

ITC 3/47

Journal of Information Technology
and Control
Vol. 47 / No. 3 / 2018
pp. 532-563
DOI 10.5755/j01.itc.47.3.18652
© Kaunas University of Technology

**Multi-Criteria Decision Making for Optimal Configuration of
Business Process Model Families**

Received 2017/07/24

Accepted after revision 2018/07/19


<http://dx.doi.org/10.5755/j01.itc.47.3.18652>

Multi-Criteria Decision Making for Optimal Configuration of Business Process Model Families

Ramo Šendelj, Ivana Ognjanović

University of Donja Gorica, Oktoih 1, 81000 Podgorica, Montenegro, e-mails: ramo.sendelj@udg.edu.me, ivana.ognjanovic@udg.edu.me

Corresponding author: ramo.sendelj@udg.edu.me

Business process model families (BPMF) provide collective representation of business processes that can be defined for a target domain of interest. They enable rapid development of appropriate business processes by setting up quick BPMF configuration based on the stakeholders' needs and preferences. In this paper, we propose a novel framework for *user-centric quality-driven* configuration of business process model families capturing different kinds of non-functional preferences and solving the optimal BPMFs configuration problem. Focusing on the following key elements: Analytic Hierarchy Process (AHP), and its extension known as CS-AHP, Fuzzy logic and Genetic Algorithms, the framework maximizes the degree of satisfaction of defined non-functional preferences while simultaneously preserving the behavioural correctness of each derived business process. Furthermore, we provide extensive analyses of the proposed framework from three different perspectives: accuracy assessment, analysis of the impact of different variability and uncertainty elements, and comparison with the published literature indicating that the proposed approach outperforms similar ones in the field.

KEYWORDS: Business Process Model families, Multi-criteria decision making, CS-AHP, Genetic algorithm, Fuzzy logic.

1. Introduction

Business process models (BPM) capture the coordination of a set of activities whose execution realizes specific business goals. Many researchers in BPM community moved one step further by analysing all the possible business processes that could be speci-

fied for a target domain of interest [14], leading to the introduction of a new class of BPMs that facilitate reusability through the so-called *business process model families* (BPMFs) [25]. The purpose of such process families is to 1) enable integral representation of the

processes of the domain; 2) provide comprehensive observation of each individual process within the domain; 3) reinforce reusability of the software components and services that are deployed in the implementation of the business process elements across all business processes of the process family, all with the aim to 4) support rapid development of business processes from the domain by considering stakeholders' needs and preferences and searching for appropriate configuration of BPFM.

Even though BPFM modelling has already been introduced by researchers within the BPM community [25], its automatic configuration which would aim to meet the specific stakeholders' preferences is still a challenge. Given that more and more services with identical or overlapping functionality are available [5], consideration of non-functional preferences is essential for selecting an ideal service among alternatives [34]. However, finding the most appropriate selection of services based on different kinds of stakeholders' preferences presents a great challenge, since: (i) there is no general framework for eliciting and analysing different kinds of preferences [19]; and (ii) automatic service selection, defined as a maximization of some quality measure of configured BPM regarding to the stakeholder's preferences, is an NP-hard problem [21].

To address this issue, we propose and test a novel approach for capturing different kinds of non-functional preferences and solving the optimal BPFMs configuration problem (defined as optimal services selection by *maximizing the degree of satisfaction of defined non-functional preferences with simultaneous preserving the behavioural correctness of each business process derived from a BPFM*). The approach is based on the following three key elements: (i) Analytic Hierarchy Process (AHP) [26] and its extension known as CS-AHP [19] used for addressing different kinds of preferences and dealing with multi-dimensional and often contradictory preferences of individuals; (ii) Fuzzy logic as a precise logic dealing with imprecision, approximate reasoning and inconsistency in information [37], and (iii) Genetic Algorithms (GA) [15] as a meta-heuristic search that solve problems using algorithms inspired by the processes of the neo-Darwinian evolutionary theory.

As such, the study addresses four interrelated goals.

Firstly, we developed a framework for specification of different kinds of non-functional preferences over BPFMs. The second aim was to develop fuzzified measurements of service quality on the basis of CS-AHP results, which was then integrated into a fitness function and GA search technique was used for evaluation of sets of available services. The third aim was to analyse the accuracy of the proposed approach, followed by the assessment of the impact of different variability and uncertainty elements of the whole approach and finally, our aim was to compare the effectiveness of the findings with prior research [15, 21]. The study related to the first two goals resulted in both innovative two-layered structure for specification of different kinds of non-functional preferences over BPFMs (as presented in Section 3.1), and integrated framework for optimal BPFM configuration (as presented in Sections 3.2-3.4). The remaining two goals are succinctly captured into the following three questions (which results are presented in Section 4).

Research Question 1 assesses the overall accuracy of the optimal configuration model:

– RQ1. *What is the accuracy of the proposed optimal configuration model?*

We determine the exact solution by applying brute-force algorithm for finding all possible combinations of available service, and then compare the effectiveness of our approach with this optimal solution.

Research Question 2 examines the accuracy of the optimal configuration model in relation to different variability and uncertainty elements implemented within both BPFM and different kinds of stakeholders' non-functional requirements and attitudes.

– RQ2a. *What is the impact of variability and commonality in BPFMs on the accuracy of the optimal configuration selections?*

– RQ2b. *What is the impact of uncertainty and variety in stakeholders' preferences on the accuracy of the optimal configuration selections?*

Research Question 3 aims to analyse the effectiveness of the proposed approach by comparing it to the published literature and cited methodologies (e.g. [15, 34]).

– RQ3. *How does the proposed framework perform in respect to other adoptions of the same algorithm(s) within the related published work?*

2. Theoretical Background

2.1. Representation of BPMFs

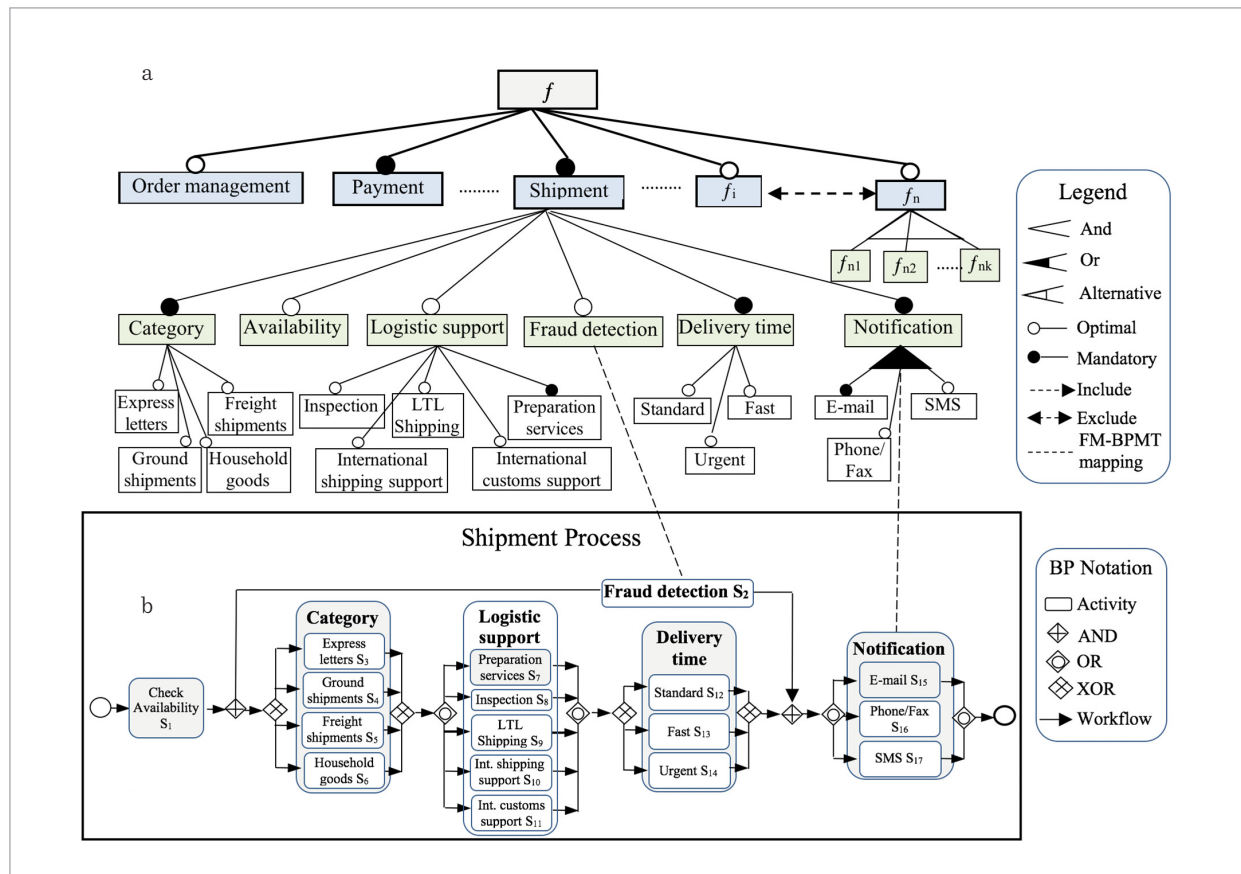
A BPMF is usually presented by a variability model, which defines the variability points of the entire family, using: 1) Feature model (FM) to describe the variability of a family of applications for a given domain; 2) Business Process Model Template (BPMT), which describes a set of reference models (i.e., business processes) of the given domain of the BPMF; and 3) set of available services with identical or overlapped functionality with different QoS properties (e.g. ISO 8402 and ITU standards, etc.). In the approach that is used through the paper, the variation points are managed and configured by means of both models, i.e. feature models and business process models, respectively [21].

