CHAPTER 6

FSSAM: A Fuzzy Rule-Based System for Financial Decision Making in Real-Time

Penka V. Georgieva

This chapter looks into some problems financial managers face when they have to make decisions in real time while confronted with restrictions, such as coping with imprecise information or processing enormous amount of financial data. However, it is not concerned with existing software systems for supporting investment decisions, neither the ones based on fundamental analysis nor those based on technical analysis of stock markets.

What the chapter describes in detail is a real-time software application - Fuzzy Software System for Asset Management (*FSSAM*). *FSSAM* collects and processes the data autonomously, and produces outputs that support the process of financial management.

Penka V. Georgieva
Burgas Free University
Bulgaria
e-mail: penka.v.georgieva@hotmail.com

Editors: G.A. Papakostas, A.G. Hatzimichailidis, V.G. Kaburlasos, *Handbook of Fuzzy Sets Comparison - Theory, Algorithms and Applications*
DOI: 10.15579/gcsr.vol6.ch6, GCSR Vol. 6, pp. 121-148, 2016

Thus, the fuzzy rule-based systems (FRBS) are presented as a type of technology which provides tools for overcoming the above-mentioned difficulties, with its unique features, such as the capacity for implementing human knowledge; error tolerance and the ability to, relatively easily, create models of complex dynamic and non-deterministic systems with volatile and/or uncertain parameters.

Keywords - Fuzzy rule-based systems, approximate reasoning, decision making, FS-SAM

6.1 Introduction

Financial decision making in real time is a key concept in financial management. Traditionally, financial investors use mainly two methods when managing assets: fundamental analysis and/or technical analysis. While the former approach is based on exploring the numerous macroeconomic events, the latter is concerned with finding patterns in price time series. Practically, both approaches aim to predict the future behavior of asset prices and thus to make investment decisions.

However, following the dynamic development of macroeconomic parameters nowadays, new methods of financial analysis are emerging, aiming at taking into consideration the continuous fluctuations in world stock markets. Among them, FSSAM is considered to be able to offer new solutions to longstanding problems.

Most financial models are built on the assumption that asset returns have some type of probability distribution. However, empirical tests conducted on real data prove the opposite [9]. At the core of FSSAM are tools that fuzzy logic, as a basis of Soft Computing, provides for dealing with large amounts of data (such as time series representing asset prices) and sometimes vague or imprecise information (as is economic information). What is more, in fuzzy modelling there are no requirements for existence of probability distributions [2, 3, 7].

FSSAM steps on a simple concept: every investor has one ultimate goal and it is achieving maximum return at minimum risk. Therefore, the key point in the process of managing financial investments is finding a reliable estimator for changes in asset prices, an estimator that takes into account both return and risk. FSSAM is an independent software system in which the procedures for the collection and storage of data, the evaluation of assets and the construction of investment portfolios are implemented. It is a fuzzy rule-based software system, and following the general structure of a fuzzy system, consists of a knowledge base (rule base and database) and an inference machine.

Conceptually, fuzzy logic is the basis of Soft Computing (SC), enriched with neural networks, evolutionary computing, probabilistic calculations and conclusions [17, 21, 29].

SC consists of several computing paradigms [16, 21]: Fuzzy Logic (FL), Neural Networks (NN), approximate reasoning and non-differential optimization methods such as Genetic Algorithms (GA) and Simulated Annealing (SA) .

Historically, the idea of SC was proposed by Zadeh in 1991 in an attempt to create a new type of AI. And on 13.03.1991 at the Conference of the Industrial Liason Program in Berkeley, Berkley Initiative on Soft Computing (BISC) was established.

Soft Computing is a key stone of modern Artificial Intelligence (AI). In traditional computing (hard computing) main goals are accuracy, security and precision. On the contrary, the starting point for SC is the understanding that accuracy and security have a high price, and therefore tolerance for imprecision and uncertainty in calculations, reasoning and decision-making is admissible (when and where possible).

However, when discussing issues of fuzzy systems and soft computing, it is a must to outline the topic of AI as a branch of computer science. The term was first used by John McCarthy in 1956 as "the science of creating intelligent machines". Historically, AI originated from the attempts to formalize human knowledge using the tools of mathematical logic, and was first applied to theorem proofs and game modelling. At that point, the bases of AI were philosophy, mathematics, algorithms, logic, psychology, informatics, linguistics. Gradually, the traditional AI, mainly focused on imitation of human behavior in language form or symbolic rules, was being enriched with new ideas and thus the modern concept of AI developed.

Conventional AI operates mostly with symbols and is based on the assumption that intelligent behavior can be stored in symbolic-structured database. However, symbolic systems are a proper basis for human expertise modelling in certain narrow areas, provided that comprehensive knowledge is available.

At present, AI is a combination of traditional AI, enriched with different methodologies for numerical calculation and subjects of AI are mainly those problems for which no direct mathematical or logical algorithms exist or can be solved only intuitively. AI, alongside with the initial basic sciences, includes some new ones: neuroscience, cognitive science, ontology, operations research, economics, probability, and optimization.

The fields of application of AI are extremely wide: expert systems, games, theorem proof, natural language processing, pattern recognition, robotics, navigation, control systems, planning systems, data mining, logistics and many others.

This chapter addresses the following topics: Fuzzy Modelling and Fuzzy Systems; Types of Fuzzy Systems, Fuzzy Rule-based Software Systems, and Fuzzy Software System for Asset Management (FSSAM). Experimental results are presented and conclusions are derived.

6.2 Fuzzy Modelling and Fuzzy Systems

The following are the main concepts in fuzzy logic: linguistic variables, linguistic modifiers, propositional fuzzy logic, deductive inference rules, and approximations. Zadeh proposed the concept of linguistic variables for modelling human way of thinking, based on the following principle: "With the increasing complexity of a system, our ability to make accurate and at the same time rational conclusions about systems' behavior is reduced until it reaches a threshold after which the accuracy and consistency are almost mutually exclusive requirements" [2].

When designing a fuzzy software system for decision-making, a key factor is the extent to which this system will be able to mimic the behavior of a previously known real system [1, 21].

The process of creating a fuzzy system, called *fuzzy modelling*, has to follow two important requirements:

(1) The structure of the fuzzy system has to be designed so that the experience of the experts can be easily implemented in it.

(2) In case the input and output data are known, the possibility for identifying the system using standard techniques has to exist [20].

There are two required stages in fuzzy modelling:

(1) Identification of the overall structure. In this stage own knowledge (common sense, laws of physics, etc.), information from experts or information obtained after "trials and errors" is used. The consecutive steps are:

  (1.1) Selecting appropriate input and output variables.

  (1.2) Choosing the type of fuzzy inference system.

  (1.3) Determining the number of linguistic terms of the input and the output fuzzy variables.

  (1.4) Designing the set of if-then rules.

(2) Identification of the structure in depth. Here a detailed description of linguistic terms is made and this includes:

  (2.1) choosing the family of membership functions;

  (2.2) choosing the values of the parameters for each of the membership functions;

  (2.3) adjusting these values.

The use of fuzzy logic in building models of systems for decision-making has several significant advantages:

(1) fuzzy logic is easy to implement because the used mathematical concepts are relatively simple;

(2) a fuzzy system can be further developed without having to design it again, because adding new rules and features does not change its structure;

(3) fuzzy logic is tolerable to ambiguity in the information, i.e. it is possible to use incorrect, incomplete and vague information;

(4) using fuzzy logic provides the opportunity for modelling nonlinear functions of arbitrary complexity;

(5) a fuzzy system can be built using experts' knowledge;

(6) fuzzy logic can be combined with standard control techniques and, although not necessarily, fuzzy systems can replace conventional control techniques, but more importantly - in many cases fuzzy systems are more easily implemented;

(7) fuzzy logic is based on natural languages.

