan**

V Praze dne 10. 1. 2013

Czech Technical University in Prague
Faculty of Electrical Engineering

Department of Cybernetics

# BACHELOR PROJECT ASSIGNMENT

**Student:** Oleg P r o k o p e n k o

**Study programme:** Software Engineering and Management

**Specialisation:** Intelligent Systems

**Title of Bachelor Project:** Simulation Tool for Local Electricity Distribution Networks

## Guidelines:

1. Review existing power system simulation tools.
2. Design and implement models of behaviors of a selected subset of power generation (combined heat and power unit, hydroelectric power plant, solar power plant), consumption (electric heating of house, electric water heater, other household appliances, electric vehicle), and storage (batteries) components. The goal is to simulate amount of electric energy production and/or consumption depending on the actual conditions.
3. Design and implement a module for weather simulation (temperature, wind speed, sunshine) being influenced by the time of the day and year period. The module will also provide weather forecast for given time period.
4. Create a simulation engine for simulation of the microgrid behavior. The input parameters are the time period (e.g. one year), time granularity (e.g. one hour) and time acceleration factor.
5. Design and implement a communication interface for bidirectional interactions between the simulator and agent-based microgrid control system. Through the interface the data on actual time, weather conditions, turned on appliances, etc. are handed over.
6. Implement simple module for simulation results analysis.

**Bibliography/Sources:** Will be provided by the supervisor.

**Bachelor Project Supervisor:** Ing. Pavel Vrba, Ph.D.

**Valid until:** the end of the winter semester of academic year 2013/2014

L.S.

prof. Ing. Vladimír Mařík, DrSc.
**Head of Department**

prof. Ing. Pavel Ripka, CSc.
**Dean**

Prague, January 10, 2013

## Declaration

I hereby declare that I wrote the bachelor thesis independently with the help of the supervisor and I have used only the literature mentioned in the reference list. I also declare that I agree with lending or publishing my work with the consent of the department.

Date: 24.05.2013

.............................................

Signature

## Prohlášení

Prohlašuji, že jsem zadanou bakalářskou práci zpracoval sám s přispěním vedoucího práce a používal jsem pouze literaturu v práci uvedenou. Dále prohlašuji, že nemám námitek proti půjčování nebo zveřejňování mé bakalářské práce nebo její části se souhlasem katedry.

Datum: 24.05.2013

.............................................

Podpis

**ACKNOWLEDGEMENTS**

I would like to gratefully acknowledge the supervisor of this thesis Ing. Pavel Vrba, Ph.D. for his very useful help, all the time and valuable advice throughout my research efforts towards this bachelor thesis.

**PODĚKOVÁNÍ**

Chtěl bych poděkovat vedoucímu bakalářské práce Ing. Pavlovi Vrbovi, Ph.D. za jeho velmi užitečnou pomoc, veškerý čas, který mi věnoval a cenné rady při zpracování bakalářské práce.

## Abstract

The goal of the thesis is to create a tool for simulation of local electricity distribution network called microgrid – in the context of intelligent systems of generation, consumption, and distribution of electricity, so called Smart Grids. The tool will provide means for simulating the physical behavior of selected components that produce or consume electricity depending on simulated external conditions (time of day, period of year, temperature, sun shine, wind speed, etc.). The simulator will provide an interface for on-line interaction with a multi-agent control system (which is not part of this work) with the aim of validating various strategies for buying and selling electricity on the local electronic power market. The main output of simulation is the graph of the total energy production and consumption in microgrid within the observed period of time and elementary suggestion on the suitability of microgrid composition with the respect of guaranteeing the supply-demand balancing.

**Key words** electricity distribution networks, smart grid, simulation, data analysis

## Abstrakt

Cílem práce je vytvořit nástroj pro jednoduchou simulaci lokální elektrické distribuční sítě, tzv. microgridu, v kontextu inteligentních systémů výroby, spotřeby a distribuce elektrické energie, tzv. Smart Grids. Nástroj bude umožňovat simulovat fyzikální chování vybraných komponent vyrábějících a spotřebovávajících elektrickou energii v závislosti na simulovaných vnějších podmínkách (denní doba, roční období, teplota, sluneční svit, síla větru, atd.). Simulátor dále bude umožňovat on-line propojení s multi-agentním řídicím systémem (není součástí této práce) za účelem otestování různých strategií pro nákup a prodej el. energie na lokální elektronické burze. Hlavním výstupem simulace jsou průběhy množství vyráběné a spotřebovávané el. energie ve zvoleném období a jednoduchá úvaha o vyváženosti kompozice microgridu z hlediska vybalancování výroby a spotřeby energie.

**Klíčová slova** elektrická distribuční síť, smart grid, simulace, analýza dat

## Résumé

The main goal of this bachelor thesis is to give an overview of Smart Grid conception which provides electrical power to smaller industrial and domestic users over the entire supply area. Besides explanation of its features, components, principles of work, operation and used mechanisms, this work discusses issues such as reliability, advantages and disadvantages of smart grid systems, benefits from its using, implementation of simulation engine in Java with description of its logic and main functions. Review of existing simulation software.

The fact that the power consumption of humanity is growing every day, the production opportunities increase, the mechanisms of interaction between costumers and providers are becoming more complex. This poses new challenges to the power supply systems.

This thesis contains six chapters. Chapter 1 “Introduction” contains information about thesis objectives.

Chapter 2 “Smart Grids” introduces the concept of smart grid systems and gives an overview of main components, which may be used wile constructing a smart grid power system.

Chapter 3 “Overview of existing simulation tools” describes main features of different software products.

Chapter 4 “Project overview” gives is a brief description of classes used in simulation engine, logic and principles of its work, necessary calculations and communication interface with multi-agent control system.

Chapter 5 “Tests” contains the results of test with different time periods and different periods of the year.

Chapter 6 “Conclusions” provides author conclusions.

## Contents

## List of figures

# Chapter 1

## Introduction

The goal of the thesis is to create a tool for simulation of local electricity distribution network called micro grid – in the context of intelligent systems of generation, consumption, and distribution of electricity, so called Smart Grids. The tool should be able to provide simulation of the physical behavior of solar, wind and gas power plants that produce electricity and behavior of buildings that consume electricity. This simulation engine should allow simulating external conditions (time of day, period of year, temperature, sun shine, and wind speed) that will have an impact on behavior of selected components and force them to change its production and consumption of electricity.

The simulator should provide an interface for interaction with a multi-agent control system (which is not part of this work) with the aim of validating various strategies for buying and selling electricity on the local electronic power market. With the help of this interface, multi-agent system should be able to carry out the adjustment of power plants output and buildings energy-consuming equipment, to obtain information concerning the parameters of the network components, as well as receive constant updating of the current state of the power grid.

In order to make further simulation data analysis more convenience, the simulator must be able to store information into the database. Information about each object participating the simulation should be stored, its parameters, and the amount of produced or consumed electric energy. Information about weather conditions, which have a direct impact on system performance, also must be stored.

There should be developed the module for loading necessary information needed for initialization from external XML file. The main output of simulation is the graph of the total energy production and consumption in microgrid within the observed period of time and weather conditions that had an impact on output.

# Chapter 2

## Smart Grids

A smart grid is an electrical grid that uses modern technologies to collect, process and react on information from objects involved in work of smart grid. Basic information contains facts about behavior of suppliers and consumers. The main task of such a grid is to improve automatically the efficiency and reliability of such a system.

## 2.1 Introduction to smart grids

Energy problems, climate instability, predicament of world's economy, environmental pollution and a lot of other facts becoming a very big global issues and affect people's lives and world economic sustainability. All these factors push and encourage people to look for new ways of development, both economic and technical. It's no secret that technology is not standing still, and it is rapidly growing every minute and every hour, and yet the issue of rational use of received new features, technologies and possibilities remains urgent every day. And there is no getting away from the fact that electricity automation is not an exception. Main development stimulating factors are:

- Climatic change
- Growing energy consumption from day to day
- Security challenges
- The need in the ability to manage the available resources more efficiently
- Desire to use clean energy

In general, the smart grid system can be considered as a very effective tool for achieving economic and technical objectives in the pursuit for clean energy and improving the ecological situation on the planet.

There is no production, town or big city which can act without its own grid. And the more effective management of such a system, the more effectively it can carry out its own task, both in technical and in economic sense.

Combined into a single platform, new modern technologies provide a new approach to the construction of power grids, going from a strict formula "generator - network - consumer" to a more flexible one. With the help of smart grid, there exists a clear possibility according to which each node of the network can be an active participant. In case, when conditions of work change the smart grid automatically can provide reconfiguration of its tasks, structure and purposes.

## 2.2 Renewable energy sources

Until recently, for a variety of different reasons, primarily because of the huge reserves of conventional energy resources, the development of renewable energy sources in the energy policy of many countries received little attention. In recent years, the situation began to change noticeably.

The need for a better environment, new opportunities to improve the quality of life and the desire to create more green energy, the concrete steps towards a low carbon economy.

It is known that electric energy, one of the consumption tools of today, is being increasingly required by all the countries in all their activity environments. Besides, the quantity of production and consumption of energy and its environmental and efficiency qualities are among the elements that contribute to development levels of countries. Energy need of the world have been covered mostly from fossil base fuels so far. [1]

And there is no doubt about the facts that exactly for this reason many countries are highly dependent on providers of such energy sources. From another point of view, such energy sources are harmful in a sense of environment pollution.

## 2.3 Role of solar power plant

The sun is the main source of energy for the earth's surface. It has served the life on earth with its endless radiation source for millions of years. Nowadays, the solar energy is still considered as one of the major alternatives for a safe and clean environment. Solar radiation is a major source of energy in the nature that circulates the ocean, runs water on the ground, blows winds in the air and continuously changes the face of earth. In the world of life, the sun energy is the first resource of all motions in the living cells. Even the fossil energy we use today as coal or petroleum is the far past remnant of the solar energy radiated over the millenniums. [2]

Nowadays modern technology is to collect the radiation from the Sun with the help of Photo Voltaic (PV) panels. It acts like a leaf of tree. A mass of silicon with an expanded surface can similarly absorb the solar radiation and perform some electricity. A PV Array is made up of PV modules, which are environmentally-sealed collections of PV Cells—the devices that convert sunlight to electricity. Often sets of four or more smaller modules are framed or attached together by struts in what is called a panel. Dirt and dust can accumulate on the solar module surface, blocking some of the sunlight and reducing output. The maximum power output of the total PV array is always less than the sum of the maximum output of the individual modules. This difference is a result of slight inconsistencies in performance from one module to the next and is called module mismatch and amounts to at least a 2% loss in system power. [3]

## 2.4 Role of wind power plant

In the past 30 years, the size of wind turbines and the size of wind power plants have increased significantly. Notably, in Tehachapi, California, the amount of wind power generation has surpassed the infrastructure for which it was designed. At the same time, the lack of rules, standards, and regulations during early wind development has proven to be an increasing threat to the stability and power quality of the grid connected to a wind power plant. Fortunately, many new wind power plants are equipped with state of the art technology, which enables them to provide good service while producing clean power for the grid. The advances in power electronics have allowed many power system applications to become more flexible and to accomplish smoother regulation.