2.1.1. Feature Model

Feature models (FM) are widely used to describe variability among applications for a given domain, by specifying features and their interdependencies. A feature is typically referred to as an increment in functionality [10, 11] and the following relationships among features are supported (see Fig. 1a): *mandatory* features- child features that must always appear with their parents; *optional* features- child features that optionally appear with their parents; *alternative* feature groups- a group of features from which only one can be included in any application; and *or* feature groups- a group of features that are optional, but at least one needs to be selected. Some FM notations also allow cardinality within feature groups. Additionally, interdependencies between features are specified using *integrality constraints*. Two of the most widely used

Figure 1

E-shop business process model family a) Feature model with detailed presentation of shipment feature, b) Part of corresponding business process model



constraints are: *includes*– when one feature implies the inclusion of another; and *excludes*– when the appearance of one feature excludes some other features. Fig. 1a shows a part of the simplified FM for an e-shop shipment family.

2.1.2. Business Process Model Template

Business process model families provide a comprehensive view on the collection of business process variants for a target domain [25]. In this paper, we use template-based approach [10] where a reference model (see Fig. 1b) is designed as a template, and variation points are managed and configured by means of two models, FMs and BPMs, respectively. Hence, we assume that there is an injective (i.e. one-to-one) mapping model (see mapping between Fig. 1a and Fig. 1b) which interconnects these two models [10], where each feature in the FM is reciprocally linked to the corresponding activity in the reference BPM.

Pursuant to the contemporary research on BPM we impose the well-formedness conditions on BPM structure [33] which guarantees formal validation and verification of the BPMFs and delivered configurations [14].

In our running example shown in Fig. 1, different methods for payment notification can be considered for different instances of products from the family. Therefore, different payment notification processes can be derived and customized in accordance with: available services, stakeholders' preferences and business objectives. Furthermore, some services are essential (e.g., *EmailNotification* feature is the dominant notification mechanism for online payment), which should be included for all payment notification instances. In addition, stakeholders can decide to include or exclude some extra-functional services (e.g. *SMS* and *Phone/Fax*) based on their needs and preferences. By making decisions which services to select, different instances of business processes from the family can be delivered (see Fig. 2c).

2.1.3. Services and Quality Properties in BPMF

Nowadays, in a more competitive business area there is ever increasing number of service providers that offer more and more services with identical or overlapping functionality [5]. Non-functional characteristics of the services, also known as QoS properties [34] are key indicators for service quality, which can be used

to differentiate services with the same functionality.

Hence, each activity in the BPMF can be delegated and bounded to one or more services, which provide the required functionality with different quality properties. Formally, the quality of a service s is denoted with a vector $Q_s = \langle q_1(s), \dots, q_k(s) \rangle$, where the function $q_i(s)$ determines the values of the i^{th} quality property.

In accordance with researches on both business process variability [30] and requirement engineering [14], it is useful to our study, to deploy the following broader classification of non-functional properties: (i) qualitative properties Q_{qual} - described qualitatively using an ordinal scale (e.g. reliability, security, availability, etc.), and (ii) quantifiable properties Q_{quant} - measured on a metric scale (e.g. price, performance, response time, etc.). The classification allows proper measurement approaches over values of non-functional properties in respect to variability in BPMF, such as:

- 1 Value aggregation over specific configuration of customized BPM (in accordance with [6]);
- 2 Range values estimation over the whole family (based on non-functional properties of available services, in accordance with [18]) (see Fig. 2a).

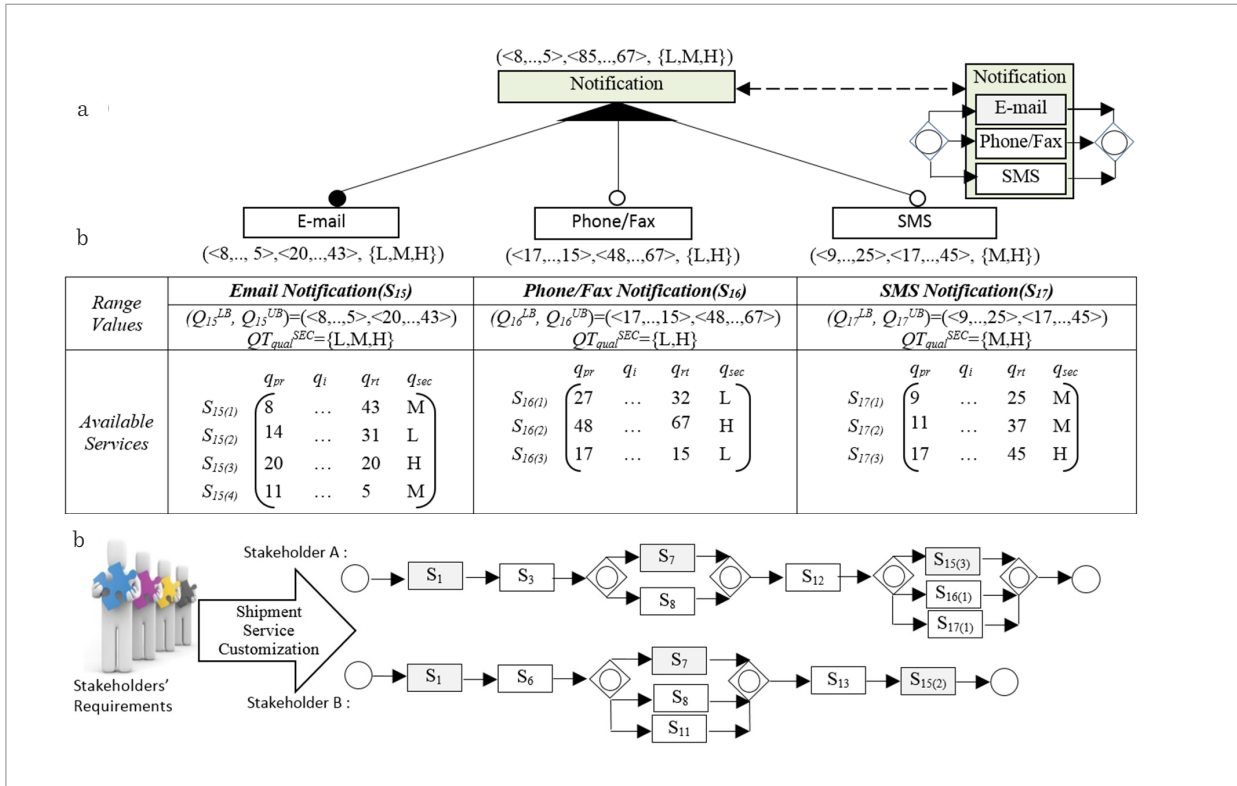
These values will be used for creation of two-layered quality structure of non-functional properties over BPMF.

Let us further explore our running example regarding the selection of methods for payment notification. Fig. 2b shows the sets of candidate services that provide different quality properties (quantitative) q_p, \dots, q_{rt} and (qualitative) q_{sec} for features: *Email Notification*, *Phone/FaxNotification* and *SMS Notification* with given overall ranges, on which bases, the range values for the whole family are estimated (see Fig. 2a). Stakeholder "A" may define that, in relation to his own financial capabilities, range values [8, 25], [25, 40] and [40, 85] represent low, medium and high price values for him, respectively. In addition, he is only prepared to pay a high development price if the response time is kept at lower values; in other circumstances, he is only willing to pay lower price. On the other hand, stakeholder "B" can set his/her preferences that the low price is his/her priority.

However, qualitative property *Security* is already defined with lexical values: *low (L)*, *medium (M)*, *high (H)* and thus both stakeholders shall define own pref-

Figure 2

a) A part of BPMF with estimated range values of non-functional properties based on b) Non-functional specification of available services; c) Selected reference



ferences over them. Let us consider that both stakeholders defined their main interests in the highest possible security levels. Defined stakeholders' attitudes and preferences impose different selection of payment notification services, and thus different reference process models, as presented in Fig. 2c.

2.2. Approaches for Preferences Modelling and Optimization of Business Process Configuration

While there has been much research in business process community focused on BPM analysis [24, 31], there has been comparatively less work examining stakeholders' preferences and BPMF configuration problems. However, the need for recommendation systems and reasoning in the presence of variability and uncertainty in the software product engineering is recently identified as research challenge, only recently starting to receive an attention [21, 24].

There are numerous approaches in the research liter-

ature aimed at eliciting stakeholders' preferences, applied in different domains (e.g. operational research, medicine, education, etc.) [20]. Key challenge in modelling and processing user preferences is posed by the need to express human opinion in a way that can be easily processed by computers [28]. Existing approaches could be categorized based on both, different kinds of preferences, input scales and quality measurements they are generating, as summarized in Table 1.