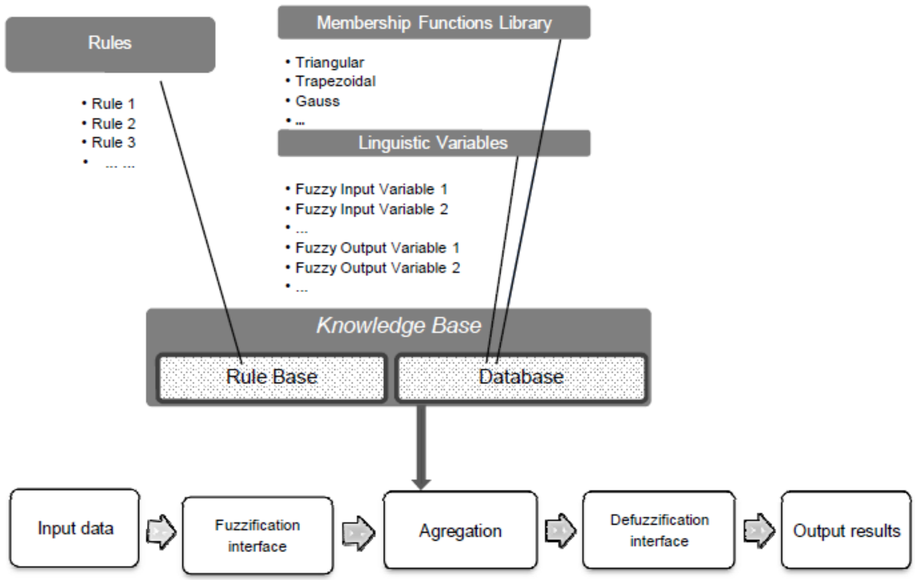


Figure 6.1: Rule-based fuzzy system.

6.3 Types of Fuzzy Systems

Fuzzy inference systems (FIS) are computational structures based on the theory of fuzzy sets, if-then rules and fuzzy logic. Since fuzzy systems vary in structure and purpose, different names such as *fuzzy expert system*, *fuzzy model*, *fuzzy associative memory*, *fuzzy logic controller*, *fuzzy system* and others are used.

The general structure of a fuzzy system has three conceptual components:

(1) rule base, including all fuzzy rules for decision-making;

(2) database, where all the membership functions, all terms used in linguistic variables and fuzzy rules of the fuzzy system are defined;

(3) inference machine, performing the procedure for deriving conclusions from given and known rules and facts.

The rule base and the database form the knowledge base (Fig.(6.1)).

A fuzzy rule-based system operates in the following sequence:

(i) the input data are fuzzified in order to obtain membership degrees to each of the terms of the input fuzzy variables;

(ii) the inference machine applies the aggregation rules, using the knowledge base and thus membership degrees to the terms of output variables are calculated;

(iii) after defuzzification, the output result is obtained (Fig.(6.1)) [27].

6.3.1 Mamdani-type Fuzzy Systems

In 1975 Ebrahim Mamdani proposed a fuzzy system for managing a steam engine and a boiler in which the linguistic rules were adapted to the human experience of the operators [23], and this was among the first systems based on fuzzy set theory. A Mamdani-type fuzzy system differs from other fuzzy systems in type of output variables – in a Mamdani-type fuzzy system the output variables are fuzzy variables that have to be defuzzified by various methods.

Each operator in the inference machine of a Mamdani-type fuzzy system corresponds to a norm or conorm:

- a *T-norm* is used for the AND operator;
- a *T-conorm* is used for the OR operator;
- a *T-norm* with given weights for the rules is used for the implication operator;
- a *T-conorm* is used for the aggregation operator;
- different approaches to transform the output fuzzy variable into a crisp value are used for the defuzzification operator.

For the *T-norms* and *T-conorms* two options are mostly used:

- min operator for the *T-norm* and max operator for the *T-conorm* and a maxmin operator for the composition;
- algebraic product for the *T-norm* and max operator for the *T-conorm* with a max product for the composition.

More details on *T-norms* and *T-conorms* can be found in literature, for instance in [10].

Other operators and compositions produce different variations of the model, but Mamdani-type fuzzy models have one thing in common - the output variables are fuzzy sets and therefore defuzzification is need. There are different methods for defuzzification of the aggregated output fuzzy variables. If $A = \{x, \mu_A(x)\,|x \in X\}$ is a fuzzy variable, then the defuzzified value $\hat{x}$ is calculated as follows:

- *Centre of gravity method*

$$\hat{x} = x_{CoG} = \frac{\int_X \mu_A(x)\, x \partial x}{\int_X \mu_A(x)\, \partial x}.$$

- *Median method*

$$\hat{x} = x_{BoA},$$

where x_{BoA} is such that $\int_{\alpha}^{X_{BoA}} \mu_A(x)\, \partial x = \int_{X_{BoA}}^{\beta} \mu_A(x)\, \partial x$ for $\alpha = \min\{x\,|x \in X\}$ and $\beta = \max\{x\,|x \in X\}$.

- *Average of maxima method*

$$\hat{x} = x_{AoM} = \frac{\int_{X'} x \partial x}{\int_{X'} \partial x},$$

where $X' = \{x \,|\mu_A(x) = \mu^* \,:\, \mu^* = \max_X(\mu_A(x))\}$.

- *Min of maxima method*

$$\hat{x} = x_{LoM} = \min\left\{x \,|\mu_A(x) = \mu^* \,:\, \mu^* = \max_X(\mu_A(x))\right\}.$$

- *Max of maxima method*

$$\hat{x} = x_{MoM} = \max\left\{x \,|\mu_A(x) = \mu^* \,:\, \mu^* = \max_X(\mu_A(x))\right\}.$$

6.3.2 Sugeno-type Fuzzy Systems

This type of model was proposed by Takagi, Sugeno and Kang as an attempt to create a systematic approach for generating fuzzy rules from a given set of input and output data.

The fuzzy rules in a Sugeno-type model have the form:

if x is A and y is B, then $z = f(x; y)$,

where A and B are fuzzy variables and $f(x; y)$ is a real function.

In case $z = f(x; y)$ is a constant function, the model is called a zero order Sugeno model and the output is a singleton; if it is a first order polynomial - a first order Sugeno model.

The Sugeno-type fuzzy systems are suitable for managing a set of linear controllers, as well as for managing dynamic non-linear systems. The Sugeno-type fuzzy systems are particularly good in cases with small continuous changes, applied on the input space and for modelling nonlinear systems by multiple linear components[28].

6.3.3 Tsukamoto-type Fuzzy Systems

In a Tsukamoto-type fuzzy model, the output variable consists of fuzzy sets with monotonous membership functions. After the execution of each fuzzy rule a crisp value, induced by the weight of the implementation of this rule, is obtained. Finally, the output is a crisp value which is calculated as a weighed average of the outputs of each rule. In this model defuzzification is not necessary.

6.4 Fuzzy Rule-based Software Systems

Following the general structure of a fuzzy system, a fuzzy rule-based software system consists of a knowledge base (rule base and database) and an inference machine (Fig.(6.2)).

Let N be the number of the input fuzzy variables $K_i, i = 1, 2, 3, \ldots, N$, and n_i be the number of terms X_ij of K_i for each i with $j = 1, 2, 3, \ldots, n_i$. Let S be the number of output fuzzy variables $Q_s, s = 1, 2, 3, \ldots, S$, and p_s be the number of terms Y_{sp} of Q_s for each s with $p = 1, 2, 3, \ldots, p_s$.

Let $\mu_i j(x)$ be the membership function of the term $X_i j$ and $\mu_{sp}(y)$ be the membership function of Y_{sp}. Then the overall number of the membership functions in the knowledge base is

$$N \times \sum_{i=1}^{N} n_i + S \times \sum_{s=1}^{S} p_s.$$

The crisp input values form a vector $x^* = (x_1^*, x_2^*, \ldots, x_N^*)$. This vector is fuzzified by calculating $\mu_i j(x_i^*)$ for each i and j. At this point there are

$$N \times \sum_{i=1}^{N} n_i$$

membership values, stored in the database after that calculation.

The next step is to aggregate. For simplicity let *min* operator be used for the *T-norm* and *T-norm* be used for the *AND* operator. Let M be the number of rules and the m-th rule R_m has the form:

if $\left\{K_{m_1} is\, X_{m_1 j_{m_1}}\right\}$ and $\left\{K_{m_2} is\, X_{m_2 j_{m_2}}\right\}$ and ... and $\left\{K_{m_k} is\, X_{m_k j_{m_k}}\right\}$ then $\left\{Q_{m_1} is\, Y_{m_1 j_{m_1}}\right\}$ and $\left\{Q_{m_2} is\, Y_{m_2 j_{m_2}}\right\}$ and ... and $\left\{Q_{m_k} is\, Y_{m_k j_{m_k}}\right\}$

and each rule has its weight w_m, $m = 1, 2, 3, \ldots, M$.

Once the m-th rule is selected and put into the template (Fig.(6.2)), two consecutive calculations are made:

(i) $\Theta_m = \min\left\{\mu_{m_1 j_{m_1}}\left(x_1^*\right), \mu_{m_2 j_{m_2}}\left(x_2^*\right), ..., \mu_{m_k j_{m_k}}\left(x_k^*\right)\right\}$and then

(ii) $\Theta_m^o = \Theta_m w_m$.