Applications such as reactive power compensation, static transfer switches, energy storage, and variable-speed generations are commonly found in modern wind power plants. [4]

The wind generator is developed to kinetic energy of wind into electric energy. There exists proportion between generated and the speed of wind. Great importance, in location for wind generator, plays the wind speed which must be continuous and high. When the air flow hits the turbine wings, it causes the turn of them and so does the connected mill. Generators convert this mechanical energy to the electric energy. Low power wind turbines in most cases use permanent magnet generators and high power turbines are interconnected with synchronous and asynchronous.

## 2.5 Role of energy storage (battery)

While electricity from the Sun can be generated during the sun day, the load of a power grid can stay without energy source during the rest of the time. The same problem we can see with the use of wind turbines, there is no such a limit with wind source like with sun radiation, but it's not a secret that wind doesn't blow constantly and with the same flow power. It means that during some hours of a day wind generator is not a sours of energy. Keeping these facts in mind, engineers made a solution for this situation. Such an interruption can be secured with the help of battery power bank. In other words, generated but still not used electricity can be stored into the battery and later can be feeded when the shortage of electricity occurs in a power grid.

There are two main schools of thought regarding deployment of battery storage technologies on the electric power distribution system. One is to provide centralized storage. The other camp would prefer to see smaller energy storage systems distributed on the distribution feeders, networked together and remotely controlled at the substation, absorbed independently according to the needs of the power system. [5]

## 2.6 Role of gas power plant

Solar and wind power plants are holding their promise of clean and limitless energy supply, but nowadays they still suffers from high costs and an inherent disadvantage of not working when there is no sun in the sky or wind isn't blowing. As mentioned earlier, to solve this problem, it's possible to use the battery storage, but it is also possible to use the traditional source of electrical energy. Such as gas power plant which can serve as the main power source or as the plant of support, it depends on the concept of the system. There is no getting away from the fact that developing of hybrid power systems can increase the efficiency but also reduces the carbon footprint of natural gas power plants. The main feature of a gas power plant, which is perfectly suited for use in such hybrid systems, is the fact that the gas power plant does not depend on weather conditions and can supply a constant amount of electrical energy. Gas-fired power technology has long been tested and used in practice everywhere. New technologies of gas-firing have allowed natural gas power plants to play an increasingly important role in the clean generation of electricity. Natural gas is the cleanest of all the fossil fuels, as evidenced in the Environmental Protection Agency's. Composed primarily of methane, the main products of the combustion of natural gas are carbon dioxide and water vapor, the same compounds we exhale when we breathe. [6]

# Chapter 3

## Overview of existing simulation tools

Power system simulation tools are well known computer simulation programs class. The main aim of such programs is to focus on the control and simulation of different power systems.

### 3.1 OpenDSS

OpenDSS (Distribution System Simulator) is an open-source code software application originally developed by Electrotek Concepts in 1997. One of the reasons behind the development of this tool was the calculation of harmonics and interharmonics on the electrical distribution network. OpenDSS works by command line with its own console. It is also designed to be used with Matlab or Excel. It can be downloaded for free online and does not require installation. It is programmed in Delphi (object-oriented Pascal) language. Based on how it was designed, it can be integrated into different software application through a COM interface. OpenDSS makes it possible to define the different elements of the electrical distribution network: lines, cable, capacitors, voltage regulators, transformers and loads. It can solve imbalanced three-phase circuits and do simulations taking into account the impedance of the distribution transformers. Even if the software works by command line, an interface was developed from the program for the graphs. Moreover, if geographical data are integrated, the software can draw the shape of the electrical distribution network. [7]

### 3.2 GridLAB-D

Gridlab-D is an open-source software application developed by PNNL (Pacific NorthWest National Laboratory), a laboratory with the US Department of Energy. GridLAB-D is also an electrical distribution network simulation application – which explains the letter D in the name. Unlike OpenDSS, GridLAB-D was first and foremost developed as a residential load simulator GridLAB-D can be used to model a home's electric loads (dishwasher, refrigerator, freezer, dryer, etc.), as well as thermostat-controlled loads. The approach is different for household appliance and water heater simulation, or air conditioning and heating demand.The software includes a water heater model and thermal parameters of a house to calculate heating, air conditioning and hot water needs. In fact, the number of square feet, wall and window insulation can be calculated. The type of heating/air conditioning (electrical, thermal) can then be defined, and the software calculates the need. In order to reflect temperature variations, it uses typical meteorological year data. For the time being, only US data are available. [7]

### 3.3 APREM

APREM (Analyse Paramétrique des Réseaux Électriques [Parametric Analysis of Power Grids] with Matlab) was developed at Montreal's École Polytechnique in order to respond to certain specific Hydro-Québec needs. APREM functions with Matlab and uses this software's object-oriented programming, combined with MEX files (files written in C++ and pre-compiled to accelerated calculations). It makes loop simulations possible by varying the load, the generation and even the topology of the grid. The user must first of all define his grid (impedances, transformers, loads, sources, switches, etc.), and can then change any circuit parameter at will and recalculate the solution However, APREM enables the user to easily and dynamically change the grid topology. [7]

## 3.4 DIgSilent PowerFactory

DIgSilent PowerFactory was produced by DIgSILENT GmbH company which is a consulting and software company providing engineering services in the field of electrical power systems for transmission, distribution, generation and industrial plants.

Development of a new version of this software began in 1993. The main objective was to improve the modeling algorithms and the use of modern software technologies. In 1997, there was PowerFactory - new generation of software for the analysis of electrical systems. The main advantages of this product is to enhance the stability calculations, developed a graphical user interface, robust object-oriented database. All this ensured the highest efficiency in the use of software to solve everyday problems. In addition to the stand-alone solution, the PowerFactory engine can be smoothly integrated into GIS, DMS and EMS supporting open system standards. PowerFactory is enabled with Database offline mode with local caching and re-synchronization, automatic rule-based database housekeeping and topological network structures visualization. PowerFactory contains built-in programming tools. This function displays the software to a new level, allowing to create some new complex functions based on the built-in functions.[8]

# Chapter 4

## Project overview

To solve the problem, namely the creation of a simulation engine for realistic weather conditions for alternative energy sources, it was decided to simulate the behaviour of such power plants like solar, wind and gas. To provide a realistic indication of weather, the basis of the weather forecast was taken from weather conditions in Prague for the whole year of 2012. As a consumer of electrical energy a usual building or multiple buildings, consuming electricity to provide heating has been selected.

## 4.1 Main purposes

- Design and implement models of behaviors of a selected subset of power generation (combined heat and power unit, hydroelectric power plant, solar power plant), consumption (electric heating of house, electric water heater, other household appliances, electric vehicle), and storage (batteries) components. The goal is to simulate amount of electric energy production and/or consumption depending on the actual conditions.
- Design and implement a module for weather simulation (temperature, wind speed, sunshine) being influenced by the time of the day and year period. The module will also provide weather forecast for given time period.
- Create a simulation engine for simulation of the microgrid behavior.
- Design and implement a mechanism for storing the real-time simulation data into a database (mainly amount of produced/consumed energy) for future off-line analysis.
- Design and implement a communication interface for bidirectional interactions between the simulator and agent-based microgrid control system. Through the interface the data on actual time, weather conditions, turned on appliances, etc. are handed over.
- Implement simple module for simulation results analysis. The module produces a graph of total consumption and production in given period and assesses the suitability of microgrid composition in the context of ensuring power supply-demand balancing

## 4.2 Architecture

According to the task, I decided, equipment, operation of which will be simulated by the program, to divide into classes. What is clear is that the equipment, according to the logic, should be divided into equipment producing electrical energy and electrical equipment consuming energy. However, equipment producing electrical energy, in its turn, is divided into subclasses. Every subclass contains some equipment, which requires certain weather conditions for the production of electricity (gas power does not depend on weather conditions). Thus, we obtain three groups of equipment generating electricity from:

- Solar energy
- Wind energy
- Gas processing energy.

To centralize the management of all the objects of the group or individual objects, each group contains a separate entity exercising control and management of individual objects of the group or the entire group as a whole. In other words, each group has its own manager:

- Solar power plant manager
- Wind power plant manager
- Gas power plant manager

In the same way the group combined with equipment that consumes electrical energy, namely, buildings that use electricity for heating. The group also its own manager:

- Civil building

And accordingly:

- Building manager

There is no getting away from the fact that other kinds of managers can be found in the program:

- Time manager
- Database manager
- Global manager

Time manager - handles all operations related with time. Database manager - handles all operations related with database manipulations. Global manager - handles management of all managers and represents communication interface for bidirectional interactions between the simulator and agent-based microgrid control system.

A module for weather simulation is represented by a special java class which is called "*WeatherStation*". The main task of this class is to provide information about the weather conditions at the particular date and time of the day.