However, there is no unique technique that can be effectively applied and it should be selected based on the specific needs of application domain [19]. For the purpose of the present study, we use CS-AHP framework [19] based on well-known the Analytical Hierarchical Process (AHP) [26]. The CS-AHP provides the quantitative means for reasoning over different kinds of preferences (e.g. conditionally and un-conditionally defined preferences, preferences about dominant importance). To date, CS-AHP has predominantly been adopted for capturing students' preferences

Table 1

Categorization of different prioritization techniques based on different scale of input/output data and different sort of preferences [19]

Method	Input scales		Output scales		Kinds of preferences		
	Ordinal scale	Initial qualitative transformed into numerical	Ordinal scale	Numerical values	Unconditional	Conditional	Special kinds of preferences
AHP [26]		X		X	X		
TCP nets [4]	X		X		X	X	
CS-AHP [19]		X		X	X	X	X
COP-network [7]		X	X			X	X

and predicting course selection [20], with suggested applications for analysing complex decisions in medicine [29], and measuring security issues in cloud environments [27].

On the other hand, different NP hard optimization problems are well known in the literature and few searching methods and techniques are commonly adopted, such as Integer Linear Programming [32], Hill-Climbing [16] and some meta-heuristic search algorithms that have shown satisfactory capabilities to handle high dimension optimization problems, e.g. Genetic Algorithms [15], Ant Colony Optimization [12] and Particle Swarm Optimization [23].

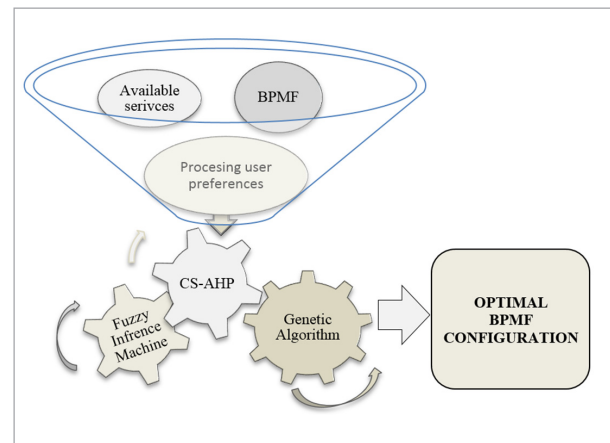
As reported in [22], these techniques have different performance characteristics, depending on target domain, search size, constraints and linearity of optimization problem. The problem of BPF configuration becomes more complex due to variability and variation points presented with two models, feature models and business process templates; and it is a reason why in our study, we use GA as a “well-known method to optimize an objective function with linear or non-linear constraints” [17].

The integrated approach for optimal configuration of BPF in accordance with stakeholders’ preferences, developed upon well-known CS-AHP algorithm, fuzzy theory and genetic algorithms, is summarized in Fig. 3 and includes:

- 1 Specification of different preferences about non-functional properties of BPF, representing in a form suitable for processing by CS-AHP, and transforming into appropriate quality measurements (including fuzzified values for quantitative properties);

Figure 3

Overview of the integrated approach for optimal BPF configuration



- 2 Defining fitness function on the basis of already defined CS-AHP measurements which includes dynamic penalty values in order to make optimal service selection with respect to defined constraints and functional preference.

3. Fuzzy Meta-Heuristic Approach for BPF Configuration

This Section describes our adaptation of the CS-AHP [19] through the creation of two-layered structure for presenting non-functional properties in BPF and different kinds of preferences over them (c.f. Fig. 4). The outcomes of CS-AHP are used for defining quality measurements over available services in BPF

and further searching for an optimal set of services determined as the most appropriate for stakeholders' preferences.

3.1. CS-AHP for Handling User Preferences over Non-Functional Properties in BPMFs

3.1.1. Overview of CS-AHP

Key elements for CS-AHP use are the following: (i) two-layered structure of concerns $C = \{c_1, \dots, c_n\}$ and qualifier tags $QT = \{\{qt_1^1, \dots, qt_{|QT_1|}^1\}, \dots, \{qt_1^n, \dots, qt_{|QT_n|}^n\}\}$ for presentation of decision criteria, (ii) annotation of available options $O = \{o_1, \dots, o_j\}$ in accordance with the structure, and (iii) specification of stakeholders' preferences over the structure in the form of relative importance between concerns and relative importance between qualifier tags of each concern.

Relative importance is typically defined with odd numbers ranging from 1 (of equal importance) to 9 (extreme importance of one over the other); and its definition can depend on fulfilment of the specified condition or dominate the others.

In order to use specified preferences for ranking available options $o_k = qt_{j_1}^1, \dots, qt_{j_n}^n, 0 \leq j_i \leq |QT_i|$ (that are associated with no more than one qualifier tag per each concern in the structure; $j_i=0$ denotes that no qualifier tag of the i^{th} concern is associated to the option), the local ranks of both, concerns and qualifier tags are computed by performing AHP, as follows.

Firstly, by performing simple AHP calculations, it gives ranks r_1, \dots, r_n for the set of concerns $c_i, i \in \{1, \dots, n\}$. The same procedure used for addressing different kinds of preferences is applied for each concern individually, which gives local ranks $r_1^i, \dots, r_{|QT_i|}^i$ for qualifier tags of concern $c_i, i \in \{1, \dots, n\}$. Finally, the ranks of qualifier tags are obtained by multiplying their local ranks by the global rank of an appropriate concern, which gives $r_i \cdot r_1^i, \dots, r_i \cdot r_{|QT_i|}^i, i \in \{1, \dots, n\}$.

By taking into consideration preferences about relative importance, summarizing function f will be used to calculate rank of each option based on the ranks of qualifier tags assigned to each option: $r(o_k) = f(r_{i_1} \cdot r_{j_1}^{i_1}, \dots, r_{i_k} \cdot r_{j_k}^{i_k})$ [19].

Final goal of CS-AHP is to assign higher ranks to the options that are more related to stakeholders' viewpoint. Therefore, the CS-AHP in this paper is applied to address the tasks of optimal BPMF configuration by allowing stakeholders to define different kinds of

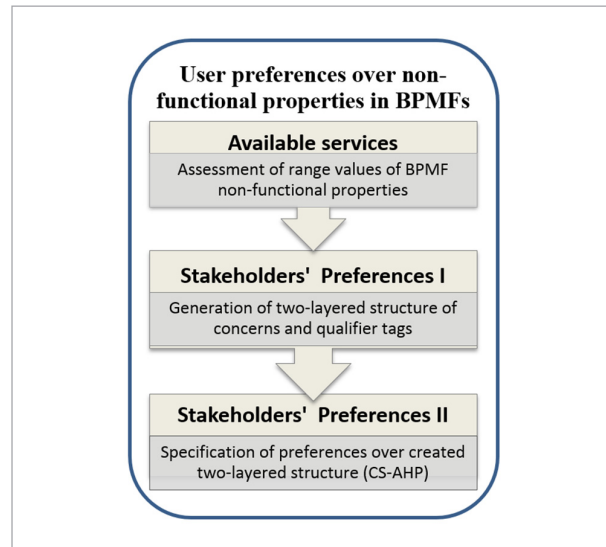
preferences over non-functional properties of available services.

3.1.2. Specification of Stakeholders' Preferences in the Form of Two-Layered Structure over BPMF

To ascertain the degree to which combination of services is appropriate for a stakeholder, we will elicit different kinds of stakeholders' preferences over QoS values $Q = \{q_1, \dots, q_k\}$ by extending the approach developed in [28], which includes the following steps (see Fig. 4).

Figure 4

Two kinds of stakeholders' preferences over BPMF: I) Creation of two-layered structure of concerns and qualifier tags, and II) Preferences over created structure



Step I. (Assessment of range values of non-functional properties over BPMF). The following procedure is applied for quantitative and qualitative QoS properties, respectively:

- 1 The range intervals $(Q_{agg}^{LB}, Q_{agg}^{UB}) = ((Q_1^{LB}, \dots, Q_k^{LB}), (Q_1^{UB}, \dots, Q_k^{UB})) \in R^k \times R^k$ of each *quantitative property* will be computed by applying sound mechanism [18] which firstly aggregates the QoS values of services associated to the activities of the BPMT and then propagates the aggregated values to the features through the mapping relations between the BPMT and the FM (see Fig. 2a).
- 2 Each *qualitative property* is assigned with lexical values (e.g. low, high, etc.); at BPMF level, we in-

clude all possible values of describing property, thus representing values that could be achieved during configuration process.

Step II. (Generation of two-layered structure of concerns and qualifier tags). Concerns $C = \{c_1, \dots, c_n\}$ are a set of quality properties (both quantitative and qualitative) that represent the important matters of interest to the stakeholders such as price, availability, or security. Qualifier tags $QT_i = \{qt_i^1, \dots, qt_i^{|QT_i|}\}$ represent the possible enumerations for each concern $c_i, 1 \leq i \leq n$ (e.g., the qualifier tags for price could be cheap, expensive, and reasonable).

In this research, qualifier tags are created by allowing stakeholders to define their own thoughts and attitudes about quality ranges. For each quantitative property $q_i \in Q_{\text{quant}}$, stakeholders are invited to divide its aggregated intervals $[Q_i^{LB}, Q_i^{UB}]$ into disjoint sub-intervals in accordance with their own attitudes and preferences (therefore, specifying quality tags). Each subinterval QT_i^j of the i^{th} quantitative property can be defined as open, semi-open or close interval $(a_i^j, b_i^j), (a_i^j, b_i^j], [a_i^j, b_i^j), [a_i^j, b_i^j]$, such as the following holds: $\bigcup_{j=1}^{k_i} QT_i^j = [Q_i^{LB}, Q_i^{UB}]$ and $QT_i^j \cap QT_i^t = \emptyset, 1 \leq j < t \leq k_i$. Qualitative properties Q_{qual} are already defined in terms of possible enumerations and thus no additional efforts are needed.