After firing all the rules the corresponding values of the membership functions $\mu_{sp}^m = \Theta_m^o$ for each term Y_{sp} of the output variables are obtained. The number of these values depends on the number of rules in which they are used.

The aggregation applies after calculating

$$P_{sp} = \max\left\{\mu_{sp}^1, \mu_{sp}^2, ..., \mu_{sp}^M\right\}$$

for each Y_{sp}, $s = 1, 2, 3, \ldots, S$ and $p = 1, 2, 3, \ldots, p_s$.

The last step is defuzzification. For implementing any of the methods for defuzzification, shown earlier, a numerical integration could be applied.

This procedure is illustrated on Fig.(6.2), whith the main actions being:

(1) selector activation, (2) rule choice, (3) template, (4) rule activation, (5) go to: Fuzzy variables, fuzzy aggregation, defuzzification, (6) interface connection, (7) reading from the database, (8) processing the next rule, (9) writing the results in the database and (10) output.

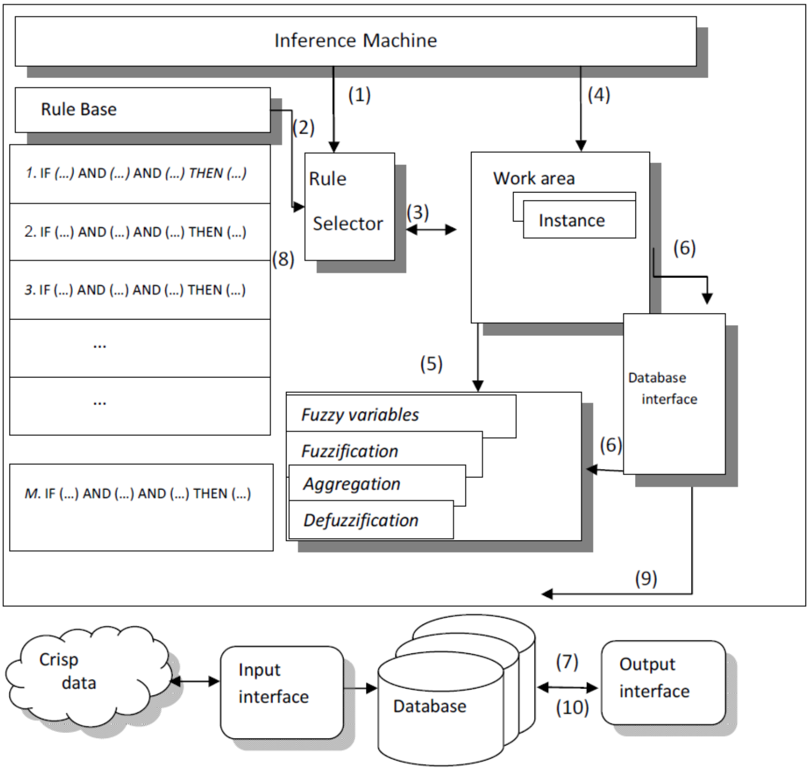


Figure 6.2: Inference machine.

6.5 Fuzzy Software System for Asset Management (FSSAM)

FSSAM is an independent software system which consists of procedures for data collection and data storage, asset evaluation and investment portfolios construction.

The application software system consists of three modules (Fig.(6.3)):

(1) *Data managing module (DMM)* with the following features: automatically submits queries to the Web server of a particular stock exchange; extracts data from the downloaded pages; writes data to the database; fills in the missing data; calculates return, risk and q-ratio for each asset in the database.

(2) *Q-measure fuzzy logic module (QFLM)*, which is an application, based on fuzzy logic. Input data are the crisp numerical values of asset characteristics, obtained from *DMM*. These crisp values are fuzzified and after applying the aggregation rules, a fuzzy variable (*Q-measure*) for each of the assets is derived. The output is a defuzzified crisp value of *Q-measure*. The linguistic variables are four: three input variables and one output variable. Input variables describe the characteristics of an asset: $K1 = \{return\}$, $K2 = \{risk\}$ and $K3 = \{q - ratio\}$. The output variable is $Q = \{Q - measure\}$. The input variables $K1 = \{return\}$, $K2 = \{risk\}$ consist of five terms, each with corresponding parameters: *Very Low*, *Low*, *Neutral*, *High*, and Very High. $K3$ consists of three terms: *Small*, *Neutral*, and *Big*. The output variable Q consists of five terms: *Bad*, *Not Good*, *Neutral*, *Good* and *Very Good*. There are 24 fuzzy rules implemented in the system. All fuzzy rules in this module have the form:
IF $\{K1\, is\, High\}$ AND $\{K2\, is\, Low\}$ AND $\{K3\, is\, Big\}$ THEN $(Q\, is\, Good)$.
As a defuzzification method, the method of center of gravity has been chosen, the composite trapezoidal rule for numerical calculation of the integrals is used and thus a crisp value for the asset quality is obtained as an output of *QFLM*.

(3) *Portfolio construction module (PCM)*, in which various portfolios are constructed.

The modules are described in detail in the next section.

6.5.1 Data Managing Module (DMM)

The first module is a Data managing module (DMM) and it is an application for collecting, storing and managing financial data in real time from the web page of the Bulgarian Stock Exchange [4]. In addition, in this module calculations of precise measurements of important asset characteristics (return, risk and q-ratio) are carried out. It consists of *Requester*, *Parser*, *Filler* and *Calculator*.

DMM consists of a two-layer application and a database (Fig.(6.4)). The main function of this module is to collect raw data (asset prices) online and to store and process the data.

The tests of the system are conducted with data from the Bulgarian Stock Exchange. The changes in Bulgarian asset prices are published on the webpage page of BSE [4].

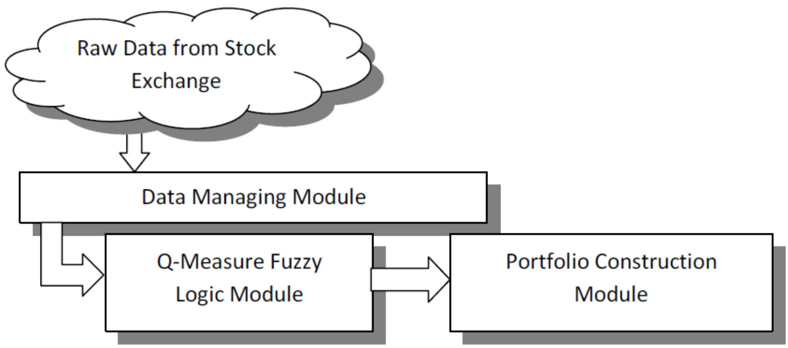


Figure 6.3: Conceptual scheme of FSSAM.

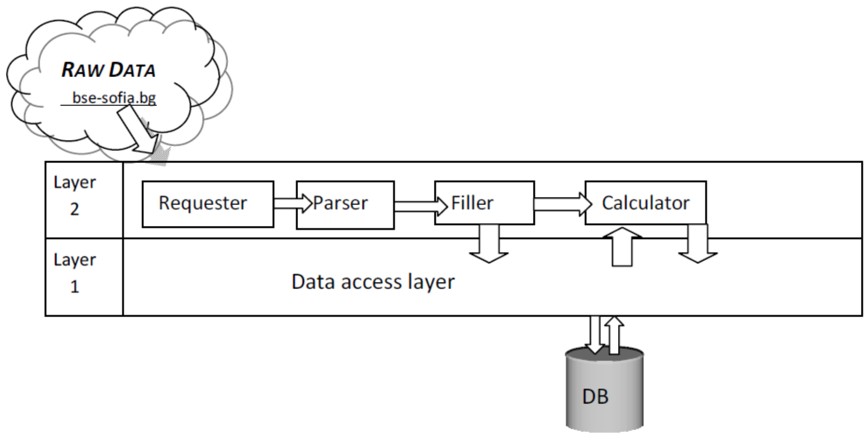


Figure 6.4: A scheme of the Data Managing Module.

The information of interest – date, BSE code, open, close, high and low prices - is in the html code of this page.

The application is started automatically.

The data access layer contains functions for managing the database.

The **Requester** realizes the request to the page of BSE:

```
public class Requester
{ public static string BSEPageRequest() {... return result;}
  public static string BSEPageRequest(DateTime date)
    { ... return result; }
  public static string BSEChartRequest(string code)
    { ... return result; }
}
```

The **Parser** selects the data that is needed in a form suitable for the next steps. For this reason the html code of the downloaded page by the **Requester** is parsed with regular expressions:

```
"<td[^>]*>([a-zA-Z0-9%-\\. ]*?)</td>|<td[^>]*><a .*>(.*?)</a></td>"
"<em>   (?(Bonds Segment)|.*?)</em>.*?<table[^>]*>(.*?)</table>"
```

The **Filler** is the part of the application that deals with the missing data. There are two types of missing data – missing name and missing price. The name of an asset is missing in the database if it is a new asset listed on the exchange or there was no trade with this asset for a period of time longer than the time the system is working. The price could be missing on the days with no deals with the given asset or on the holidays.