This program, like any other, for the correct execution of the task is waiting for input parameters. All parameters needed for initialization, this program receives from the .xml file. For processing the incoming data is responsible *Configuration* class, whose main task is to load parameters of the simulation and hardware configuration. The configuration file has a strict recording developed format where each tag defines an option or equipment parameter. Examples of records about building, solar power plant and wind power plant in configuration file:


```
<House>
  <agentName>House1</agentName>
  <type>Civil</type>
  <maximalPrice>9</maximalPrice>
  <priceCoefficient>2</priceCoefficient>
  <idealMaxConsumption>500</idealMaxConsumption>
  <heatingCapacity>5000</heatingCapacity>
```

```
  <heatingResistance>250</heatingResistance>
  <maxHeatingPower>20000</maxHeatingPower>
  <startTempInside>18</startTempInside>
  <startHeatingPower>0</startHeatingPower>
</House>
```

Solar power plant:

```
<SolarPowerPlant>
        <agentName>SolarPowerPlant1</agentName>
        <type>Solar</type>
        <minimalPrice>3.1</minimalPrice>
        <priceCoefficient>5.231</priceCoefficient>
        <idealMaxProduction>245</idealMaxProduction>
        <idealSunRadiation>1000</idealSunRadiation>
        <countOfSources>500</countOfSources>
        <ambientTempIdeal>25</ambientTempIdeal>
        <tempCoefficient>0.43</tempCoefficient>
 </SolarPowerPlant>
```

Wind power plant:

```
<WindPowerPlant>
        <agentName>WindPowerPlant1</agentName>
        <type>Wind</type>
        <minimalPrice>4.5</minimalPrice>
        <priceCoefficient>4.38</priceCoefficient>
        <idealMaxProduction>1500000</idealMaxProduction>
        <rDiameter>70</rDiameter>
        <windRatedSpeed>11.8</windRatedSpeed>
        <windCutOutSpeed>24</windCutOutSpeed>
        <countOfSources>1</countOfSources>
</WindPowerPlant>
```

The body of simulation is contained in *SimulationTask* class which extends *java.util.TimerTask*. *TimerTask* class represents a task that can be scheduled for one-time or repeated execution by a Timer. Simulation process by itself starts in overridden *run* method of *SimulationTask* class.

## 4.3 Imported open-source libraries

**JFreeChart** (jfreechart-1.0.14 used version) is a free Java chart library that makes it easy for developers to display quality charts in their applications. The JFreeChart project was founded more than twelve years ago, in February 2000, by David Gilbert. Today, JFreeChart is the most widely used chart library for Java. [9]

**Joda-Time** (joda-time-2.2 used version) is open-source time-handling library and licensed under the business friendly Apache License Version 2.0. It provides a quality replacement for the Java date and time classes. The design allows for multiple calendar systems, while still providing a simple API. The 'default' calendar is the ISO8601 standard which is used by XML. The Gregorian, Julian, Buddhist, Coptic, Ethiopic and Islamic systems are also included. Supporting classes include time zone, duration, format and parsing. The library is intended to provide all the functionality that is required for date-time calculations. [10]

## 4.4 Power plant and building implementation

As mentioned earlier, there are three types of power stations in the program: solar, wind and gas. To refer these types enumeration is used. It's a special data type that enables for a variable to be a set of predefined constants. Type names are located in *TypesOfElEquipment* enumeration: SOLAR, WIND, GAS.

Each power plant is a separate instance of its own class. Every concrete power plant class extends general class *PowerPlant.* In its turn *PowerPlant* class implements *Manageable* interface and extends class *java.util.Observable. Observable* class represents an observable object, or "data" in the model-view paradigm. It can be subclassed to represent an object that the application wants to have observed. An observable object can have one or more observers. An observer may be any object that implements interface *Observer*. After an observable instance changes, an application calling the *Observable's notifyObservers* method causes all of its observers to be notified of the change by a call to their *update* method. In other words, the multi-agent system is provided with the ability to assign to each power station one or several observers, who will be alerted if the actual amount of electrical energy produced by station changes.

All plants can be in several states, namely: normal and closed states. Behavioral software design pattern, called "State", was used for this purpose. Every state gives power plant special functionality. For example, power plant in a stopped state can't give information about current amount of electricity it produces because the station, actually, produces nothing, it is stopped.

Power plant doesn't include the only one source of electrical energy, means, for example, solar power plant is not the only one PV panel, as well as wind power plant doesn't include only a single turbine. In order to indicate count of turbines or PV panels, in the configuration file, one can specify the exact amount of electricity-generating sources for each power plant.

As well as power plant, each building represents a separate instance of its own class, implements *Manageable* interface and extends class *Observable*. What allows multi-agent system to monitor the temperature change in the building relative to the ambient temperature and the activity of the heating system, and based on this data in order to set correct heating power to achieve the optimum temperature inside the building.

Interface *Manageable* brings basic functionality to all classes which implement it. It brings the ability to provide information like type of manager, equipment name or ability to calculate current parameters and produce/consume possibilities.

### 4.4.1 Calculations carried out by building

During simulation with the turn of a new hour each building is supposed to calculate needed amount of necessary electricity in order to provide heating in terms of rising up or support actual optimal temperature inside. (This action also can be provided by control multi-agent system) For this purpose the next equation is used:

$$(Tin \times Hc) - (\Delta T \times Hr \times T) + (Hp \times T) = Tend \times Hc \qquad (1)$$

Where: *Tin* – temperature inside the building, *Hc* – heating capacity, $\Delta T$ – difference between the temperature inside and outside the building, *Hr* – heating resistance, *T* – period of time, *Hp* – heating power, *Tend* – needed temperature.

**Heating capacity (*Hc*)** specifies the amount of heat required to change the temperature inside a building by one degree without loses.

**Heating resistance (*Hr*)** specifies amount of heat which building is losing with the difference between temperature inside and outside at one degree.

To get the amount of heating power *Hp* to reach optimal temperature *Tend:*

$$Hp = (\Delta T \times Hr) + (Tend - Tin) \times Hc \qquad (2)$$

### 4.4.2 Calculations carried out by wind power plant

The output power of a wind turbine is given by the following equation:

$$P = \frac{1}{2}\rho\pi R^2 C v^3 \qquad (3)$$

Where ρ is air density(it was decided to use constant value 1.225 kg/m$^3$ of air density at sea level and at 15°C according to ISA (International Standard Atmosphere),[12] air has a density of approximately), $v$ is a wind speed, and power coefficient *C.* In this equation *R* means rotor radius and thus $\pi R^2$ is a swept area.

**Power coefficient** is a non-linear function depending on the tip speed ratio λ and β blade pitch angle. Having in mind that 0.35-0.45 are common values of power coefficient even in the best designed wind turbines, it was decided to use 0.4 value.

### 4.4.3 Calculations carried out by solar power plant

Efficiency of solar panel is the ratio of the panel's output to the energy of sunlight. The energy conversion efficiency is a certain percentage of solar panel's ability to convert solar radiation into electrical energy. By convention, solar cell efficiencies are measured under standard test conditions. These conditions specify a temperature of 25°C and an irradiance of 1000 W/m$^2$. [13] With the increasing temperature, productivity decreases. In this simulation program decreasing coefficient is constant for all solar panels during tests 0.43.This coefficient was taken from official documentation of "Suntech" solar panels producer. [14] It can be set with the configuration file and inner tag of solar power plant "tempCoefficiant".

## 4.5 Weather simulation

One of the key components and main tasks of this program is the weather simulation.  For the accuracy of the information about the weather conditions, information was taken from Strahov dormitory weather station, supported by "Silicon Hill" club, measured data for the year 2012.

Type of weather data is declared in enumeration *TypesOfWeatherData:*

- TEMPERATURE
- HUMIDITY
- WIND
- SOLARPOWER
- PREASURE

Class called *WeatherStation* represents real weather station and brings all functionality to get correct weather data right on time it needed. It can supply with needed information on specific weather data such as temperature or pressure etc., or return an object *WeatherCondition* containing all available information about weather condition on the specified day and time.

In order to get archived information about weather conditions during the year 2012 from Srahov weather station web-side, I've wrote a special script in Java, which sends requests for each day of the year about weather conditions and automatically saves the information to a text file. So, having received complete information on the data interested me, I had a little to edit a text file with the saved data, called *weather.txt* and prepare it for processing by *WeatherStation* class.  Therefore, after each program request to update weather information, WeatherStation's methods *getWeatherData* or *getWeatherCondition* refers to a text file, which contains necessary information. After providing this information, it is stored into SQL database. So that if necessary one can always request the database and obtain this information for further processing.

## 4.6 Database

In order to store required information about program steps, weather conditions, energy produce results and multi-agent systems acts according to setting required power plants produce amount of electricity and level of heating power in buildings, it was decided to use Java DB.

The Java DB database is Sun's supported distribution of Apache Derby. Java DB is a fully transactional, secure, standards-based database server, written entirely in Java, and fully supports SQL, JDBC API, and Java EE technology. The Java DB database is packaged with the GlassFish application server, and is included in JDK 6 as well. It is a relational database management system that is based on the Java programming language and SQL. Java DB is the Oracle release of the Apache Derby project. [11]

Every manipulation with database is provided with *DBSManager* class. Program stores information about buildings, power plants, consumed electricity, produced electricity, required produce provided by multi-agent system, weather conditions and periodic data as a product of the simulation. Corresponding tables:

- createBuildings
- createPowerPlants
- consume
- produce
- requiredProduce
- weather
- periodical

## 4.7 Communicational interface

One of the main tasks of this project is to design and implement a communication interface for bidirectional interactions between the simulator and agent-based microgrid control system. Mentioned earlier, the *GlobalManager* class provides the necessary interface for multi-agent systems for correct interaction and simulation control. With the help of this interface control system can:

- Register/unregister power plant or building observers.
- Register time observer, which always will be updated with current simulation hour ticks.
- Unregister all observers of observable object.
- Get count of registered observers.
- Get information about current electricity produce of any power plant.
- Get technical information about buildings and power plants, such as maximum heating power, maximal possible produce of any power plant, etc.
- Get specified or complete weather condition data with provided date and time.
- Get count of specified power plants involved in the simulation.

- Get current simulation time as an object, called *TimeSample.* This object provides current simulation time, namely: year, month, day and hour.
- Ask to calculate possible produce of power plant with provided weather condition data.
- Ask to calculate temperature inside a building according to preset heating power and provided weather condition data.
- Get count of buildings or power plants.
- Inform power plant about amount of required electricity produce.
- Set heating power in a building.
- Switch any power plant to on/off state.
- Get state of any power plant.
- In case of need in weather forecast for specific date, but not in an accurate data from archive, interface can provide agent-based system with this information. It means that archived data will be returned to control system a little bit modified to simulate the weather forecast which is never accurate.

To address a specific object, power plant or building, while using *GlobalManager's* methods, it's necessary to provide the name of the object you are going to manipulate with. Every object which behavior is going to be simulated in this program has its own unique name, as its identifier. It means that during initialization, power plant or building receives and stores its own name from configuration file, and agent-based system is supposed to do the same thing, while creating agents to represent simulated objects. And these agents should have the same names as simulated objects they are representing. Of course, it's possible to obtain all names of power plants and buildings involved in this simulation with the help of interface and rename agents or just store these names for later use, but it's recommended to name agents of control system with the same names.