Step III. (Specification of preferences over created two-layered structure- CS-AHP). Stakeholders should define set of preferences over created two-layered structure in the form of CS-AHP mutual relations at two levels: level of concerns, and level of qualifier tags, thus providing essential inputs for CS-AHP algorithm.

As a result, CS-AHP provides ranks of each concern and qualifier tags, which will be used for defining quality measures for each combination of services.

Now, let us further explore our running example to illustrate the whole step-wised approach of specifying stakeholders' preferences and use of CS-AHP algorithm.

By aggregating values of non-functional properties of available services for feature *Notification* (see Fig. 2b), we obtain ranges for each property (see Fig. 2a). Let us assume that stakeholder "A" additionally defined his preferences over the obtained range for total response time [5, 67] (see Fig. 2a and feature *Notifi-*

cation), as follows: interval of low response time is [5, 32], medium response time (32, 50] and high response time is any value above 50. Stakeholder "A" is highly interested in lower values; i.e. low values are more important than medium, medium values are more important than high values; and low values are extremely more important than high values. Let us further assume that the CS-AHP algorithm used for processing these requirements gives the following ranks: $r(\text{ResponseTime.Low})=0.72$; $r(\text{ResponseTime.Medium})=0.19$; $r(\text{ResponseTime.High})=0.09$;

Stakeholder "A" previously (in Section 2.1.3) defined his conditional preferences over total price and security level, and ranks are calculated accordingly:

- in case of low response, the ranks for total price are: $r(\text{Price.Low})=0.16$; $r(\text{Price.Medium})=0.19$; $r(\text{Price.High})=0.65$, otherwise $r(\text{Price.Low})=0.63$; $r(\text{Price.Medium})=0.26$; $r(\text{Price.High})=0.11$
- the ranks for security level are: $r(\text{Security.Low})=0.07$; $r(\text{Security.Medium})=0.26$; $r(\text{Security.High})=0.67$

If we consider price as most important criterion, and security as more important criterion than response time, then ranks are: $r(\text{Price})=0.57$, $r(\text{ResponseTime})=0.14$, $r(\text{Security})=0.29$. Finally, if we consider two combinations of services: $S_{15(1)}$, $S_{16(1)}$ and $S_{15(3)}$, $S_{16(1)}$, $S_{17(1)}$ that gives the summarized values of 35, 43 and L; 56, 32 and M (for price, response time and security) respectively, their final ranks are calculated as: $(0.57*0.26+0.14*0.19+0.29*0.07)/3=0.065$ and $(0.57*0.65+0.14*0.72+0.29*0.26)/3=0.182$. Thus, combination of services $S_{15(3)}$, $S_{16(1)}$, $S_{17(1)}$ is more preferable configuration of feature *Payment* for stakeholder "A".

By considering the calculated ranks, we can also conclude that CS-AHP rank values are assigned to the whole intervals (e.g. 0.19 is assigned to interval [25, 40] of medium costs) which does not respond to realistic situations and attitudes among costs of 25 and 40 (in local currency).

This is why additional refinements are needed over the results obtained by CS-AHP algorithm in order to develop main instruments for measuring the level of satisfaction of user requirements over both the whole BPF and each particular configuration.

3.2. Fuzzy Approach for Measuring Quality of Process Configuration

Stakeholders' attitudes about aggregated quantitative properties very often include the assigned lexical meaning (e.g. low price, high response time, etc.) which is usually understandable for stakeholders, but faces restrictions (e.g. imprecision, semantics, methods of representing knowledge and deduction). Zadeh in [37] clearly emphasized that the use of fuzzy logic is essential in dealing with imprecision and approximate reasoning when formalizing and processing natural language.

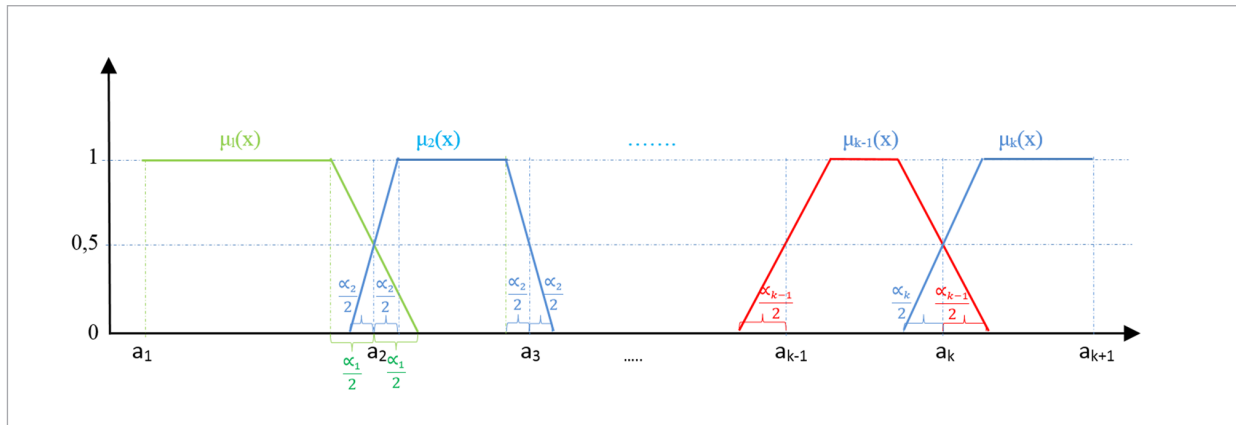
Key element of fuzzy set theory is fuzzy linguistic variable presented by fuzzy sets and appropriate membership function [36].

Fig. 5 illustrates how stakeholders' attitudes over defined qualifier tags $QT_i = [a_i, a_{i+1}]$, $1 \leq i < k$ (with no losing generality presented with close sub-intervals) can be presented by introducing fuzzy sets A_i and corresponding fuzzy functions μ_i .

The process of defining fuzzy set for each qualifier tag may be more or less complicated for the stakeholders. In our study, we use fuzzy sets with a trapezoidal membership function defined over subintervals that correspond to each qualifier tag (as defined by stakeholders for each quantitative property) and tolerance parameters α_i for $1 \leq i < k$ (see Fig. 5), as follows:

$$\mu_i(x, a_i, a_{i+1}, \alpha_i, \alpha_{i+1}) = \begin{cases} 1, & x \in \left[a_i + \frac{\alpha_i}{2}, a_{i+1} - \frac{\alpha_i}{2} \right] \\ & \text{or } x \in \left[a_i, a_i + \frac{\alpha_i}{2} \right] \\ & \text{or } x \in \left[a_{k+1} - \frac{\alpha_k}{2}, a_{k+1} \right] \\ \frac{1}{2} + \frac{x - a_i}{\alpha_i}, & i > 1, x \in \left[a_i - \frac{\alpha_i}{2}, a_i + \frac{\alpha_i}{2} \right] \\ \frac{1}{2} - \frac{x - a_i}{\alpha_i}, & i < k, x \in \left[a_{i+1} - \frac{\alpha_i}{2}, a_{i+1} + \frac{\alpha_i}{2} \right] \\ 0, & \text{otherwise} \end{cases} \quad (1)$$

Figure 5
Definition of fuzzy sets and corresponding membership functions



Furthermore, by assigning CS-AHP ranks of qualifier tags to corresponding fuzzy sets we can define more realistic quality measurements [37], compared to widely used approach of assigning ranks to the whole range of qualifier tag, as illustrated in Fig. 6.

To this end, let us introduce the following notations:
 – b_i^1, b_i^2 - intersection points of function $\mu_i(x, a_i, a_{i+1}, \alpha_i, \alpha_{i+1})$ line $y = r_i$, such that $b_i^1 < b_i^2$;