Here is a part of the source code of the **Filler**:

```
public class Filler
{ public static void CheckEquity(List<EquityModel> equityList)
     {... }
   public static void FillMissingEquities(List<EquityModel> equities)
     {... }
   public static void CopyEquityDatas(List<EquityModel> equities)
     {... }
}
```

The function *CheckEquity* checks the existence of the name of the asset after parsing the page in the database after the last parsing of the page. If it exists the last price is added to the list with prices of this asset, if not – the *FillMissingEquities* adds its name to the list of assets and then starts to fill in the list with its prices. *CopyEquityDatas* fills in the database with the last available price for the asset if it is already in the database and not at exchange page.

The **Calculator** uses the information from the **Filler** and derived below formulae (1), (2) and (3) for calculating the return, risk and r/R ratio for each asset.

Return, Risk and r/R Ratio of an asset

Let $P_1, P_2, \ldots, P_T$ be the sequence of daily prices P_t of an asset A, $t = 1, 2, \ldots, T$. Then the geometric mean of returns is an accurate measure for the change of the invested sum:

$$R_g = \sqrt[T-1]{\prod_{t=2}^{T} r_t},$$

where $r_t = \frac{P_t}{P_{t-1}}$ is the *return* for day t, $t = 1, 2, \ldots, T$.

Nevertheless, if one needs to study the dynamics of price changes and the standard deviation of the returns, it is appropriate to take the logarithms of returns and then the next formulae are derived:

$$ln\,(r_t) = ln\left(\frac{P_t}{P_{t-1}}\right) = ln\,(P_t) - ln\,(P_{t-1})\,,$$

which is called *log return* for day t, $t = 1, 2, \dots, T$ and

$$\bar{r_g} = ln\left(R_g\right) = ln\left(\sqrt[T-1]{\prod_{t=2}^{T} r_t}\right) = \frac{1}{T-1}\sum_{t=2}^{T} ln\left(r_t\right),$$

which is the arithmetic mean of log returns.

In time series formed from prices, some data may be missing, e.g. there may have not been any trading activities. One way to compensate these missing values is to add the last non-missing price the corresponding number of times $P_{t-1}, P_{t-1}, \dots, P_{t-1}, P_t$.

Then the annual return is calculated as follows:

$$AR = \sqrt[s]{\prod_{t=2}^{T} r_t} = \sqrt[s]{\frac{P_T}{P_1}},$$

where $s = \frac{\sum_{t=2}^{T} \Delta_t}{D}$; Δ_t is the number of days between the non-missing observations at days $t-1$ and t decreased by 1 and D is the number of days in the financial year.

And the mean annual norm of return is $ANR = AR - 1 = \sqrt[s]{TR} - 1$.

In case one uses log returns, the above considerations should be made very carefully because of the different number of days between the observations.

Thus, if the return for the period Δ_t is $r_t^* = \frac{r_t - 1}{\Delta_t} + 1$, then the log return at the moment t is $ln\left(r_t^*\right)$ and so the arithmetic mean of log returns is calculated as:

$$\bar{r_g^*} = ln\left(AR\right) = ln\left(\sqrt[T-1]{\prod_{t=2}^{T} r_t^*}\right) = \frac{1}{T-1} \times \sum_{t=2}^{T} ln\left(r_t^*\right) \tag{6.1}$$

Moreover, $AR^* = (e^{\bar{r_g^*}} - 1) \times D$ is the annual return and $ANR^* = (e^{\bar{r_g^*}} - 1) \times D - 1$ is the annual norm of return [5, 9].

The annual norm of return ANR^* is an adequate estimator for the exact annual return of the asset.

The commonly used measure of risk in investment theory is the variability of returns. The variability shows to what extent returns change over time and thus estimates the probability of gain or loss in future moment. The variability is calculated by different statistical tools, based on probability distributions and most often the variance of the returns [5, 14, 15, 19, 22, 24, 25].

If log returns are used, then the estimator of variance as an arithmetic mean of log-returns is calculated as follows:

$$s^2 = \frac{1}{T-2} \times \sum_{t=2}^{T} \left(ln\left(r_t\right) - \bar{r_g}\right) \tag{6.2}$$

and the r/R ratio [11] equals the quotient of return and *risk*:

$$q = \frac{\bar{r_g^*}}{s}. \tag{6.3}$$

6.5.2 Q-measure Fuzzy Logic Module (QFLM)

The final goal in the process of decision making is to find an optimal solution for a situation in which a number of possible solutions exists. Bellman and Zadeh proposed a fuzzy model for decision-making in which objectives and goals are described as fuzzy sets and the solution is an adequate aggregation of these sets. There are various algorithms for building a fuzzy system [2, 6, 7, 18, 20].

The fuzzy system proposed in this chapter is particularly designed and built with respect to its financial application.

QFLM is an application based on fuzzy logic. Input data for this module are the crisp numerical values of asset characteristics from DMM. These crisp values are fuzzified and after applying the aggregation rules a fuzzy variable *Q-measure* for each of the assets is obtained. The output of this module is a defuzzified crisp value of *Q-measure*.

6.5.2.1 Input variables of QFLM

The calculations of the crisp values of the input variables: annual return, risk and q-ratio are derived in DMM. These crisp values are fuzzified with the predefined linguistic variables (LVs). The output variable is one: *Q-measure* of an asset. The definitions and notation from [18] are being followed in describing the LVs of QFLM.

The names of LVs are $X_1 \triangleq return$, $X_2 \triangleq Risk$, $X_3 \triangleq q-ratio$, $Y \triangleq Q-measure$. The term-sets of LVs are $T(X_1) = \{X_{1j}\}$, $T(X_2) = \{X_{2j}\}$, $T(X_3) = \{X_{3k}\}$, $T(Y) = \{Y_j\}$ for $j = 1, ..., 5$; $k = 1, 2, 3$ and

$$X_{ij} \triangleq \begin{pmatrix} Very\,Low & i=1,2 & j=1 \\ Low & i=1,2 & j=2 \\ Neutral & i=1,2 & j=3 \\ High & i=1,2 & j=4 \\ Very\,High & i=1,2 & j=5 \\ Small & i=3 & j=1 \\ Neutral & i=3 & j=2 \\ Big & i=3 & j=3 \end{pmatrix}, \quad Y_j \triangleq \begin{pmatrix} Bad & j=1 \\ Not\,Bad & j=2 \\ Neutral & j=3 \\ Good & j=4 \\ Very\,Good & j=5 \end{pmatrix}.$$

The universes of discourse of LVs are $U_{X1} = U_{X2} = U_{X3} = U_Y = R$. Three types of membership functions are used:

- Gaussian membership function $\mu_G(x) = e^{-\frac{1}{2}\left(\frac{x-\beta}{\alpha}\right)^2}$.
- Bell membership function $\mu_B(x) = \frac{1}{1+\left|\frac{x-\gamma}{\alpha}\right|^{2\beta}}$.
- Sigmoid membership function $\mu_S(x) = \frac{1}{1+e^{-\alpha(x-\beta)}}$.

The corresponding type of membership functions (MF) and values of the parameters are shown on Table 6.1.

For each input variable a degree of membership to the corresponding term is calculated.

Table 6.1: Type and parameters of membership functions of terms.

Terms	MF	α	β	γ
X_{11}	$\mu_S(x)$	-20	0	-
X_{12}	$\mu_G(x)$	0.05	0	-
X_{13}	$\mu_G(x)$	0.08	1.1	-
X_{14}	$\mu_G(x)$	0.05	1.2	-
X_{15}	$\mu_S(x)$	20	1.3	-
X_{21}	$\mu_S(x)$	-2	0	-
X_{22}	$\mu_G(x)$	0.05	0.1	-
X_{23}	$\mu_G(x)$	0.07	0.3	-
X_{24}	$\mu_G(x)$	0.05	0.5	-
X_{25}	$\mu_S(x)$	2	0.7	-
X_{31}	$\mu_S(x)$	-0.3	20	-
X_{32}	$\mu_B(x)$	20	4	40
X_{33}	$\mu_G(x)$	0.3	60	-
Y_1	$\mu_G(x)$	0.1	0	-
Y_2	$\mu_G(x)$	0.1	0.25	-
Y_3	$\mu_G(x)$	0.1	0.5	-
Y_4	$\mu_G(x)$	0.1	0.75	-
Y_5	$\mu_G(x)$	0.1	1	-

6.5.2.2 Fuzzy inference

In this application, a Mamdani-type fuzzy inference (MFIS) system is chosen. As a result of MFIS, a fuzzy output is obtained and this is the major reason for which MFIS are widely used in decision support applications. There are four stages in the fuzzy inference process:

(1) Evaluation of the antecedent for each rule.