## 4.8 Simulation logic

If we are speaking about concrete steps, program needs to work in a correct way, first of all I can quite confidently say, that starting point is program's initialization. From the initialization it gets all information it needs to set all its parameters. As was mentioned earlier, for initialization it uses configuration file in .xml format. Every tag of configuration file brings some important and necessary information.

*Timer*, inner content of this tag brings information for simulation, related to time parameters:

*year from* – year, simulation starts from.
*month from* – month, simulation starts from.
*day from* – day, simulation starts from.
*year to* – year, simulation end point.
*month to* – month, simulation end point.
*day to* – day, simulation end point.
*day duration in seconds* - count of seconds, to simulate a day.
*simulation period* - the period in which data being summarized during the simulation.

Considering that the program uses several different types of power plants, the description of them somewhat differs with each other.

*WindPowerPlant* tag describes parameters of wind power plant. According to the same logic, configuration file contains such tags as: *SolarPowerPlant*, *GasPowerPlant* and *House;* with parameters of these objects.

With the start of program initialization, initialization of database storage is taking place to store information about newly created objects of simulation in future. After successful creation of DB, appropriate managers are initialized: *TimeManager, BuildingManager, PowerPlantManager*. All managers are programmed with "Singleton" programming pattern. It means that the only one manager of this type can be created.

In the next step of initialization its turn of creation of power plants and buildings according to received information from configuration file. Creating a new object, class *Initializator* automatically stores information about this object into the database.

The last step of initialization belongs to *TimeManager*. There will be set the time frames, count of days for simulation and duration of a day in seconds. When the initialization is over, program waits until agent-based system prepare itself, register all needed observers and invoke *GlobaManager's* method *startSimulation.* This method will create a new object called *SimulationTimer*. In the constructor of this object will be created new *java.util.Timer* and scheduled to perform *SimulationTask's run* method every second, in case if one second is used as a parameter for "day duration" tag in configuration file. *SimulationTask* class extends *java.util.TimerTask.* It means that every activation of *run* method will simulate only a day and this action will be performed until the whole amount of needed days is not simulated.

Inside the *run* method every hour a sample of weather condition is taken, stored into the database and used as a base for creating a new object, called *WeatherCondition*, which is passed as argument to all power plans and buildings in order they to calculate their new output according to new weather conditions. Output of all simulated objects is stored into database, where it changed or not. Then all observed power plants or buildings notify theirs observers about new output or its condition.

Every new hour observers of time are notified about the fact that the next hour starts. During an hour all power plants are waiting for income data from control system to set required produce and buildings are waiting for information how to set its heating level. New state of objects after processing income information is stored into database.

# Chapter 5

## Tests

### 5.1 Simulation of wind power plant

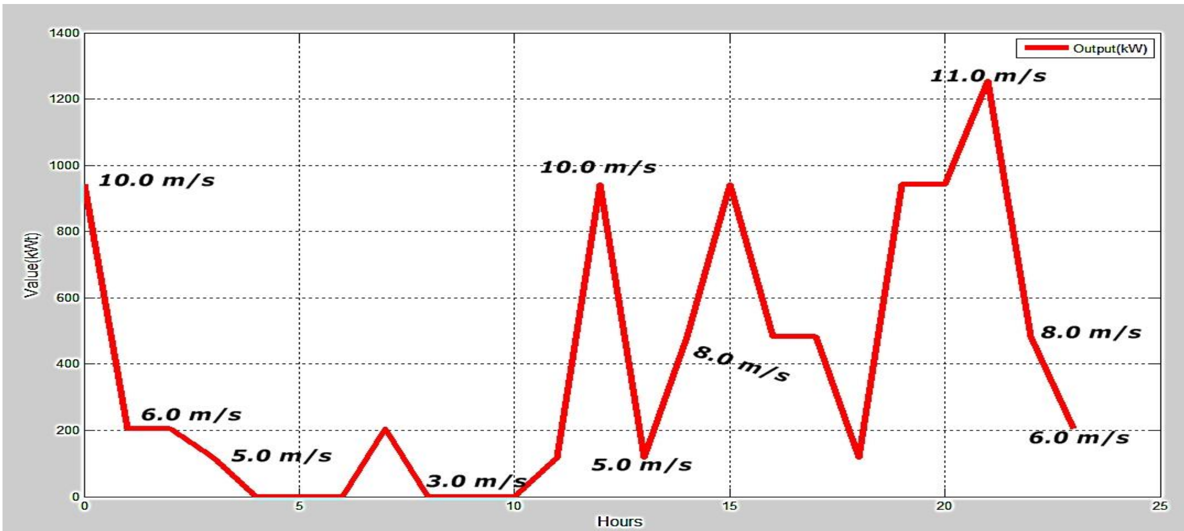


Figure 16. One day wind power plant output. Date: 2012-01-01

**Figure 1** demonstrates output of wind turbine during one day and dependency of output on a wind speed.

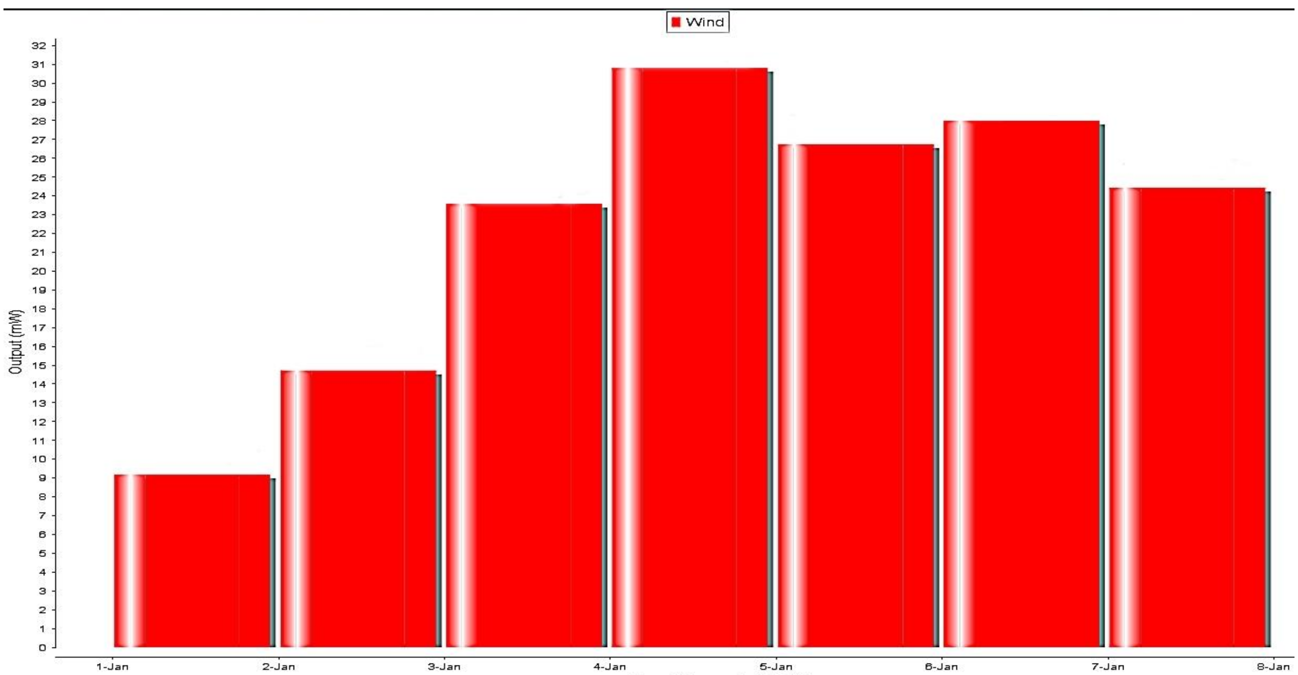


Figure 17. One week wind power plant output. Date: 2012-01-01 - 2012-01-07

**Figure 2** illustrates output in $mW$ of one wind turbine during a week. Values of average wind speed during each day are shown at **Figure 3**

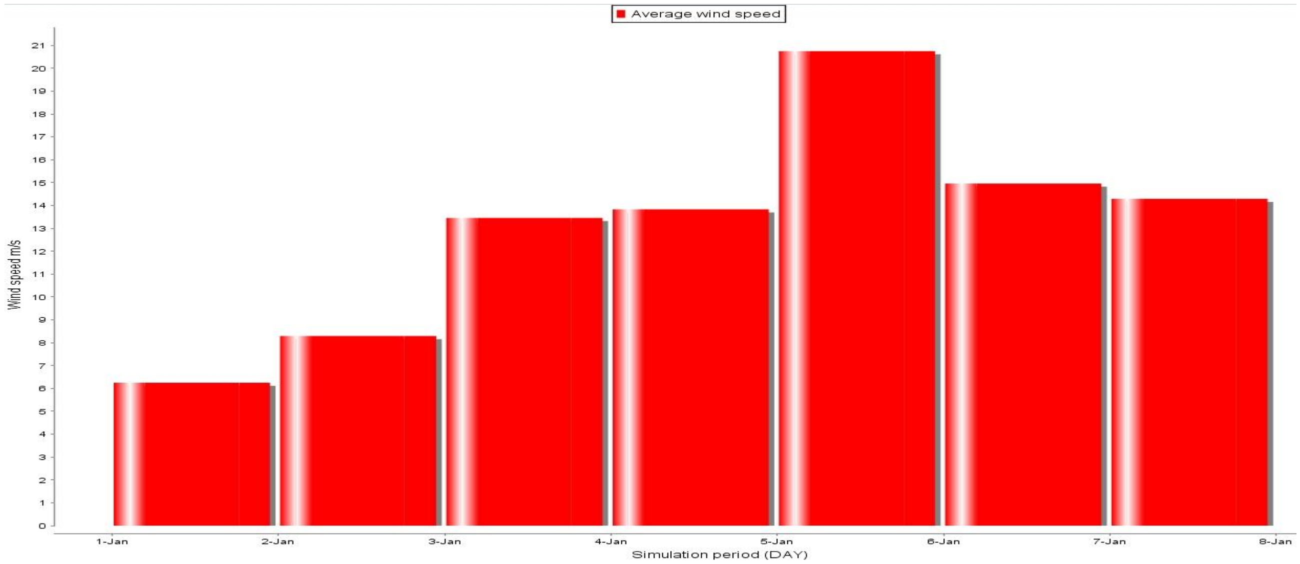


Figure 18. Average wind speed during a week. Date: 2012-01-01 - 2012-01-07

**Figure 3** shows average wind speed each day during a week. One can notice that the biggest average wind speed refers to date 2012-01-05 but the output during this day as shows **Figure 2** is not the biggest one. It caused by the fact that if wind speed bigger than the value of wind cutout speed, specified in configuration file, than wind turbine stops. In this test wind cutout speed is set for 24m/s.