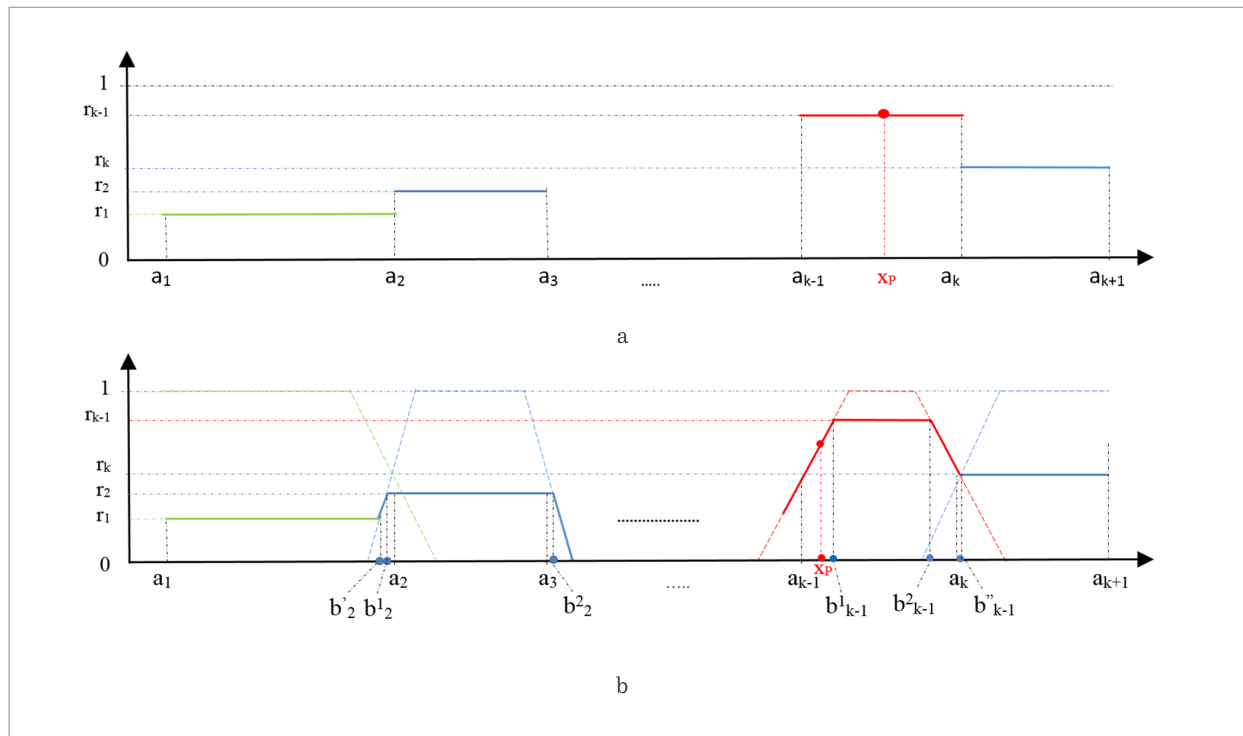
- b_i^1 - intersection point of function $\mu_i(x, a_i, a_{i+1}, \alpha_i, \alpha_{i+1})$ with line $y = r_{i-1}$, which is the closest to α_i ;
- b_i^2 - intersection point of function $\mu_i(x, a_i, a_{i+1}, \alpha_i, \alpha_{i+1})$ with line $y = r_{i+1}$, which is the closest to α_{i+1} .

The quality measurement μO is defined as follows:

$$\mu(x, a_1, \dots, a_{k+1}, r_1, \dots, r_k, \alpha_1, \dots, \alpha_k) = \begin{cases} r_i, & \begin{aligned} & i=1 \text{ and } x \in [a_1, b_1^2] \text{ and } r_1 > r_2 \\ & \text{or } i=1 \text{ and } x \in [a_1, b_2^1] \text{ and } r_1 \leq r_2 \\ & \text{or } i=k \text{ and } x \in [b_{k-1}^1, a_{k+1}] \text{ and } r_{k-1} > r_k \\ & \text{or } i=k \text{ and } x \in [b_k^1, a_{k+1}] \text{ and } r_{k-1} \leq r_k \\ & \text{or } 1 < i < k \text{ and } x \in [b_i^1, b_i^2] \text{ and } r_{i-1} < r_i \text{ and } r_i > r_{i+1} \\ & \text{or } 1 < i < k \text{ and } x \in [b_{i-1}^2, b_i^2] \text{ and } r_{i-1} > r_i \text{ and } r_i > r_{i+1} \\ & \text{or } 1 < i < k \text{ and } x \in [b_i^1, b_{i+1}^1] \text{ and } r_{i-1} < r_i \text{ and } r_i < r_{i+1} \\ & \text{or } 1 < i < k \text{ and } x \in [b_{i-1}^2, b_{i+1}^1] \text{ and } r_{i-1} > r_i \text{ and } r_i < r_{i+1} \end{aligned} \\ \frac{1}{2} + \frac{x - a_{i+1}}{\alpha_{i+1}}, & i < k \text{ and } r_i < r_{i+1} \text{ and } x \in [b_{i+1}^1, b_{i+1}^1] \\ \frac{1}{2} - \frac{x - a_i}{\alpha_i}, & i < k \text{ and } r_i > r_{i+1} \text{ and } x \in [b_i^2, b_i^1] \end{cases} \quad (2)$$

Figure 6

Definition of fuzzy sets and corresponding membership functions with assigned CS-AHP ranks over qualifier tags



Finally, based on defined fuzzy membership functions and assigned CS-AHP ranks, we can estimate most preferable value of the QoS property, defined as crisp de-fuzzified value (most commonly used de-fuzzifica-

tion method in *Centre of singletons*):

$$x_p(\mu, Q_{agg}^{LB}, Q_{agg}^{UB}) = \frac{1}{Q_{agg}^{UB} - Q_{agg}^{LB}} \int_{Q_{agg}^{LB}}^{Q_{agg}^{UB}} \mu(x) dx \quad (3)$$

However, the use of CS-AHP values and fuzzy logic can be formalized by defining fuzzified quality measure over BPF, as follows:

Definition 1. (Fuzzified quality measurement over CS-AHP two-layered structure in BPF).

For a given model (C, QT, P, BPF), in which $C = C_{quant} \cup C_{qual}$ is a set of quantitative and qualitative concerns, $QT = QT_{quant} \cup QT_{qual}$ is a set of qualifier tags (QT_{quant} are defined over aggregated intervals $[Q_{aq}^{LB}, Q_{aq}^{UB}]$ of quantitative concerns C_{quant} ; and QT_{qual} over possible values of C_{qual}), and P represents the set of specified preferences over the two-layered structure in BPF, the fuzzified quality measurement is defined on the bases of CS-AHP output ranks over the set of concerns (written as $r_{|C|}^C, \dots, r_{|C_{quant}|}^C, r_{|C_{quant}|+1}^C, \dots, r_{|C|}^C$), a collection of qualifier tags (written as $r_1^n, \dots, r_{|QT_1|}^n, \dots, r_1^n, \dots, r_{|QT_n|}^n$), corresponding to the sets of qualifier tags $QT = \{QT_1, \dots, QT_{|QT_{qual}|}, \dots, QT_n\}$, where $QT_i = [QT_i^{LB}, QT_i^{UB}]$, $1 \leq i \leq |QT_{quant}|$ and $QT_j = \cup QT_{qual}^{j,k}$, $|QT_{quant}| < j \leq n$, and tolerance parameters $\alpha_1^1, \dots, \alpha_{|QT_1|}^1, \dots, \alpha_1^n, \dots, \alpha_{|QT_n|}^n, \alpha_1^1, \dots, \alpha_{|QT_1|}^1, \dots, \alpha_1^n, \dots, \alpha_{|QT_n|}^n$, as follows:

- 1 Quality measurement of aggregated value v_i for concern $c_i \in C$:

$$f(v_i, c_i) = \begin{cases} r_i^C \cdot r_{|QT_{qual}|}^i \\ r_i^C \cdot \mu(v_i, QT_i^{LB}, QT_i^{UB}, r_{|QT_1|}^1, \dots, r_{|QT_n|}^n, \alpha_1^i, \dots, \alpha_{|QT_i|}^i) \end{cases} \begin{matrix} c_i \in C_{qual}, v_i \in QT_i^{i,k} \\ c_i \in C_{quant}, v_i \in [QT_i^{LB}, QT_i^{UB}] \end{matrix} \quad (4)$$

- 2 Quality measurement of aggregated values $(v_1, \dots, v_{|C|})$ for set of concerns C:

$$f(v_1, \dots, v_{|C|}, C) = \sum_{c_i \in C} f(v_i, c_i) \quad (5)$$

Preferable values of quality properties:

$$w(c_i) = \begin{cases} QT_{qual}^j : r_{|QT_{qual}|}^i = \max_l r_{|QT_{qual}|}^{i,l} & c_i \in C_{qual} \\ x_p(\mu, QT_i^{LB}, QT_i^{UB}) & c_i \in C_{quant} \end{cases} \quad (6)$$

3.3. BPF Configuration Goals

In addition to requirements about preferable values of QoS properties, hard requirements can be defined as limiting values of specific non-functional properties over the whole BPF and its parts [6]. Formally, **hard constraints** can be defined as a set of constraints cl_i : $cl_i \langle s_1, \dots, s_k \rangle \leq u_i, i \in \{1, \dots, l\}$, where s_i represents the selected service which corresponds to the i^{th} activity and u_i is a constant limitation value.

Definition 2. (Business Process Families Optimal Configuration Goal).

Given a BPF with a CS-AHP two-layered quality structure ($C = C_{quant} \cup C_{qual}$, $QT = QT_{quant} \cup QT_{qual}$, P, BPF), a set of hard constraints H_c and a set of available services $\{s_1, \dots, s_{|a_i|}\}$ per each activity a_i of the BPF, the goal of the configuration of the BPF is to find a valid combination of services under satisfaction of hard constraints which maximizes the overall **fuzzified quality measurement** in respect to **stakeholders' preferable values of quality properties**.

3.4. GA Approach for BPF Configuration

In this section, we present the way in which GA is used for selection of valid combination of services in relation to the BPF optimal configuration goals. GA is an adaptive heuristic search algorithm with identified key elements suitable for presentation of BPF [21]:

- 1 *Service Chromosome* – combination of services representing individuals in the constant-size ‘population’. We use array encoding $(e_1, e_2, \dots, e_{|A|})$ where e_i corresponds to configuration option of the i^{th} activity (it is either an index of available services, or indicates that the i^{th} activity will not be included in final configuration).
- 2 *Population* is generated from initial population by applying traditionally utilized operators [15]: k -point crossover operator (which splits the genome at k crossover points and combines parts among parental genomes) and mutation operator (which makes random modifications on individual genomes, in order to prevent convergence to local optima).