(2) Obtaining a conclusion for each rule.

(3) Aggregation of all conclusions.

(4) Defuzzifying.

The *AND* and *THEN* operators are implemented by min fuzzy *T-norm*, whereas the aggregation is implemented by *max* fuzzy *T-conorm*. Center of gravity method is used for defuzzification of the output.

As there are three input variables with 5, 5 and 3 terms accordingly, the universe of all possible rules consists of 75 rules. In the system, 24 of the rules are chosen by experts. Although these rules adequately describe the most important possible situations that might arise in the process of investment decision-making, the list of fuzzy rules can be extended without changing the system's architecture. The fuzzy rules model the decision making process intuitively and have IF-THEN form:

Table 6.2: Fuzzy rules for decision making.

No	return	Risk	q-ratio	Q-measure	Weight
1	Very High	Very Low	Big	Very Good	1
2	Very High	Low	Big	Very Good	1
3	High	Very Low	Big	Very Good	1
4	High	Low	Big	Very Good	1
5	Very High	Very Low	Big	Good	0.8
...	...	...	...	...	...
24	Very Low	High	Neutral	Not Good	0.8

if $(r^{*} is\, X_{1i})$ and $(s^{*} is\, X_{2j})$ and $(q^{*} is\, X_{3k})$ then $(Q - measure\, is\, Y_p)$
for $i = 1, ..., 5$; $j = 1, ..., 5$; $k = 1, ..., 3$ and $p = 1, ..., 5$.

Some of the rules that are implemented (with their respective weights) are shown in Table 6.2.

In the next step the fuzzy rules are fired. At this point additional expert knowledge is taken into account by assigning weights to each rule in the structure. In this way for a crisp input $(r^{*},\, s^{*},\, q^{*})$ the obtained membership values are:

$$\theta^{*} = \min \left\{ \mu_i \left(r^{*} \right), \mu_i \left(s^{*} \right), \mu_i \left(q^{*} \right) \right\}$$

and then respectively

$$\theta^{**} = w \times \theta^{*}$$

where w is the corresponding weight of the rule. This procedure for the seventh rule in FLQM is illustrated on Fig.(6.5).

After applying all the rules, several values for each term of the output variable *Q-measure* are calculated. Aggregation is the process of bringing together the outcomes of all the fuzzy rules. Choosing a suitable aggregation operator is a key issue when a fuzzy system is designed. As an aggregation method the max fuzzy *T-conorm* is applied in the proposed model and thus the fuzzy output variable is obtained (Fig.(6.5)).

The overall fuzzy output generally constitutes a multimodal non-zero distribution of possible crisp values over a subset of the output space. In the defuzzification stage, one of those possible crisp values has to be selected. The design of a sound defuzzication method is important as it affects the interpretation of the fuzzy response. A desirable defuzzification procedure should require a low computational effort to allow its implementation in real-time applications. At the same time, it should allow a smooth response and mapping accuracy to be obtained over all or most of the output space. What is more, a defuzzification method should ease the design of the fuzzy system and keep the decision making logic transparent to the user.

The center of gravity (*CoG*) has been chosen as a defuzzification method. According to it, the crisp output value is calculated as:

$$\hat{Q} = CoG\left(Q\right) = \frac{\int_{-\infty}^{+\infty} xQ\left(x\right) \partial x}{\int_{-\infty}^{+\infty} Q\left(x\right) \partial x}$$

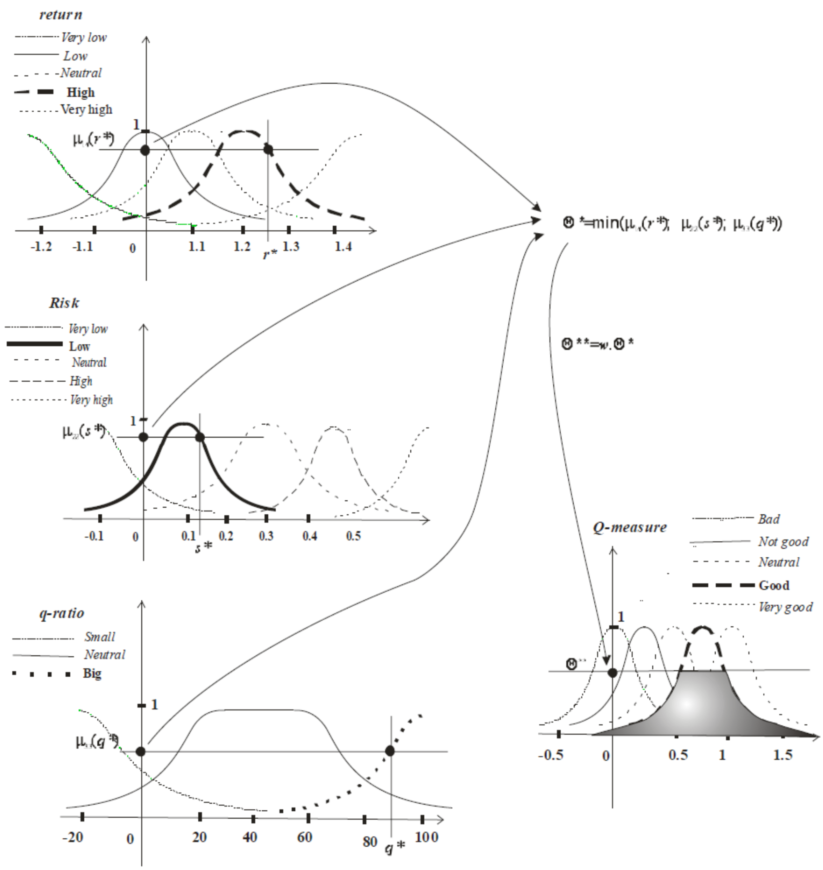


Figure 6.5: Firing rule number 7: IF (return is high) AND (Risk is low) AND (q-ratio is big) THEN (Q-measure is good) .

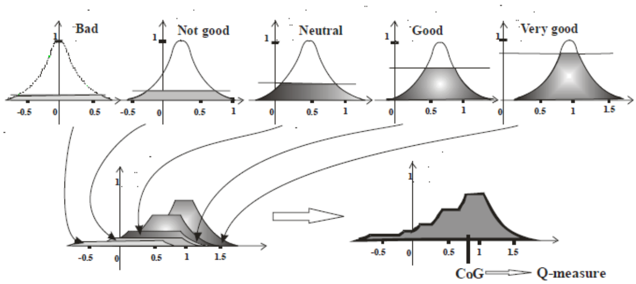


Figure 6.6: Aggregation and defuzzification for obtaining the *Q-measure* of an asset.

which is illustrated in Fig.(6.6).

6.5.3 Portfolio Construction Module (PCM)

The third and last module of the system is the Portfolio construction module. In PCM several portfolios are constructed and the investors' utility preferences are the key factor for choosing the optimal portfolio.

First, all assets in the database are sorted in descending order by their *Q-measure*. Then, if the investor would like to hold a portfolio with no more than k assets, the first k assets from the sorted list are taken. Let $A = \{A_1, A_2, A_3, \ldots, A_k\}$ be the set of the top k financial assets. Now all possible combinations with $1, 2, \ldots, k$ elements are constructed and recorded. The number of these combinations is $2^k - 1$, because the empty set is not taken into consideration. Next, for each combination of assets a portfolio is constructed in the following manner.

Let x_j be the share of asset A_j. Then according to x_j is calculated as:

$$x_j = \frac{Q_j}{\sum_{j=1}^{k_0} Q_j},$$

where k_0 is the number of financial assets in the particular portfolio, $k_0 = 1, 2, \ldots, k$ and Q_j is the *Q-measure* of A_j, obtained in QFLM.

Next, the three characteristics R_p, σ_p and q_p of the portfolio are calculated for all $2^k - 1$ portfolios according to the formulas:

$$R_p = \sum_{j=1}^{n} x_j \times r_j, \;\; \sigma_p = \sum_{j=1}^{n} x_j \times s_j, \;\; q_p = \frac{R_p}{\sigma_p}.$$

Finally, each portfolio is put through the *Q-measure* Fuzzy Logic Module of the software application in order to obtain the portfolios' *Q-measure*[26]. The portfolios with their characteristics are stored in the database.