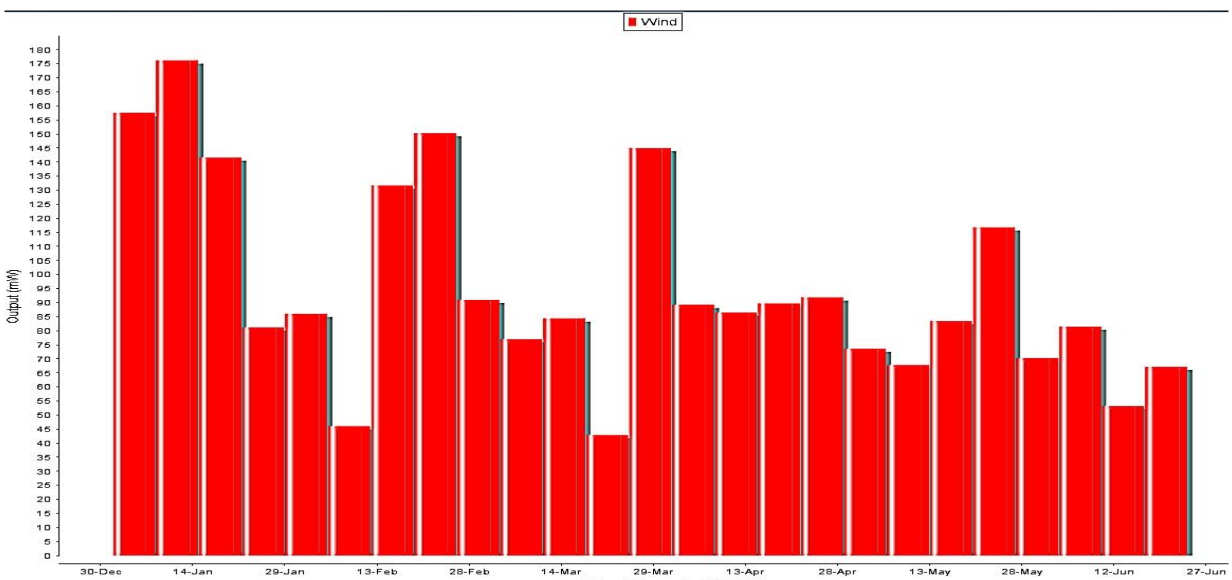


Figure 19. A half of a year wind power plant output.
Date: 2012-01-01 - 2012-07-01

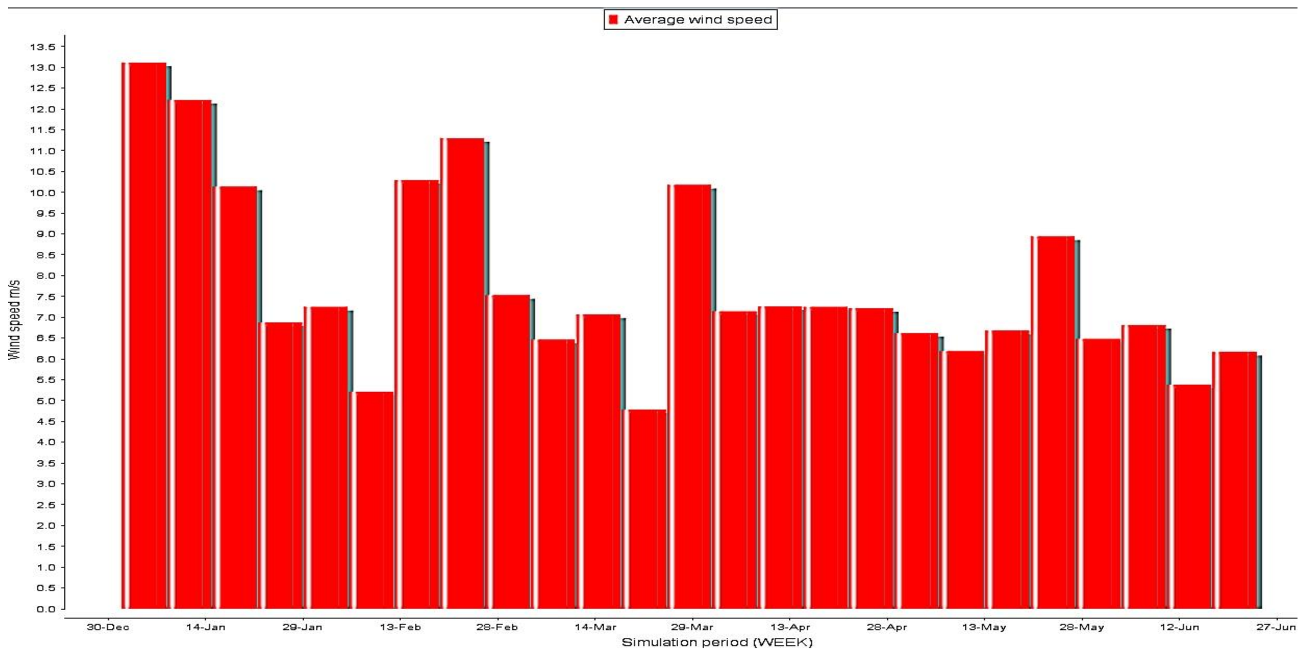


Figure 20. Average wind speed during a half of a year. Date: 2012-01-01 - 2012-07.01

## 5.2 Simulation of solar power plant

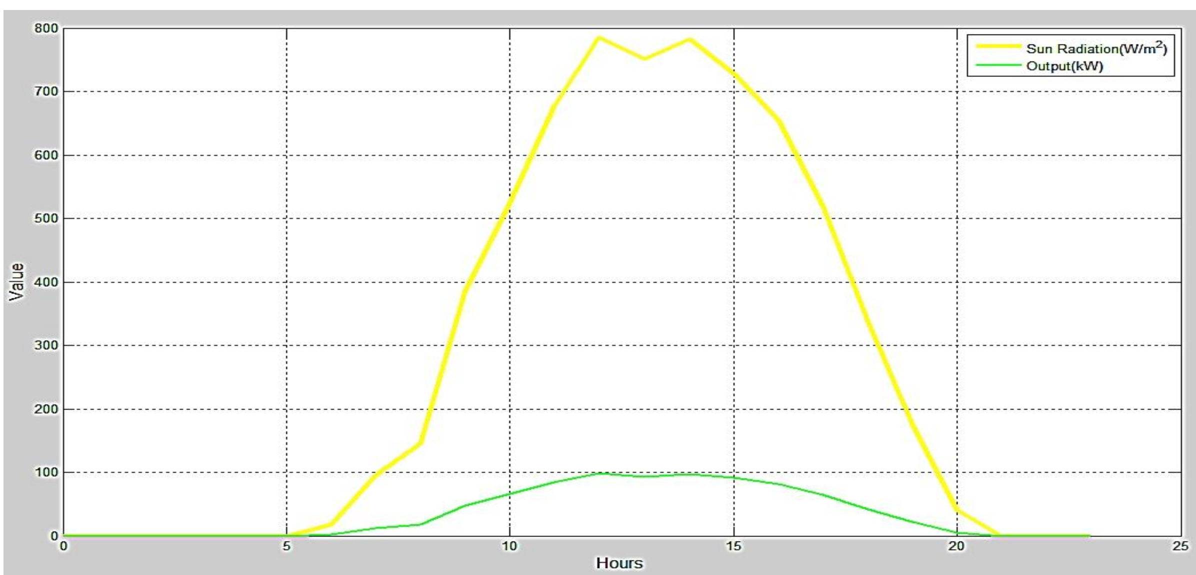


Figure 21. One day solar power plant output. Date: 2012-08-01

**Figure 6.** Demonstrates one day output of a solar power plant,which contains 500 solar panels with efficiency level 25 %