These activities may violate optionality and integrity constraints within the BPF [21], and commonly used approach [15] defines *servicesTransform* algorithm as sound mechanism for transforming gener-

ated chromosome during any of GA steps into valid service selections.

– *Fitness evaluation*- A fitness function is used for assessing each member of population as well as making performance analyses over the population. With reference to previous work [15, 21, 28] as well as to developed fuzzified measurements over CS-AHP two-layered structure, the fitness function will incorporate two types of penalty factors, as follows:

– *Penalty factor 1*: weighted distance from constraint satisfaction measured as:

$$D(e_1, \dots, e_k) = \sum_{i=1}^l cl_i(e_1, \dots, e_k) \cdot y_i, \tag{7}$$

where

$$y_i = \begin{cases} 0, & cl_i(e_1, \dots, e_k) \leq u_i \\ 1, & cl_i(e_1, \dots, e_k) > u_i \end{cases} \tag{8}$$

If the weight for the penalty factor is low, there is the risk that individuals will not be discarded although they violate the constraints [15]. Thus, the dynamic penalty [15] is adopted with weight $w_1(gen)$ that increases with the number of generations gen .

– *Penalty factor 2*: weighted relative distance between quality measurements of considered combination of services and the combination of most preferable properties defined by stakeholders. The dynamic penalty value with weight $w_2(gen)$ will increase when the combination becomes further away from the most preferable combination, thus navigating convergence process.

Finally, the fitness function is defined as:

$$\begin{aligned} Fitness(e_1, \dots, e_{|A|}, \alpha_1^1, \dots, \alpha_{|QT_1|}^1, \dots, \alpha_1^n, \dots, \alpha_{|QT_n|}^n, C, QT) = \\ = \frac{1}{f(v_1(e_1, \dots, e_{|A|}), \dots, v_{|C|}(e_1, \dots, e_{|A|}), C)} + w_1(gen)D(e_1, \dots, e_{|A|}) + \\ + w_2(gen) \frac{f(v_1(e_1, \dots, e_{|A|}), \dots, v_{|C|}(e_1, \dots, e_{|A|}), C) - f(w(c_1), \dots, w(c_{|C|}), C)}{f(w(c_1), \dots, w(c_{|C|}), C)} \end{aligned} \tag{9}$$

By iterating this process, the population elements are evaluated in respect to fitness function, and more preferable individuals are selected (i.e. those with

lower value of fitness function). The process is completed when one of the following conditions is satisfied: (i) all hard constraints are met and the lowest value of the fitness function is reached (in predefined number of iterations), or (ii) the best-achieved individual chromosome remains unchanged for a given number of iterations.

4. Analysis

4.1. Experimental Setup

To test accuracy of the proposed approach, we use the *closeness to an optimal solution* measured as relative distance between quality values of two combinations of services. This experimental study is aimed to investigate research questions (RQ1, RQ2a, RQ2b, RQ3) and therefore, different simulation analyses were performed by utilizing the following random generators with different input parameters, as presented in Table 2:

- 1 FM generator is used for generation of feature models;
- 2 BPF generator extends FM generator by creating business process templates in accordance with injective (one-to-one) mapping with FM;
- 3 CS-AHP generator is used for generation of both two-layered structure of concerns and qualifier tags and sets of preferences over developed structure.

The domain characteristics, developed in previous empirical research, will be also included in our study (as presented in Table 2). Furthermore, commonly used GA parameters are [15] population size =50; number of generations =200; crossover probability =1 (always applied), mutation rate = 0.1.

To address RQ1, we used the brute force algorithm to determine all valid combinations of services in order to enforce the hard constraints and find the most preferable one. Finally, we run the proposed GA approach to find an optimal configuration and to compare it with those obtained with brute force algorithm.

To tackle RQ2a and RQ2b, we controlled creation of BPF and CS-AHP structure [21] by identifying different experimental groups related to different variability parameters, as presented in Table 3.

Table 2

Random generators with input parameters and domain characteristics (M- Mean values; SD- Standard deviation)

Generator/ Algorithms	Input parameters	Domain characteristics	
		Value	Reference
FM generator [15]	No. of features	Min=14; Max=287; M = 76,86; SD = 96,25	[2]
	Distribution of commonality and variability patterns	-	-
BPMF generator [21]	Distribution of composition patterns	-	-
	Distribution of sequential and parallel-AND patterns	-	-
	No. of services per activity	Max= 100	[21]
CS-AHP generator [19]	No. of concerns	Max= 10	[19]
	No. of qualifier tags	Max=7	[19]
	No. of conditionally defined preferences	-	-
	No. of preferences about dominant importance	-	-
	Qualitative QoS properties	4 groups of QoS characteristics; random number of characteristics to each group	[6]
	Quantitative QoS properties: range values		
	Quantitative QoS properties: distribution of qualifier tags	-	-
Fuzzy tolerance parameters	Min= 5%; Max=20%	-	

Table 3

Description of experimental groups for RQ2

Research question	Experimental groups	Description		Research question	Experimental groups	Description	
		Characteristics	Value			Characteristics	Value
RQ2a	1	Distribution of optional features	25%	RQ2b	16	Quantitative and qualitative QoS properties	25%
	2		50%		17		50%
	3		75%		18		75%
	4		100%		19		100%
	5	Feature groups/patterns	Equally AND, OR, XOR		20		Qualifier tags
	6		AND-OR	21	1 dominant		
	7		AND-XOR	22	2 dominants		
	8		OR-XOR	23	3 dominants		
	9		AND	24	Unconditionally defined preferences	25%	
	10		OR	25		50%	
	11	XOR	26	75%			
	12	BPMF patterns: sequential and parallel-AND	25%	27		100%	
	13		50%				
	14		75%				
	15		100%				

On the other side, to address RQ3, we identified the following key areas where GA, fuzzy logic and CS-AHP algorithm were used for solving different configuration problems in wide range of literature: FM configuration [15], service composition [5, 34], and BPF configuration [21]. In order to make comparisons and analyse the results, we applied our approach over similar domain constraints and limitations (e.g. for FM evaluation: only variability/commonality variations are included; for service composition evaluation: only composition patterns are included, etc.) and compared the results.

Each simulation is preformed 1000 times and the collected data are further analysed.

4.2. Simulation Analysis

4.2.1. Analysis Techniques

Given the type of the collected data in the simulations, the analyses employed were standard descriptive statistics [3] including mean (M) and standard deviation (SD) values. All parameters as being inputs for generators of FMs, BPF as fuzzified CS-AHP two-layered structure were considered as *interval data*.

The ANOVA test was used to check for significant differences in the accuracy in the prediction as a result of changing the parameters' values among groups identified in RQ2. For results which were not normally distributed, we used parametric tests over log-transformed data.

A *t*-test was used to assess whether the fuzzification of the whole approach significantly improved the accuracy of the configuration (as compared to our previous work in [21]). Shapiro-Wilk test of normality was used and confirmed that the *t*-test assumptions were satisfied for all uses of the *t*-test in the study.

4.2.2. Results

The results are organized according to the research questions.

Research question 1. This research question calculates the accuracy of the proposed approach to identify optimal configuration of BPF, compared to brute force algorithm, as well as against the reported previous work such as [20].

The mean value of the relative distances between the obtained combination of services and the most preferable configuration (obtained by brute force algo-

rithm) was 10.04% (SD = 1.27%). Thus, the fuzzified approach accurately found 89.96% of most preferable BPF configurations.

This result shows that presented approach may be considered as most appropriate for optimal configuration of BPF (previously reported accuracy was 89.77% [21] and 89.59% [28]), while detailed analyses of the accuracy of the model in respect to model variability and uncertainty, and comparative analyses are presented in RQ2 and RQ3, respectively.

Research question 2. The descriptive statistics for the experiments over groups 1-15, and 16-27 are reported in Tables 4 and 5, respectively.

The ANOVA results show a non-significant difference between groups 1-4 (related to different distributions of optional features in FM): $F(3; 823) = 15.47$; $p = 0.721$ and groups 12-15 (related to different distributions of sequential and parallel AND patterns): $F(3; 825) = 13.37$; $p = 0.681$.

On the other hand, significant difference in the assessed accuracy of different distributions of patterns in FM model is showed ($F(6; 2845) = 1.72$; $p = 0.000$). The Tukey's post hoc test revealed that there was significant difference only between group 5 and groups 9-11.

Thus, results showed that optimality of our approach depends only on variability patterns in FMs, and does not depend on distribution of optimal feature or patterns in BPF, since they are mostly determined by FMs.

Furthermore, the results demonstrate a significant difference in the assessed accuracy in the case of different distribution of QoS properties: ($F(3; 847) = 2.07$; $p = 0.005$). The Tukey's post hoc test revealed that there was significant difference only between groups 16 and 19. Thus, the approach is significantly close to optimal solution in case of considering only qualitative properties, while by taking into account quantitative properties and different kinds of preferences, additional complexity and uncertainty are imposed.

We also proved that the significant difference in the assessed accuracy of different distributions of qualifier tags over the range intervals of quantitative properties existed: ($F(3; 847) = 1.68$; $p = 0.000$). The Tukey's post hoc test revealed that there was significant difference only between groups 21 and 22-23. Thus, the accuracy of the proposed approach may be increased by adjusting the tolerance parameters in membership functions based on qualifier tags distributions.