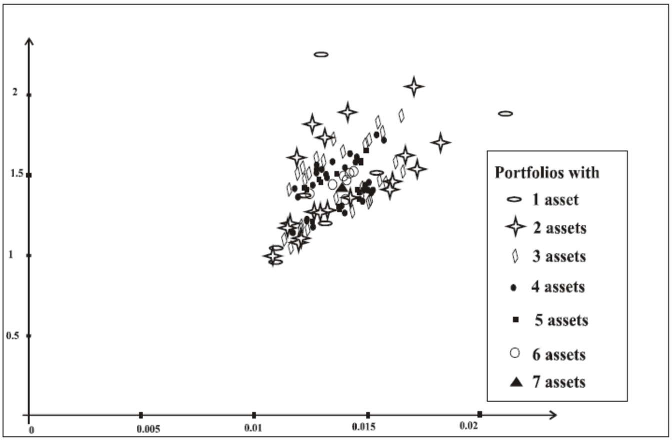


Figure 6.7: Disposition of 127 possible portfolios.

6.6 Applications of FSSAM

6.6.1 Individual financial asset management

In investment management, the most important point is gaining high profit with lowest possible risk. However, for the non-speculative investor it is essential to what extent these two characteristics (return and risk) are stable over time. FSSAM is built on one additional characteristic: the q-ratio, which is the quotient of return and risk and reflects the degree to which the taken risk is justified by adequate returns. The conducted empirical tests for all assets listed on BSE for different periods of time, show that the *Q-measure* is a proper indicator of the quality of the asset over time. If the *Q-measure* is less than 0.4 (whatever return and risk) a dramatic decrease of price occurs in up to about 3 months. At the same horizon and a *Q-measure* between 0.4 and 0.6 the price of the asset does not change significantly and even if it increases, the transaction costs will exceed the potential benefits. When the *Q-measure* is greater than 0.6 the asset price increases steadily and such an asset is considered suitable for purchase. Detailed results for individual asset management are published in [10].

6.6.2 Portfolio management

For illustration, 127 possible portfolios derived from FSSAM are presented in Fig.(6.7). On a particular day, the software system detected seven assets from BSE with high *Q-measures* (above 0.8): 5BN, 6AS, 6A6, 5BD, 4EC, 3JR and 5ALB. If all these assets are used, the portfolio constructing module creates 127 possible portfolios. The total investment capital is fixed to BGN $150,000$ and all the portfolios have *Q-measure* above 0.8. Each of these portfolios is represented as a point on the graph shown in Fig.(6.7), investment risk being plotted on the x axis, and the return being plotted on the y axis.

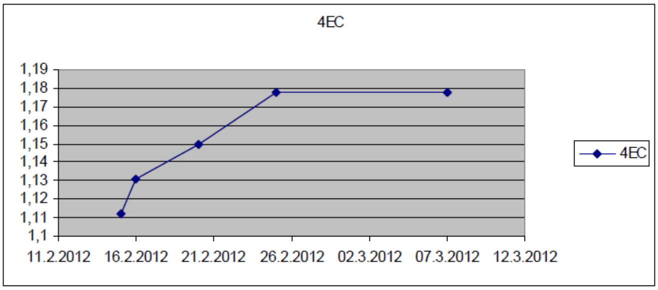


Figure 6.8: Change of price of 4EC in 1, 5, 10 and 21 days after 15.02.2012 (data from Table 6.5).

Investors can choose any portfolio depending on their preferences for return, risk or number of included shares.

6.7 Results

In this section, as an illustration of FSSAM, some results obtained from real data are presented. The data are from the Bulgarian Stock exchange and the currency used is BGN. The constructed portfolios are tracked over time and compared with portfolios obtained with traditional techniques.

6.7.1 Individual assets

The application is used to assess all assets stored in the data base. The results of the evaluation of 51 assets listed on the Bulgarian Stock Exchange on 15.02.2012 are shown in Table 6.3.

As can be seen from this table, the system returns the characteristics of each of the 51 assets that are listed in the data base.

The performance over time of one of these assets is analyzed below.

6.7.1.1 Performance of 4EC

Table 6.4 shows the results for return, risk, annual rate of return, return/risk ratio and *Q-measure* of 4EC. The system was tested for various periods back from the selected date 15.02.2012: 10 days. 25 days. 50 days. 100 days. 120 days. 150 days. 200 days. For asset 4EC the *Q-measure* for all periods is greater than 0.75, which indicates a steady increase in its price.

The system is designed so that the *Q-measure* can be used to predict the changes in the price of an individual asset. Table 6.5 and Fig.(6.8) present the asset price of 4EC for different periods of time: in 1 day, 5 days, 10 days, 21 days after 15.02.2012, and Fig.(6.9) shows a one-year change - from 28.06.2011 to 20.06.2012.

The table and the graphs clearly show that the asset price of 4EC rises. The results show that the calculated *Q-measure* meets the requirements of the system to capture

Table 6.3: Results after evaluation of 51 assets.

Asset code	Return	Risk	Annual norm of return	Return/Risk	Q-measure
3JR	0.86588	0.00878	0.59763	68.0317	0.34158
4CF	0.80388	0.01906	0.41165	21.6024	0.19791
4EH	0.87871	0.02594	0.63614	24.5232	0.20375
5F4	0.84348	0.02959	0.53043	17.9287	0.20056
6C4	0.70206	0.02366	0.10619	4.48764	0.19297
E4A	0.86022	0.01301	0.58065	44.6162	0.28087
1VX	1.17391	0.01598	1.52174	95.2483	0.80900
3JU	0.98535	0.01714	0.95606	55.7954	0.34062
3MZ	0.9636	0.03284	0.89081	27.1293	0.23761
3NB	0.88569	0.02745	0.65706	23.9397	0.20406
3NJ	0.92778	0.03047	0.78333	25.7124	0.21761
3ZL	0.57388	0.03674	-0.2784	-7.5773	0.18919
4BJ	0.92946	0.01994	0.78837	39.5334	0.28443
4I8	0.75111	0.03454	0.25333	7.33382	0.19471
4ID	0.91796	0.02987	0.75389	25.2389	0.21195
4IN	0.73468	0.05748	0.20403	3.54992	0.19380
4L4	1.3125	0.0228	1.9375	84.9622	0.82888
4O1	0.77039	0.04901	0.31116	6.34955	0.19614
52E	0.77956	0.03154	0.33867	10.7391	0.19668
53B	0.76792	0.03119	0.30375	9.73956	0.19594
55B	0.87291	0.01459	0.61872	42.4134	0.27763
57B	1.02665	0.02625	1.07994	41.1446	0.34086
5BN	1.09127	0.01054	1.2738	120.84	0.81218
5IC	0.67961	0.0351	0.03883	1.10655	0.19220
5MH	0.86707	0.01985	0.6012	30.2933	0.25051
5ODE	0.92487	0.02239	0.77462	34.5974	0.27467
5ORG	1	0	1	0	0.19943
5OTZ	0.60976	0.03989	-0.1707	-4.2797	0.19051
5SR	0.7616	0.0267	0.2848	10.6649	0.19536
5V2	1.17043	0.03182	1.51129	47.5021	0.51089
6A8	0.94975	0.00983	0.84925	86.4039	0.78095
6A9	1	0.03503	1	28.5484	0.26337
6AB	0.91379	0.00709	0.74138	104.603	0.80937
6C4P	0.94717	0.02696	0.8415	31.2182	0.26877
6L1	0.69	0.03249	0.07	2.15438	0.19252
6S5	1.30435	0.0211	1.91304	90.6738	0.81145
6S7	0.66667	0.04533	1.554312234e-015	3.428996082e-014	0.19178
AO0	1.17958	0.04112	1.53874	37.4181	0.50135
C81	0.71667	0.3248	0.15	0.46182	0.20175
E4AP	0.9986	0.01688	0.99581	59.0087	0.35244
G0A	1.13636	0.01319	1.40909	106.855	0.81151
SO5	1.0582	0.01477	1.1746	79.5393	0.75408
4EC	1.72136	0.02378	3.16409	133.044	0.81247
5BD	1.0679	0.03434	1.2037	35.0483	0.36046
5BU	1.22139	0.02292	1.66418	72.6153	0.68653
5H4	1.15873	0.00812	1.47619	181.686	0.81245
6A6	1.12038	0.01152	1.36113	118.107	0.81215
6BMA	0.80769	0.03925	0.42308	10.7795	0.19841
6F3	1.00442	0.03394	1.01327	29.8562	0.27900
BLKC	0.96061	0.02072	0.88182	42.5645	0.30396
ZNOA	1.38056	0.07691	2.14167	27.8465	0.68352

Table 6.4: Changes of the characteristics of 4EC over different periods after the initial date 15.02.2012.