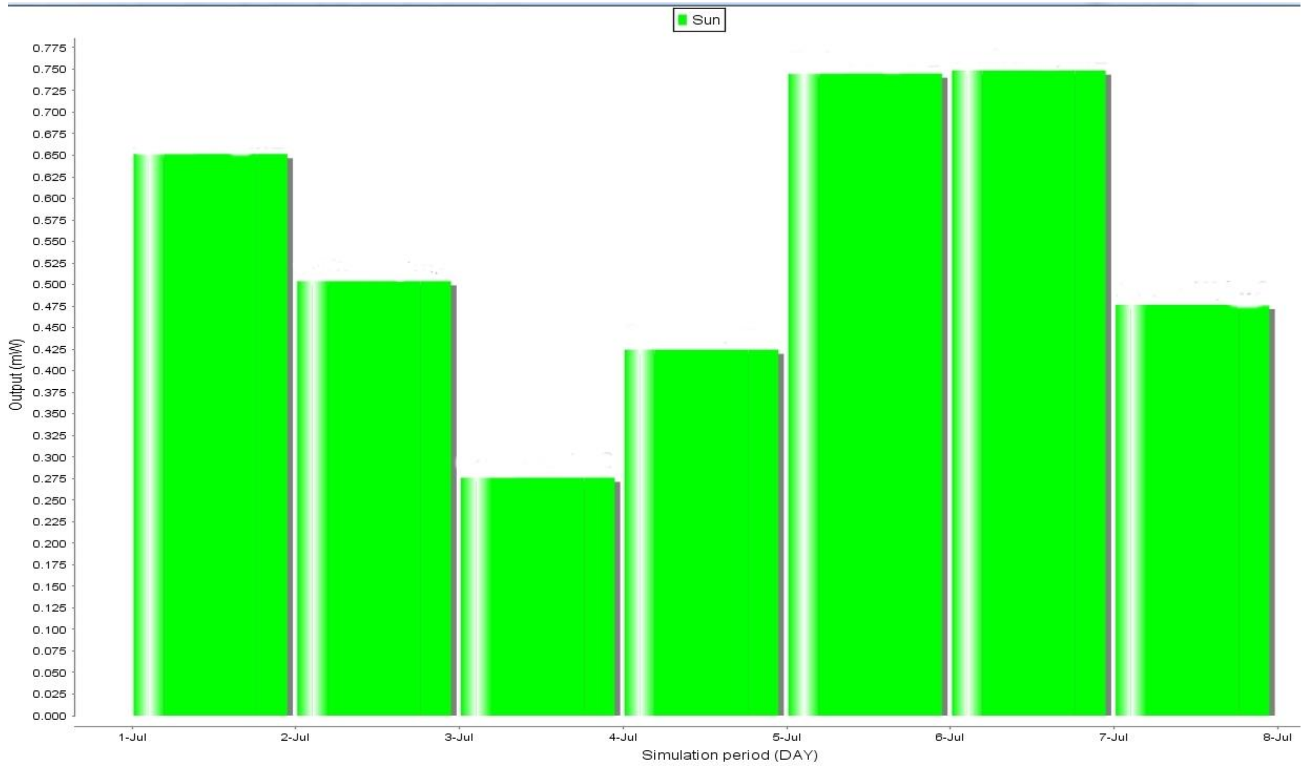


Figure 22. One week solar power plant output. Date: 2012-07-01 - 2012-07-07

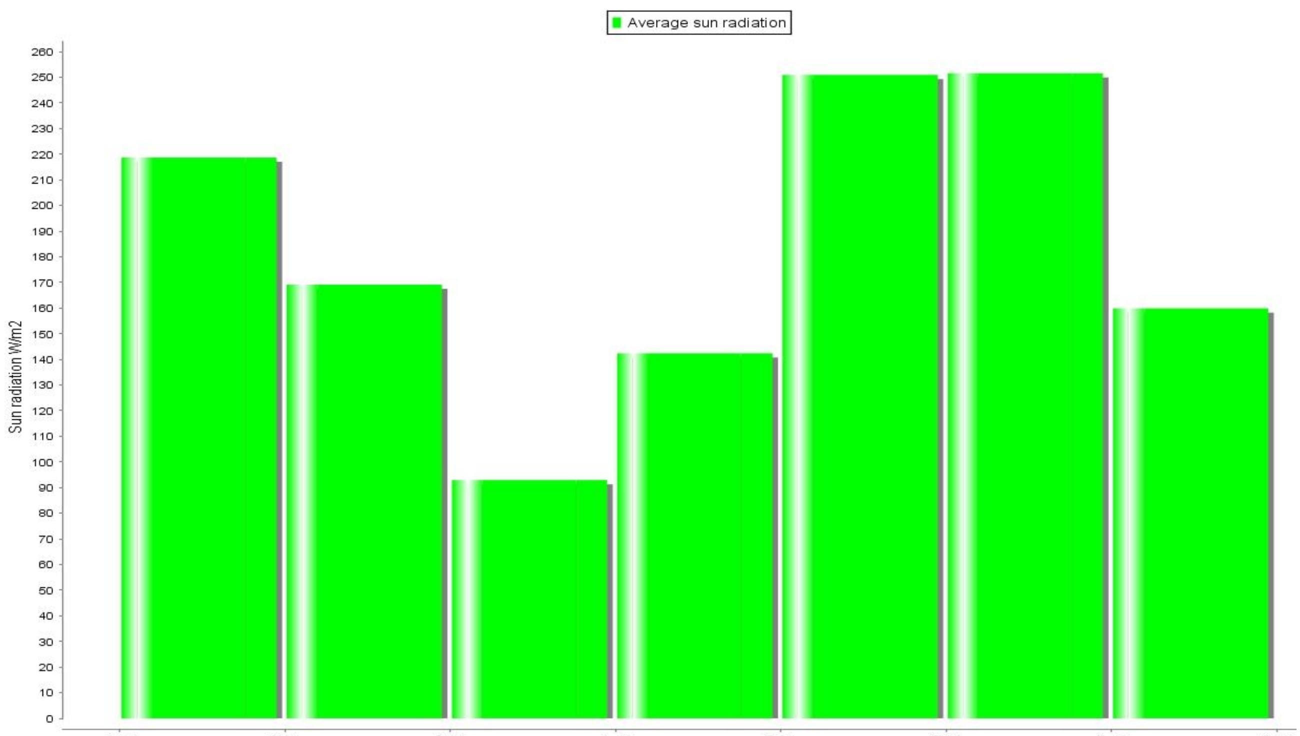


Figure 23. Average value of sun radiation during a week. Date: 2012-07-01 - 2012-07-07

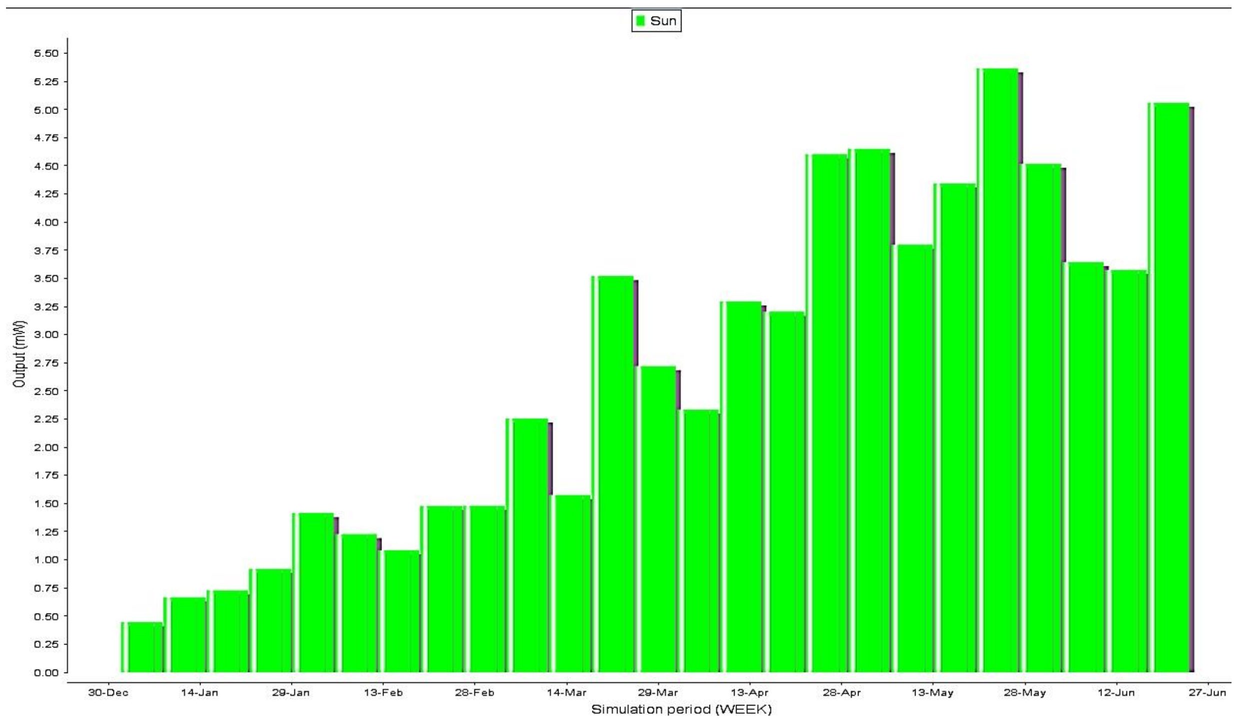


Figure 24. A half a year solar power plant output. Date: 2012-01-01 - 2012-07-01

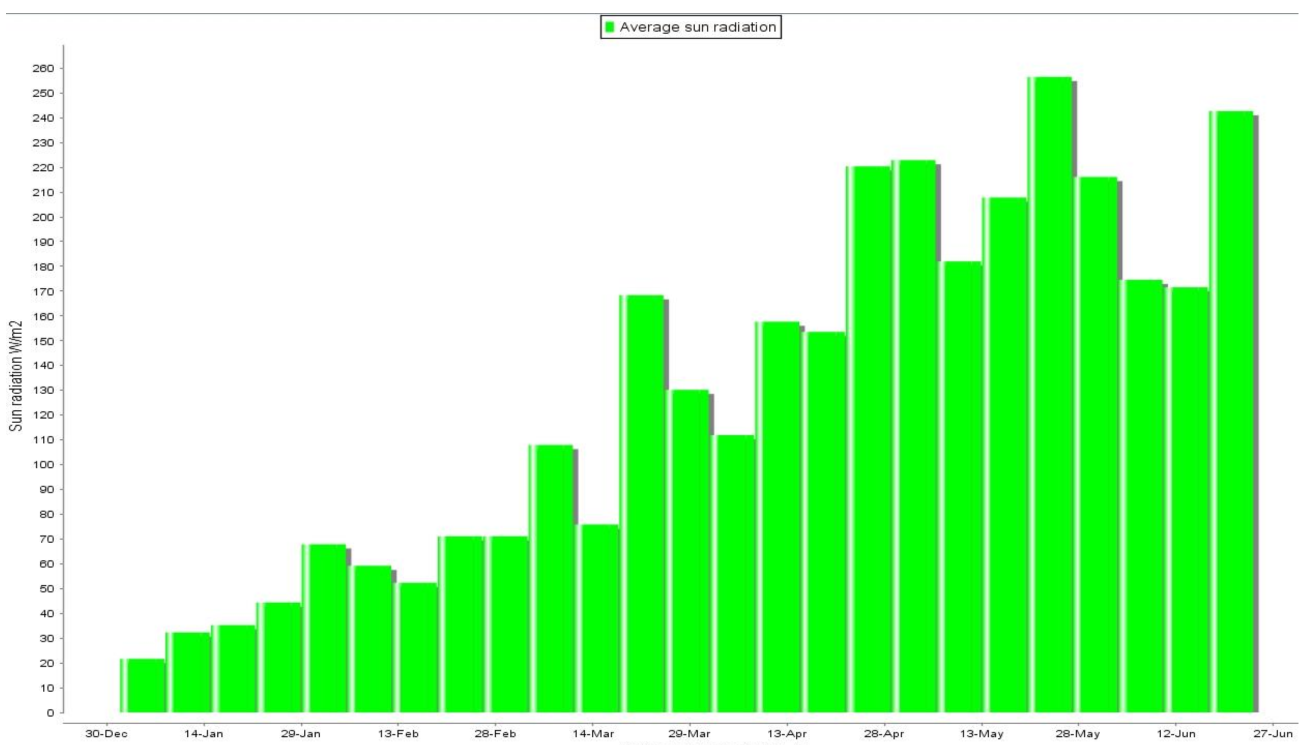


Figure 25. Average sun radiation during a half of a year. Date: 2012-07-01 - 2012-07-07