Table 4

Mean and standard deviation values of the assessed accuracy in experiments from the conducted study in research question RQ2a

Experimental Group	Percentage ratio of optional features in FM model	Assessed accuracy	Experimental Group	Distribution of sequential and parallel AND patterns	Assessed accuracy
		Mean; St. Dev.			Mean; St. Dev
1	25%	90.11%; 1.03	12	25%	90.16%; 0.79
2	50%	90.05%; 0.91	13	50%	89.92%; 1.24
3	75%	90.08%; 0.95	14	75%	89.83%; 1.07
4	100%	89.76%; 0.82	15	100%	89.93%; 0.93
Experimental Group	Distribution of patterns in FM model	Assessed accuracy	Experimental Group	Distribution of patterns in FM model	Assessed accuracy
		Mean; St. Dev			Mean; St. Dev
5	33% (AND), 33% (OR), 33% (XOR)	89.63%; 0.86	9	100% (AND)	90.13%; 0.72
6	(random) AND-OR	89.75%; 0.94	10	100% (OR)	90.06%; 0.93
7	(random) AND-XOR	89.71%; 1.04	11	100% (XOR)	89.93%; 1.04
8	(random) OR-XOR	89.69%; 0.88			

Table 5

Mean and standard deviation values of the assessed accuracy in experiments from the conducted study in research question RQ2b

Experimental Group	Distribution of quantitative/ qualitative QoS properties	Assessed accuracy	Experimental Group	Distribution of quantitative/ qualitative QoS properties	Assessed accuracy
		Mean; St. Dev.			Mean; St. Dev.
16	25%	93.74%; 0.84	18	75%	90.03%; 1.45
17	50%	92.35%; 1.02	19	100%	88.21%; 1.78
Experimental Group	Distribution of qualifier tags over range intervals	Assessed accuracy	Experimental Group	Distribution of conditional preferences	Assessed accuracy
		Mean; St. Dev.			Mean; St. Dev.
20	Almost equal qualifier tags	89.97%; 0.82	24	25%	89.83%; 0.96
21	One dominant qualifier tag	90.06%; 0.78	25	50%	89.72%; 1.07
22	Two dominant qualifier tags	90.03%; 1.24	26	75%	89.69%; 1.72
23	Three dominant qualifier tags	89.98%; 1.27	27	100%	89.74%; 1.65

Table 6

A comparative comparison of the proposed approach with respect to the published literature and cited methodologies

Optimisation approach / Prioritization technique	Reference	Variability and commonality (FM)	Composition patterns (BPMT)	Uncertainty in stakeholders' preferences	Assessed accuracy
					Mean; St. Dev
Genetic algorithm / SAW	[5]		X		95%; -
Experiment in this study			X	X	91.75%; 1.32
Genetic algorithm / SAW	[15]	X			86-90%; -
Experiment in this study		X		X	90.04%; 0.96
Genetic algorithm / Fuzzy logic	[34]		X	X	87%; 0.7
Experiment in this study			X	X	90.45%; 2.05
Genetic algorithm / CS-AHP	[21]	X	X	X	89.77%; 0.98
Experiment in this study		X	X	X	89.96%; 1.27

The results also demonstrate the significant difference in the assessed accuracy between groups 24-27 (related to different distributions of conditional preferences): $F(3; 852)=1.79$; $p=0.000$). The Tukey's post hoc test revealed that there was significant difference only between groups 24 and 27.

Research question 3. Table 6 outlines the results obtained by other applications of GA, fuzzy logics and CS-AHP algorithm for different configuration problems

The use of our approach revealed a considerable decrease of 3.25% in the performance compared to the results of the GA with Simple Additive Weighting (SAW) technique applied to service composition. On the other hand, our approach showed almost similar accuracy results compared to the use of GA with fuzzy logics for personalized web service selection [34] and FM configuration [15], while significant improvement was identified compared to the previous work on use of GA and CS-AHP algorithm for BPMT configuration [21]. A t-test confirmed that the increase in the accuracy was significant ($t(1969) = 5.17$, $p < 0.05$) as compared to the accuracy of the model where fuzzy refinements were not used over CS-AHP rank values (as reported under RQ1).

5. Discussion

The results of the evaluation showed that the highest accuracy is achieved compared to other approaches with complex weighting mechanism for prioritization of stakeholders' preferences that correspond to real-world scenarios. To assist in the interpretation and impact of considered variability and uncertainty, in both, domain space and stakeholders' preferences, we conducted two evaluation studies.

Firstly, by controlling input parameters that correspond to variability and uncertainty elements, we analysed their impact on accuracy level. We observed that variability patterns in the FM have key impact for accuracy level of the configuration process. Furthermore, the experiments show that the ability to define different kinds of non-functional preferences has significant influence on the effectiveness of the whole approach. On the other hand, integrated elements of fuzzy logic proved their ability to capture stakeholders' preferences in a manner that is closer to real scenarios and human attitudes.

Finally, we compared the accuracy level of the proposed approach to those of other related published studies. The presented approach showed significant

improvements and increased accuracy level compared to the previous work [21]. It benefits from (i) improved fitness function (as being critical for presentation of domain characteristics [1]), and (ii) refined quality measurement over CS-AHP values generated by taking into consideration more subjective and realistic information about non-functional properties (as initially investigated in [34]).

When our approach is considered in relation to tackling FM configuration [34], even under specification of different kinds of preferences with CS-AHP, the results showed performance characteristics close to values reported in the literature. This probably resulted in the guided search which captured the critical importance of differentiating good solutions from poor ones [1], as beneficial to integrating fuzzified measurements and crisp values.

6. Conclusion and Future Work

In this paper, we proposed a novel framework for optimal BPFM configuration by considering various stakeholders' preferences, compared to traditional models that considered only preferences defined over either- quantitative and qualitative QoS properties. Our framework provides the following major benefits to the process of optimal BPFM configuration:

- 1 It presents a framework for specification of different kinds of *preferences and attitudes about qualitative and quantitative QoS properties* over BPFM including conditional preferences and attitudes about their values;
- 2 It proposes a fuzzified extension of the CS-AHP method for measuring the service quality. In addition, aiming to find optimal BPFM configuration it is combined with GA search technique which is guided dynamically by using *crisp values* as referenced values for most preferable quality properties;
- 3 Empirical analyses in this paper indicated that the proposed approach outperforms similar ones in the field.

While both concepts of BPFM configuration and

user-centric service-oriented modelling approach [8] are already known in the literature, to our best knowledge, this is the first approach that considers *user-centric quality-driven* configuration of business process families. The proposed approach fulfils the preferences of an effective multi-criteria prioritization technique [19] and effective use of genetic programming for software engineering modeling [1, 9].

Furthermore, compared with other QoS models (e.g.[15]), proposed framework gives QoS properties with un-static final ranks and as such, they are not simply aggregated by static weighting factors. The approach integrates both *qualitative* and *quantitative QoS properties*, which in terms of domain-dependent quality characteristics [6] enable the scalability of our solution up to potential use in different domains and scenarios.

In a broader sense, the study contributes to the body of research knowledge in the fields of business process management [24, 31] and automated software engineering [13]. While the proposed research can be connected with the field of requirement engineering [35], the questions and method adopted generates results and findings that are more applicable for AHP practitioners to aid practical use of different kinds of preferences in different scenarios.

We propose that future research needs to be undertaken in order to: (i) include *automatic resolution of inconsistencies* in defining preferences over CS-AHP structure [19], (ii) extend introduced fuzzified CS-AHP structure to include *variant-wise quantifiable properties* [30] as properties with no meaning for single features or with no ability to be quantified from the aspect of the influence of individual features on the non-functional properties of a concrete variant, (iii) develop recommendation strategies for making selection of appropriate GA operators and *fuzzy tolerance parameters* that might correspond to different QoS properties groups and stakeholders' preferences, and (iv) develop hybrid solutions [38] by integrating other meta-heuristic approaches with good performance characteristics in related optimization issues.