4EC					
Number of days	**Return**	**Risk**	**Annual norm of return**	**Return/Risk**	**Q-measure**
10	1	0.00582262	1	171.7439757	0.81244408
25	1.17921527	0.02157357	3.509013786	162.6533453	0.812479437
50	1.28703703	0.01905535	3.009259259	157.9219874	0.812481545
100	1.73208722	0.025468852	3.196261682	125.4968862	0.812469577
120	1.72136222	0.023782192	3.164086687	133.0443653	0.812474403
150	1.60926193	0.022748406	2.218523878	97.52436543	0.811988257
200	1.63770250	0.021201632	1.637702504	77.24417096	0.755784754

Table 6.5: Prices of 4EC in 1, 5, 10 and 21 days after 15.02.2012.

Period	**Price**
1	1.112
5	1.131
10	1.150
21	1.178

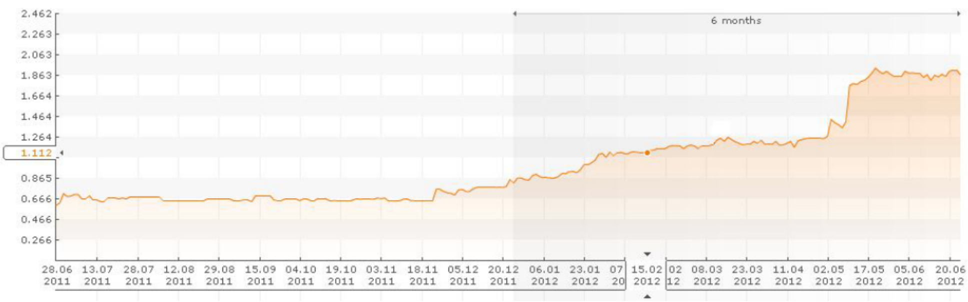


Figure 6.9: One-year graph of prices of 4EC.

price changes. The detailed results in [10, 12, 13] definitely prove that FSSAM is a reliable tool for detecting future performance of asset prices.

6.7.2 Portfolio construction

First, some results from 20.06.2012 are presented. The portfolio, constructed with FSSAM, consists of the 10 best assets and the initial investment capital is $100,000$ BGN. The initial characteristics of this portfolio are:

$$R_p = 1.4652573;\ s_p = 0.0185322\ and\ Q = 0.7583479.$$

It is important to point out that the unused capital is 64.58 BGN. The change in the invested capital over the next 3 months is shown in Table 6.6.

On the first day (21.06.2012), the unused capital was 64.58 BGN, which was 0.064568% of the total sum. After 15 days (02.07.2012), the price of the portfolio increased by 717.26 BGN up to $100,652.68$ BGN. So the rate of return for the period is 0.72% and the annual rate of return was 17.23%. After one month (16.07.2012), the price of the portfolio increased by a total of $3,109.13$ BGN, e.g. the rate of return for the period was 3.11% and 37.33% per annum. After 2 months (15.08.2012), the price of the portfolio investment had already risen by $12,700.27$ BGN, which was 12.71% rate of return for the period and 76.25% for the year. After 3 months (14.09.2012), the price increased by $18,926.86$ BGN, meaning that the rate of return for the period was 18.94% and 75.76% for the year.

To check the effectiveness of FSSAM on the same date (21.06.2012) two more investment portfolios are constructed based on Markowitz model, using the procedure described in [8]. The invested amount of money was again $10,000$ BGN. The asset shares and the changes over time are shown in Table 6.7.

The results are illustrated in Fig.(6.10). For the period from 21.06.2012 to 10.07.2012 portfolio yield obtained from FSSAM was less than the yield of both portfolios generated model of Markowitz. Then, from 11.07.2012 to 20.08.2012 the return from the two portfolios from Markowitz model significantly decreased and even became negative, while the yield of the FSSAM portfolio showed clear growth.

In the next period, the portfolios obtained with the model of Markowitz, began gradually to increase profitability. The period for which the FSSAM system was set to give reliable results (positive return) was 2 months. As seen from the comparison, results obtained from FSSAM completely satisfy this requirement.

The second example shows portfolios that were constructed in similar way on 05.03.2015 under the conditions of the FSSAM model as well as Markowitz model.

After the initial portfolios construction, the asset prices are observed, and the corresponding capital K is calculated as a sum of the asset price multiplied by its share in the portfolio as in example one. And Profit is the difference between the initial capital ($100,000$ BGN) and the portfolio value, given it was sold on that date. As demonstrated in Table 6.8 and on Fig.(6.11), Portfolio FSSAM again showed not only greater returns, but much more stable behaviour in the selected quarterly interval as well.

Table 6.6: Three-month change of a portfolio, constructed with the 10 best assets on 20.06.2012 using FSSAM.

Name	Number of assets	Share in portfolio	20.06.2012		02.07.2012		16.07.2012		15.08.2012		14.09.2012	
			Price	Total	Price	Total	Price	Total	Price	Total	Price	Total
3JU	204	0.10196994	50.00	10200.00	45.012	9182.45	45.012	9182.45	49.00	9996.00	60.00	12240.00
5ORG	111	0.09991100	90.00	9990.00	90.00	9990.00	90.00	9990.00	81.00	8991.00	81.00	8991.00
6A6	5742	0.09991006	1.74	9991.08	1.783	10237.99	1.81	10393.02	2.018	11587.36	2.02	11598.84
BLKC	22300	0.09990622	0.448	9990.40	0.48	10704.00	0.53	11819.00	0.705	15721.50	0.685	15275.50
4EC	5400	0.09990151	1.85	9990.00	1.91	10314.00	1.91	10314.00	1.989	10740.60	2.02	10908.00
SO5	5708	0.09990047	1.75	9989.00	1.875	10702.50	1.705	9732.14	1.705	9732.14	1.754	10011.83
5BD	13228	0.09987639	0.755	9987.14	0.738	9762.26	0.798	10555.94	0.88	11640.64	0.956	12645.97
5BN	2588	0.09963701	3.85	9963.80	4.07	10533.16	4.50	11646.00	4.75	12293.00	5.28	13664.64
57B	211	0.09951558	47.00	9917.00	48.60	10254.60	47.50	10022.50	52.00	10972.00	59.00	12449.00
55B	211	0.09947182	47.00	9917.00	42.52	8971.72	44.50	9389.50	51.95	10961.45	52.50	11077.50
total		**1**		**99935.42**		**100652.68**		**112635.69**		**112635.69**		**118862.28**
difference				**-64.58**		**717.26**		**3109.13**		**12700.27**		**18926.86**

Table 6.7: Portfolios constructed with Markowitz model.

Portfolio 1				Portfolio 2			
Name	**Share in portfolio**	**Price**	**Total**	**Name**	**Share in portfolio**	**Price**	**Total**
4BJ	0.0929	1	9290	4BJ	0.0515	1	5150
4PX	0.0929	24.101	54396	4PX	0.5473	24.101	54709
C81	0.3629	0.249	36290	C81	0.4011	0.249	40110
total	**1**		**99976**		**0.9999**		**99969**
difference			**-24**				**-31**

Table 6.8: Portfolios performance over 3-month period (05.03.2015-18.06.2015).

	05.03.2015	**19.03.2015**	**02.04.2015**	**06.05.2015**	**03.06.2015**
Portf_1					
K	99959	100099	100389	100947	103737
Profit	-41	99	389	947	3737
Portf_2					
K	99966	100022	97523	100305	101683
Profit	-34	22	-2477	305	1683
Portf_FSSAM					
K	99947	100106	101494	107663	112245
Profit	-53	106	1494	7663	12245

6.8 Conclusion and Future Development

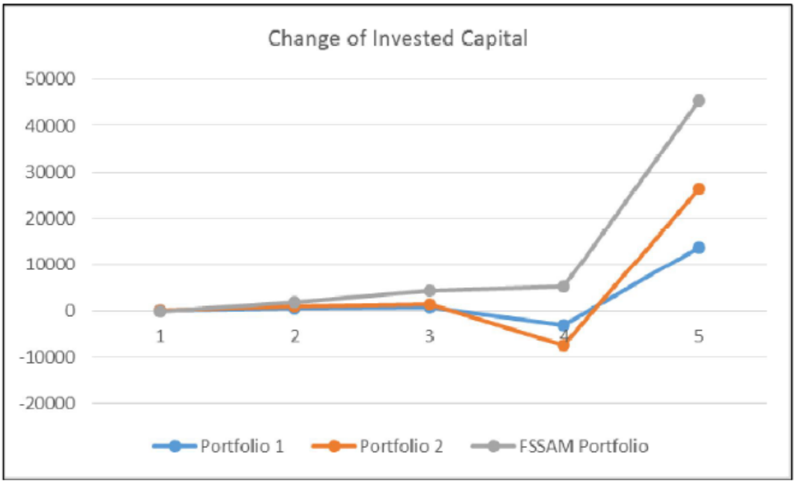
In this chapter the general description and requirements for designing and creating a decision support system based on fuzzy logic are presented.