## 5.3 Building energy consumption simulation

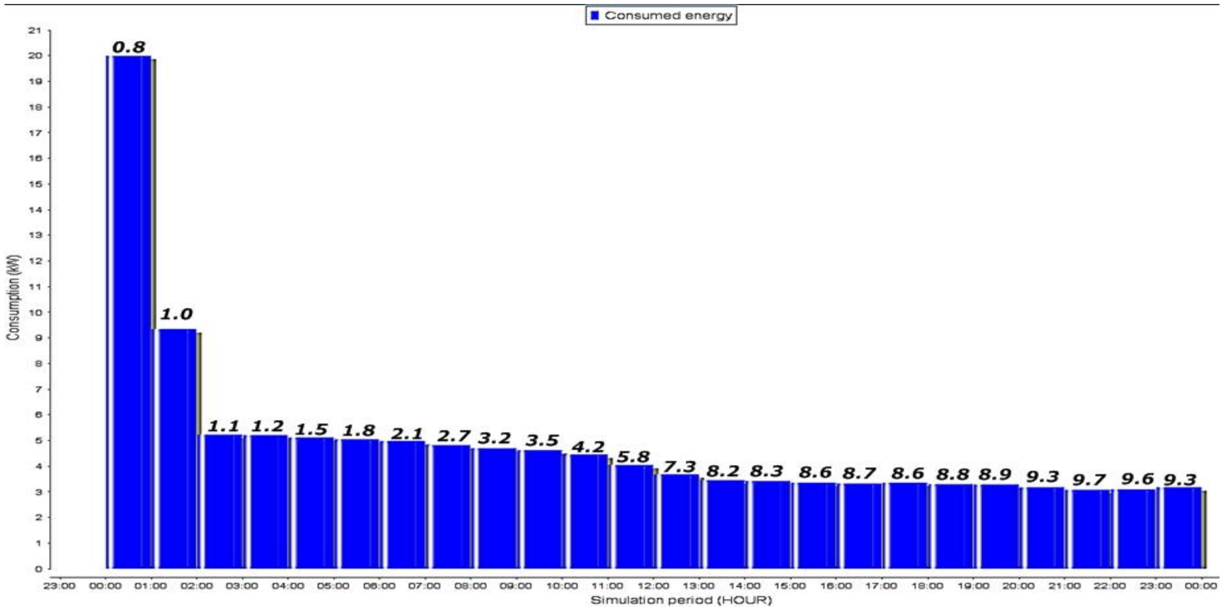


Figure 26.One day building energy consumption simulation. Date: 2012-01-01

**Figure 11** describes the need of electricity of one building during a day. One can notice that the very first stack is much bigger than others. It's caused by the fact that, the default temperature inside the building is 18℃ and desired temperature is 22℃. So the heating system needs more energy to rise the temperature inside to the desired level and then just support optimal level. There is a value of outside temperature above each stack for every hour of simulated period.

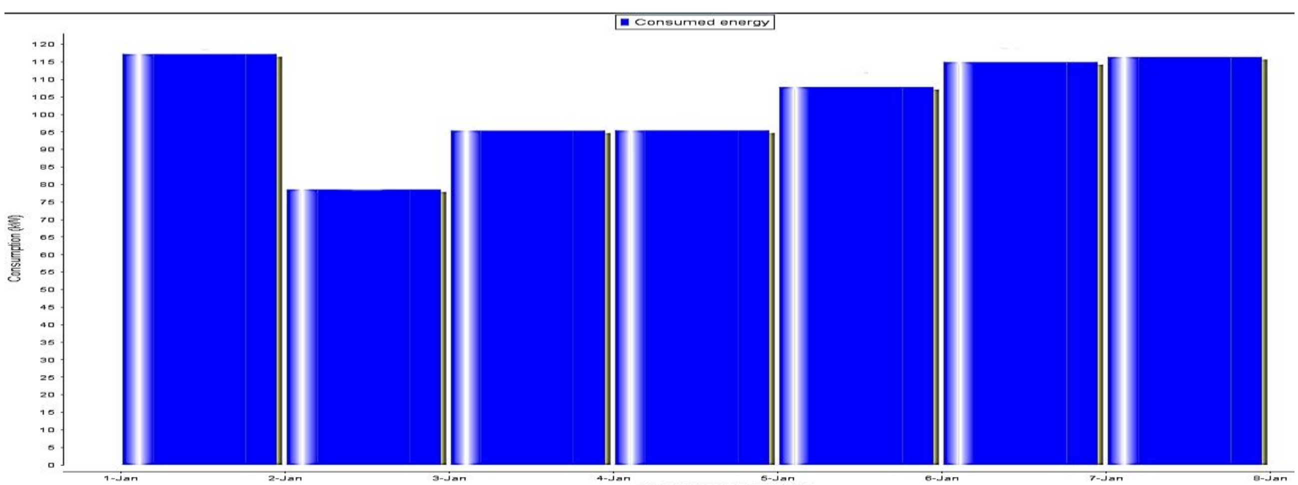


Figure 27. One week building energy consumption simulation. Date: 2012-01-01 - 2012-01-07

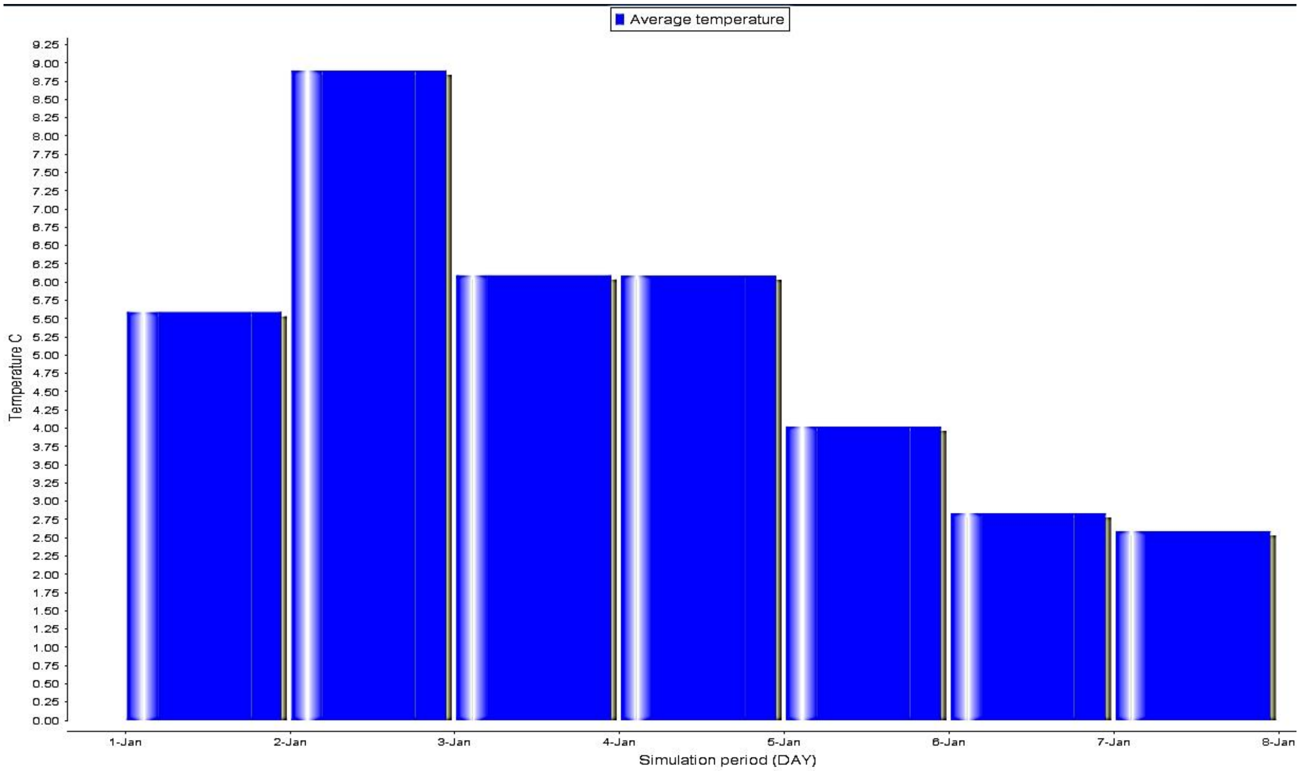


Figure 28. Average temperature during a week. Date:2012-01-01 - 2012-01-07

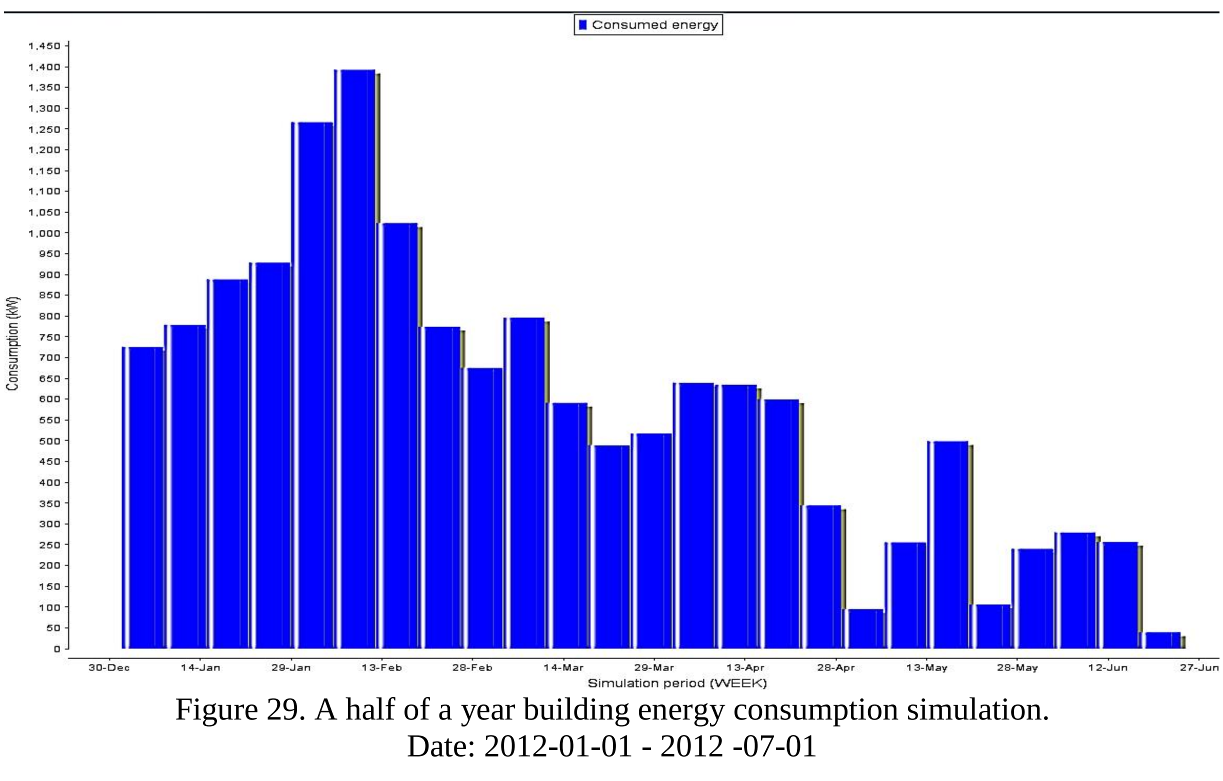


Figure 29. A half of a year building energy consumption simulation.
Date: 2012-01-01 - 2012 -07-01