References

1. Afzal, W., Torkar, R. On the Application of Genetic Programming for software Engineering Predictive Modeling: A Systematic Review. *Expert Systems with Applications*, 2011, 38, 11984-1199. <https://doi.org/10.1016/j.eswa.2011.03.041>
2. Bagheri, E., Gašević, D. Assessing the Maintainability of Software Product Line Feature Models Using Structural Metrics. *Software Quality Journal*, 2011, 19(3), 579-612. <https://doi.org/10.1007/s11219-010-9127-2>
3. Blaikie, N. *Analyzing Quantitative Data*. Sage, London, 2003. <https://doi.org/10.4135/9781849208604>
4. Brafman, R. I., Domshlak, C. Introducing Variable Importance Tradeoffs into CP-Nets. *Proceedings of the 18th Conference on Uncertainty in AI*, San Francisco, USA, 2002, 69-76.
5. Canfora, G., Di Penta, M., Esposito, R., Villani, M. An Approach for QoS-Aware Service Composition Based on Genetic Algorithms. *Proceedings of the 7th Annual Conference on Genetic and Evolutionary Computation*, Washington DC, USA, 2005, 1069-1075. <https://doi.org/10.1145/1068009.1068189>
6. Cardoso, J., Sheth, A. P., Miller, J. A., Arnold, J., Kochut, K. J. Quality of Service for Workflows and Web Service Processes. *Journal of Web Semantics: Science, Services and Agents on the World Wide Web*, 2004, 1(3), 281-308. <https://doi.org/10.1016/j.websem.2004.03.001>
7. Chen, S., Buffett, S., Fleming, M. W. Reasoning with Conditional Preferences Across Attributes. *Proceedings of the 20th Conference of the Canadian Society for Computational Studies of Intelligence on Advances in Artificial Intelligence*, Montreal, Canada, 2007, 369-380. https://doi.org/10.1007/978-3-540-72665-4_32
8. Cheng, D. Y., Chao, K. M., Lo, C. C., Tsai, C. F. A User Centric Service-Oriented Modeling Approach. *World Wide Web*, 2011, 14, 431-459. <https://doi.org/10.1007/s11280-011-0115-7>
9. Cpalka, K., Lapa, K., Przybyl, A. A New Approach to Design of Control Systems Using Genetic programming. *Information Technology and Control*, 2015, 44(4), 433-442. <https://doi.org/10.5755/j01.itc.44.4.10214>
10. Czarnecki, K., Antkiewicz, M. Mapping Features to models: A Template Approach Based on Superimposed Variants. In: Glück R., L. M. (Eds.), *Generative Programming and Component Engineering*, Lecture Notes in Computer Science, Springer, Berlin, Heidelberg, 2005, 422-437. https://doi.org/10.1007/11561347_28
11. Damaševičius, R., Paškevičius, P., Karčiauskas, E., Marcinkevičius, R. Automatic Extraction of Features and Generation of Feature Models from Java Programs. *Information Technology and Control*, 2012, 41(4), 376-384. <https://doi.org/10.5755/j01.itc.41.4.1108>
12. Dorigo, M., Di Caro, G., Gambardella, L. M. Ant Algorithms for Discrete Optimization. *Artificial Life*, 1999, 5(2), 137-172. <https://doi.org/10.1162/106454699568728>
13. Godena, G., Lukman, T., Heričko, M., Strmčnik, S. The Experience of Implementing Model-Driven Engineering Tools in the Process Control Domain. *Information Technology and Control*, 2015, 44(2), 135-147. <https://doi.org/10.5755/j01.itc.44.2.6258>
14. Gröner, G., Bošković, M., Pareiras, F. S., Gašević, D. Modeling and Validation of Business Process Families. *Information Systems*, 2013, 38(5), 709-726. <https://doi.org/10.1016/j.is.2012.11.010>
15. Guo, J., White, J., Wang, G., Li, J., Wang, Y. A Genetic Algorithm for Optimized Feature Selection with Resource Constraints in Software Product Lines. *Journal of Systems and Software*, 2011, 84, 2208-2221. <https://doi.org/10.1016/j.jss.2011.06.026>
16. Mahdavi, K., Harman, M., Hierons, R. M. A Multiple Hill Climbing Approach to Software Module Clustering. *Proceedings of the IEEE International Conference on Software Maintenance*, Los Alamitos, California, USA, 2003, 315-324. <https://doi.org/10.1109/ICSM.2003.1235437>
17. Mariajayaprakash, A., Senthilvelan, T., Gnanadass, R. Optimization of Process Parameters Through Fuzzy Logic and Genetic Algorithm – A Case Study in a Process Industry. *Applied Soft Computing*, 2015, 30, 94-103. <https://doi.org/10.1016/j.asoc.2015.01.042>
18. Mohabbati, B., Gašević, D., Hatala, M., Asadi, M., Bagheri, E., Bošković, M. A Quality Aggregation Model for Service-Oriented Software Product Lines Based on Variability and Composition Patterns. In: Kappel G., M. Z., Motahari-Nezhad H.R. (Eds.), *Service-Oriented Computing*, Lecture Notes in Computer Science, Springer, Berlin, Heidelberg, 2011, 436-451. https://doi.org/10.1007/978-3-642-25535-9_29
19. Ognjanović, I., Gašević, D., Bagheri, E. A Stratified Framework for Handling Conditional Preferences: An

- Extension of the Analytic Hierarchy Process. *Expert Systems with Applications*, 2013, 40(4), 1094-1115. <https://doi.org/10.1016/j.eswa.2012.08.026>
20. Ognjanović, I., Gašević, D., Dawson, S. Using Institutional Data to Predict Student Course Selections in Higher Education. *Internet and Higher Education*, 2016, 29, 49-62. <https://doi.org/10.1016/j.iheduc.2015.12.002>
 21. Ognjanović, I., Mohabbati, B., Gašević, D., Bagheri, E., Bošković, M. A Metaheuristic Approach for the Configuration of Business Process Families. *Proceedings of the IEEE International Conference on Service Computing*, Hawaii, USA, 2012, 25-32. <https://doi.org/10.1109/SCC.2012.6>
 22. Palubeckis, G., Karčiauskas, E., Riškus, A. Comparative Performance of Three Metaheuristic Approaches for the Maximally Diverse Grouping Problem. *Information Technology and Control*, 2011, 40(4), 277-285. <https://doi.org/10.5755/j01.itc.40.4.977>
 23. Poli, R., Kennedy, J., Blackwell, T. Particle Swarm Optimization. *Swarm Intelligence*, 2007, 1(1), 33-57, <https://doi.org/10.1007/s11721-007-0002-0>
 24. Rocha, R. D. S., Fantinato, M. The Use of Software Product Lines for Business Process Management: A Systematic Literature Review. *Information and Software Technology*, 2013, 55, 1355-1373. <https://doi.org/10.1016/j.infsof.2013.02.007>
 25. Rosa, M. L., Dumas, M., Ter Hofstede, A. H. M., Mendling, J. Configurable Multi-Perspective Business Process Models. *Information Systems*, 2011, 36(2), 313-340. <https://doi.org/10.1016/j.is.2010.07.001>
 26. Saaty, T. L. *The Analytic Hierarchy Process*, McGraw-Hill, New York, 1980.
 27. Šendelj, R., Ognjanović, I., Semantically Enhanced Cyber Security over Clouds: Methodological Approach. *Proceedings of the International Conference on Advances in Information Processing and Communication Technology – IPCT 2014*, Rome, Italy, 2014, 126 - 131. <https://doi.org/10.15224/978-1-63248-021-7-90>
 28. Šendelj, R., Ognjanović, I. Framework for Optimal Selection Using a Metaheuristic Approach and AHP Algorithm. In: De Felice, F., Petrillo, A., Saaty, T. (Eds.), *Applications and Theory of Analytic Hierarchy Process*, InTech Open Access, 2016, 193-217. <https://doi.org/10.5772/63991>
 29. Šendelj, R., Ognjanović, I., Ammenwerth, E., Hackl, W. Towards Semantically Enabled Development of Service-Oriented Architectures for Integration of Socio-Medical Data. *Proceedings of the 5th IEEE Mediterranean Conference on Embedded Computing (MECO)*, Bar, Montenegro, 2016, 436-440. <https://doi.org/10.1109/MECO.2016.7525687>
 30. Siegmund, N., Rosenmüller, M., Kuhlemann, M., Kästner, C., Apel, S., Saake, G. SPL Conqueror: Toward Optimization of Non-Functional Properties in Software Product Lines. *Software Quality Journal*, 2012, 20, 487-517. <https://doi.org/10.1007/s11219-011-9152-9>
 31. Skersys, T., Butleris, R., Kapocius, K., Vileiniskis, T. An Approach for Extracting Business Vocabularies from Business Process Models. *Information Technology and Control*, 2013, 42(2), 178-190. <https://doi.org/10.5755/j01.itc.42.2.2310>
 32. Srinivasan, V., Shocker, A. D. Linear Programming Techniques for Multidimensional Analysis of Preferences. *Psychometrika*, 1973, 38, 337-369. <https://doi.org/10.1007/BF02291658>
 33. Vanhatalo, J., Völzer, H., Leymann, F. Faster and More Focused Control-Flow Analysis for Business Process Models Through SESE Decomposition. In: Krämer, B. J., Lin, K. J., Narasimhan, P. (Eds.), *Service-Oriented Computing – ICSOC 2007*, Lecture Notes in Computer Science, Springer, Berlin, Heidelberg, 2007, 43-55. https://doi.org/10.1007/978-3-540-74974-5_4
 34. Wang, H. C., Lee, C. S., Ho, T. H. Combining Subjective and Objective QoS Factors for Personalized Web Service Selection. *Expert Systems with Applications*, 2007, 32, 571-584. <https://doi.org/10.1016/j.eswa.2006.01.034>
 35. Wasielewska, K., Ganzha, M., Paprzycki, M., Szmeja, P., Drozdowicz, M., Lirkov, I., Badica, C. Applying Saaty's Multicriterial Decision Making Approach in Grid Resource Management. *Information Technology and Control*, 2014, 43(1), 73-87. <https://doi.org/10.5755/j01.itc.43.1.4587>
 36. Yuen, K. K. F. Membership Maximization Prioritization Methods for Fuzzy Analytic Hierarchy Process. *Fuzzy Optimization and Decision Making*, 2012, 11, 113-133. <https://doi.org/10.1007/s10700-012-9119-8>
 37. Zadeh, L. Is there a Need for Fuzzy Logic? *Information Sciences*, 2008, 178(13), 2751-2779. <https://doi.org/10.1016/j.ins.2008.02.012>
 38. Zhao, C. Y., Wang, J. L., Qin, J., Zhang, W. Q. A Hybrid Algorithm Combining Ant Colony Algorithm and Genetic Algorithm for Dynamic Web Service Composition. *The Open Cybernetics & Systemics Journal*, 2014, 8, 146-154. <https://doi.org/10.2174/1874110X01408010146>