Fuzzy Software System for Asset Management (FSSAM) is described and results are shown. FSSAM is designed and used for assessing financial assets – individual as well as financial portfolio investments. This model is based on the *Q-measure* of an asset: a characteristic which combines return, risk and their ratio, and being modelled with fuzzy logic tools, it intuitively reflects the process of investment decisions in economic environment with an enormous amount of data, which is often incomplete and imprecise.

A major difference from existing models is that there are no requirements for probability distributions of returns as empirical tests on real data show absence of such distributions.

The created software system is used for conducting tests on real data from BSE. Some of the results are presented in this chapter. First, the system calculates the *Q-measure* of every asset on BSE and this quantity can be used in the process of investment decision. In addition, the system executes a procedure for portfolio allocation which allows the investors to base their decision on financial market information, provided by the model and on their personal preferences. One advantage is that as an output, the investor can choose between several portfolios that differ in number of assets, return and risk but still show high *Q-measures*.

Although the realized software application shows very good results, the fuzzy systems have one disadvantage in general: they are not flexible to changes. There are various directions for the future improvement of the model: adjusting the parameters of the membership function with a neural network; expanding the number of fuzzy rules and managing the investment (individual or portfolio) over time.



	1	2	3	4	5
FSSAM Portfolio	-24	518	707	-3123	13759
Portfolio 1	-31	472	575	-4338	12658
Portfolio 2	-65	717	3109	12700	18927

Figure 6.10: Change in invested capital (data from Table 6.8).

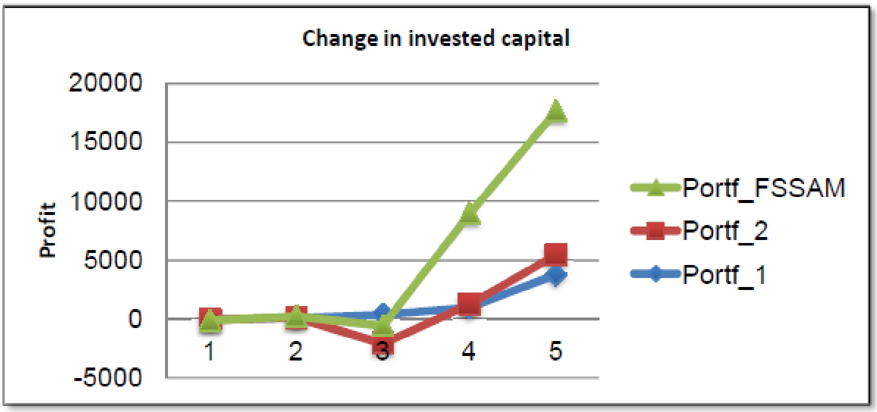


Figure 6.11: Change in invested capital (data from Table 6.8).

References

[1] J. Alcala-Fdez and J.M. Alonso. A survey of fuzzy systems software: taxonomy, current research trends, and prospects. *IEEE Transactions on Fuzzy Systems*, 24(1):40–56, 2016.

[2] R.E. Bellman and L.A. Zadeh. Decision-making in a fuzzy environment. *Management Science*, 17(4):B–141–B–164, 1970.

[3] G. Bojadziev and M. Bojadziev. *Fuzzy Logic for Business, Finance, and Management*. World Scientific Publishing, Singapore, 1997.

[4] BSE. Bulgarian stock exchange. `www.bse-sofia.bg`.

[5] J.Y. Campbell, A.W. Lo, and A.C. MacKinlay. *The Econometrics of Financial Markets*. Princeton University Press, USA, 1997.

[6] C. Carlsson and R. Fuller. Fuzzy multiple criteria decision making: recent developments. *Fuzzy Sets and Systems*, 78(2):139–153, 1996.

[7] Y. Fang, K.K. Lai, and S. Wang. *Fuzzy Portfolio Optimization: Theory and Methods*, volume 609 of *Lecture Notes in Economics and Mathematical Systems*. Springer Berlin Heidelberg, 2008.

[8] P. Georgieva. Minimizing the risk in investment portfolio management (in bulgarian). *Biznes posoki*, pages 35–41, 2005. Did not find in WEB.

[9] P. Georgieva. Distributions of financial assets characteristics. In *XVI International Symposium on Electrical Apparatus and Technologies (SIELA)*, pages 101–106, 2009.

[10] P. Georgieva. A research on soft computing models for real-time management (in bulgarian). Technical report, Sofia: Academic Drinov - Bulgarian Academy of Sciences, 2013. Did not find in WEB.

[11] P. Georgieva and I. Popchev. Cardinality problem in portfolio selection. In *11th International Conference on Adaptive and Natural Computing Algorithms(ICANNGA)*, volume 7824 of *Lecture Notes in Computer Science*, pages 208–217, 2013.

[12] P. Georgieva and I. Popchev. Fuzzy logic Q-measure model for managing financial investments. *Comptes rendus de l¡¯Acade'mie bulgare des Sciences*, 66(5):651–658, 2013.

[13] P.V. Georgieva, I.P. Popchev, and S.N. Stoyanov. A multi-step procedure for asset allocation in case of limited resources. *Cybernetics and Information Technologies*, 15(3):41–51, 2015.

[14] J.E. Hanke and A.G. Reitsch. *Understanding Business Statistics*. McGraw-Hill Education, 1990.

[15] R.A. Haugen. *Modern Investment Theory*. Prentice-Hall, NJ - USA, 5th edition, 2000.

[16] F. Hoffmann, M. Koppen, F. Klawonn, and R. Roy. *Soft Computing: Methodologies and Applications*, volume 32 of *Advances in Soft Computing*. Springer Berlin Heidelberg, 2005.

[17] J.S.R. Jang, C.T. Sun, and E. Mizutani. *Neuro-Fuzzy and Soft Computing: A Computational Approach to Learning and Machine Intelligence*. Prentice-Hall,, 1997.

[18] R. Jang. *Fuzzy inference systems*. Prentice-Hall, NJ - USA, 1997. Did not find

in WEB.

[19] G.G. Judge, R.C. Hill, W. Griffiths, H. Lutkepohl, and T.C. Lee. *Introduction to the Theory and Practice of Econometrics*. John Wiley & Sons, New York - USA, 1988.

[20] A. Kendel. *Fuzzy Expert Systems*. CRC Press, Inc., Boca Raton, FL - USA, 1991. Did not find in WEB.

[21] A. Konar. *Artificial Intelligence and Soft Computing: Behavioral and Cognitive Modeling of the Human brain*. CRC Press, Inc., Boca Raton, FL - USA, 2000.

[22] H.E. Leland. Beyond mean-variance: performance measurement in a nonsymmetrical world. *Financial Analysts Journal*, 55(1):27–36, 1999.

[23] E.H. Mamdani and S. Assilian. An experiment in linguistic synthesis with a fuzzy logic controller. *International Journal of Man-Machine Studies*, 7(1):1–13, 1975.

[24] H.M. Markowitz. *Portfolio Selection: Efficient Diversification of Investments*. John Wiley & Sons, New York - USA, 1959.

[25] D. Nawrocki. Portfolio management using portfolio theory techniques and the PMSP professional software package. *Financial Markets and Portfolio Management*, pages 219–235, 1992. Did not find in WEB.

[26] I. Popchev and P. Georgieva. *Innovative Techniques in Instruction Technology, E-learning, E-assessment, and Education*, chapter A Fuzzy Approach for Solving Multicriteria Investment Problems. 427-431. 2008.

[27] L.S. Riza, C. Bergmeir, F. Herrera, and J.M. Ben¨ªtez. Frbs: fuzzy rule-based systems for classification and regression in R. *Journal of Statistical Software*, 65(6):1–30, 2015.

[28] M. Sugeno. *Industrial Applications of Fuzzy Control*. Elsevier Science, Inc., New York - USA, 1985.

[29] A. Zilouchian and M. Jamshidi. *Intelligent Control Systems Using Soft Computing Methodologies*. CRC Press, Inc., Boca Raton, FL - USA, 2000.