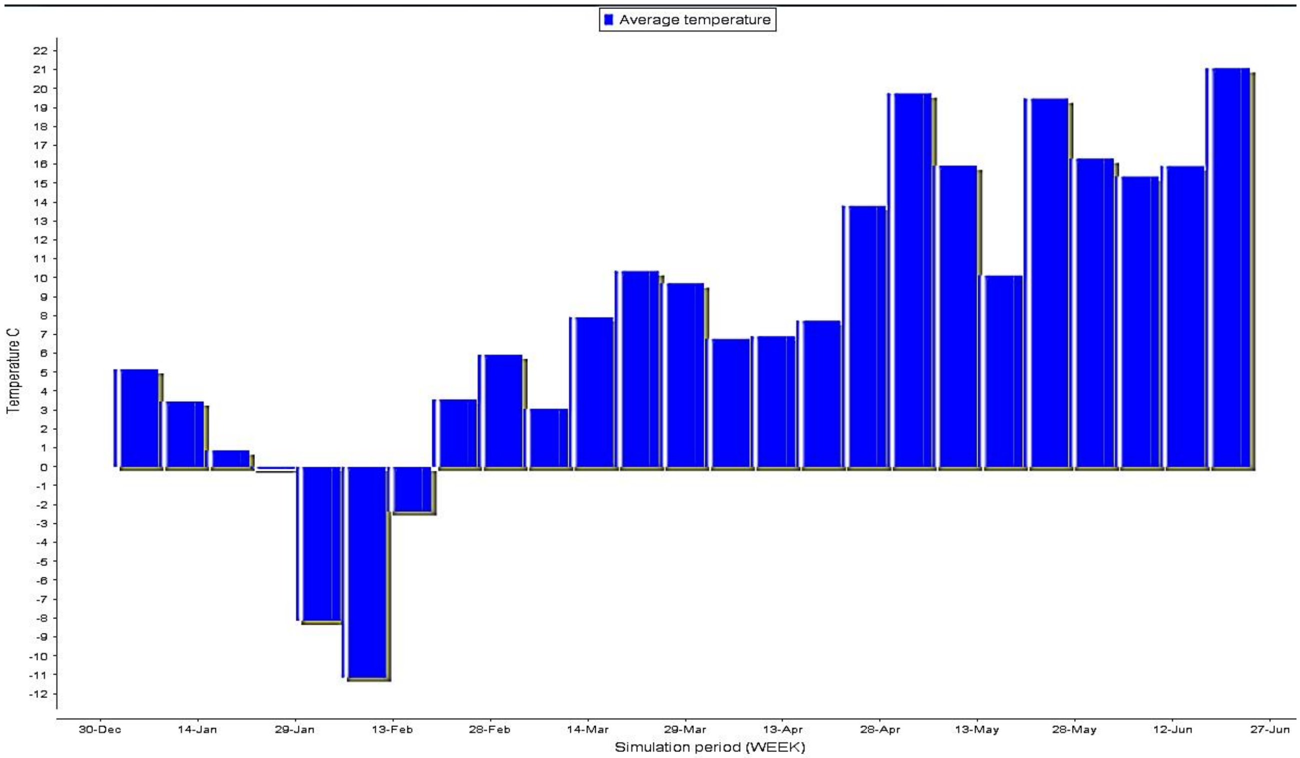

Figure 30. Average temperature during a half of a year. Date: 2012-01-01 - 2012-07-01

# Chapter 6

## Conclusions

Software for the simulation of power systems is very important and necessary tool for modeling, controlling and testing of power grids. With each passing day the relationship between objects of such power systems are becoming more complex and labyrinthine. In order to control and manage such systems, engineers definitely need reliable tools. After all, it is not just about software which is helping engineers to develop and test models of power systems, and maybe later some of these complex models of power grids will be destined to be realized in practice. Therefore, there is absolute necessary precision, reliability and correctness of calculations in such a piece of software, because exists some possibility that our comfort and well-being of our fast developing society may depend exactly on these calculations, but not only our development, lives of people as well.

Main goals of this project were successfully completed. Simulation of changes in weather conditions is based on real data from meteorological station. Day time changing, week and month changing meet all the parameters and features of the current calendar. While modeling the behavior of power plants were used scientific data and technical documentation of several existing pieces of equipment. The results of the simulation are available in section five. There is no doubt about the fact that it cannot be stated with certainty that all the calculations correspond to the real behavior of the objects involved in the simulation, but pretty close enough to get a clear idea about the properties and characteristics of such power grids. During the creation of the simulator wasn't taken into account the possible loss of electrical power, associated with the transportation and converting, features of the voltage and frequency of the electricity and possible use of appropriate transformers, switching, and control circuits. Necessary communication interface was designed, implemented and tested as well. It provides the ability for control multi-agent system to influence the course of the simulation, to get necessary updates on the present state of the power grid and parameters of the objects. Program out can help to imagine mechanisms and scenarios that occur during the simulation.

I had to meet with several types of power plants as alternative sources of energy. All of them have their own strengths and weaknesses. For example, if we talk about solar power plants, it is obvious that the maximum production of energy is possible only during the sunny day, but the power consumption is increasing in the evening as a rule. Changing weather causes change in intensity of energy production, which requires participation of additional expensive equipment. However, if to consider the solar generators, not as a primary, but as secondary power source, then surely it's a very good idea. In general, we can safely conclude that sooner or later, these technologies will meet the high expectations, because they are renewable power sources and sources of clean energy and we can't get away from the fact that this is quite important. If we are talking about wind turbines, then we should never forget that in different parts of our planet and different time the wind is blowing in different ways and with different power. During the construction of wind power plants require prior research and development wind maps. Wind farms are expected to be extended over large areas and it means significant distance from consumers, and this fact creates additional costs for energy transporting.

Trying to program this simulation engine, I realized that the capabilities and features of such software can be expanded to quite large frames. My program is just one small piece of a large high-end engine that can be created. There are prospects for improvement and additions to my program. For example, the way of communication with multi-agent system is very important thing and I believe that someday I will be able to improve my software, for example, as part of my future master's thesis. There are many ways to improve my project. Buildings may have not only heating, but also air-conditioning systems or so-called climate control systems, electric cars, smart household equipment and own sources of energy. I believe that every building can act like an independent member of electric grid and not only can play a role of "consumer".

# References


[1] Hasan Çimen, Emrah Oguz, Yüksel Oğuz, Harun Oğuz "*Power Flow Control of Isolated Wind-Solar Power Generation System for Educational Purposes*" Department of Electric&Electronic Engineering, Afyon Kocatepe University. IEEEXplore, Available from: <http://80.ieeexplore.ieee.org.dialog.cvut.cz/stamp/stamp.jsp?tp=&arnumber=636026>

[2] M. Yahyavi, M. Vaziri, S. Vadhva, "*Solar Energy in a Volume and Efficiency in Solar Power Generation*" IEEEXplore, Available from: <http://80.ieeexplore.ieee.org.dialog.cvut.cz/stamp/stamp.jsp?tp=&arnumber=555890>

[3] Gray Davis "A guide to photovoltaic (PV) system design and installation" consultation report June 14, 2001 Available from:< http://www.energy.ca.gov/reports/2001-09-04_500-01-020.PDF >

[4] Muljadi, E. ; Butterfield, C.P. ; Chacon, J. ; Romanowitz, H. Power Engineering Society General Meeting, 2006. IEEE Publication Year: 2006 Available from: <http://80.ieeexplore.ieee.org.dialog.cvut.cz/stamp/stamp.jsp?tp=&arnumber=170924>

[5] Hill, C.A. ; Such, M.C. ; Dongmei Chen ; Gonzalez, J. ;Grady, W.M. Smart Grid, IEEE Transactions on Volume: 3 , Issue: 2 Publication Year: 2012 , Page(s): 850 - 857 Available from: <http://80.ieeexplore.ieee.org.dialog.cvut.cz/stamp/stamp.jsp?tp=&arnumber=619874>

[6] Web: Electric Generation Using Natural Gas. Available at: <http://www.naturalgas.org/overview/uses_eletrical.asp>

[7] Marc-André Moffet ; Frédéric Sirois, Ph.D ; David Beauvais "Review of Open Source Power Grid Simulation Tools for Long-Term Parametric Simulations", July 11, 2011, Canada. Available from: <http://canmetenergy.nrcan.gc.ca/sites/canmetenergy.nrcan.gc.ca/files/files/pubs/201137_e.pdf>

[8] Web: DIgSILENT official web-site. Available from: < http://www.digsilent.de/index.php/products-powerfactory.html >

[9] Web: JFreeChart official web-site. Available from: <http://www.jfree.org/index.html>

[10] Web: Joda-Time official web-site. Available from: < http://jodatime.sourceforge.net/>

[11] Web: Working with the Java DB (Derby) Database. NetBeans official web-site. Available from: < https://netbeans.org/kb/docs/ide/java-db.html>

[12] The International Standard Atmosphere (ISA). Available from: <http://home.anadolu.edu.tr/~mcavcar/common/ISAweb.pdf >

[13] Solar cell efficiency. Wikipedia. Available from:
<http://en.wikipedia.org/wiki/Solar_cell_efficiency>

[14] "Suntech" producer of solar panels. Datasheet. Available from:
< http://www.suntech-power.com/images/stories/pdf/datasheets/en-GB/STP_285-305_Ve(poly)_EN_web.pdf>