A Metaheuristic Search Framework to Derive Cancer Care Services from Business Process Models

Hamzeh Aljawawdeh
Software Engineering Research Group (SERG)
Faculty of Environment and Technologies
University of the West of England
Bristol, BS16 1QY, United Kingdom
hamzeh.aljawawdeh@uwe.ac.uk

Mohammed Odeh
Software Engineering Research Group (SERG)
Faculty of Environment and Technologies
University of the West of England
Bristol, BS16 1QY, United Kingdom
mohammed.odeh@uwe.ac.uk

Christopher Simons
Computer Science Research Group
Faculty of Environment and Technologies
University of the West of England
Bristol, BS16 1QY, United Kingdom
chris.simons@uwe.ac.uk

Nawras Lebzo
IT Department
The King Hussein Cancer Centre (KHCC)
Amman, Jordan
NL.10688@KHCC.JO

***Abstract*—Cancer Care involves not only handling patients' medical or physical needs but also other services to facilitate patient needs which are underpinned by appropriate software systems that assist in patient care processes. The Service-Oriented Architecture (SOA) model of computing has become widely adopted and can provide efficient and agile business solutions in the face of rapid changes to business requirements. Instead of adopting a more traditional way of building an IT system for Cancer Care by rigidly piecing together a collection of hardware, software and networking, SOA offers the opportunity to build the IT systems in an increasingly flexible and reconfigurable way. However, current service identification methods can suffer from shortcomings such as a lack of computational support, and not being able to address all the necessary activities of the service identification. To address these shortcomings, this paper presents a comprehensive metaheuristic search framework for deriving SOA-based services applied to Cancer Care business process models. This framework is evaluated using both quantitative and qualitative methods with the help of domain experts at King Hussein Cancer Centre (KHCC), Jordan. Evaluation by domain experts confirmed that the resulting services are feasible (i.e., valid services that can be practically applied for real-life projects) that the domain experts might not have arrived at manually. Statistical analysis shows candidate services produced by the search-based framework are superior to the services produced manually by domain experts at KHCC with respect to metrics for coupling and cohesion.**

***Index Terms*—Service Identification Methods, Cancer Care Software Services, SBSE, Search-based Software Engineering, Business Process Modelling, BPMN**

I. Introduction

Service Oriented Architecture (SOA) has been widely utilised for providing effective and agile solutions to achieve business-IT alignment [1]. Utilising SOA systems helps to keep abreast with the rapid changes in the business environment [2]. Using an SOA paradigm to build software systems for the Cancer Care supports the on-going improvement of the underlying business processes that support it. Identifying software services that satisfy the Cancer Care is one of the key activities in developing SOA solutions that contribute to the fields of SOA and Healthcare [3]. However, there is a body of evidence to suggest that identifying the right services is a difficult, non-trivial, and cognitively demanding task to perform [4]. Identifying the wrong services at this stage causes deleterious consequences for downstream development [5]. Errors made at this stage can be propagated through to the next stages of design, implementation, and verification [6].

However, service identification methods (SIMs) can suffer from serious shortcomings. For example, the majority of SIMs are not fully comprehensive as they do not cover all the phases of service identification (i.e., the majority of SIMs address the service identification phase alone), and therefore, some critical aspects of service identification may be ignored [1]. In addition, there is a lack of computational support such that SIMs rely heavily on the software engineer to fulfil the service identification activities [7].

Computationally intelligent support has, however, been applied to address software development in the field of software engineering. For example, Search-Based Software Engineering (SBSE) is an approach that applies bio-inspired metaheuristic search techniques such as genetic algorithms or ant colony optimisation to software systems engineering problems [8], [9]. Finding the optimum solution using dynamic programming or linear programming may not be feasible or practical for large-scale software engineering problems because of the

computational complexity. Thus, researchers and practitioners have used metaheuristic search techniques to find near optimal or good-enough software solutions [10]. Because of this, we hypothesise that SBSE techniques can be utilised to address service identification as an optimisation problem. Indeed, we propose that using SBSE techniques can computationally support the process of deriving candidate service for Cancer Care, while at the same time enriching the quality of these services via a selected set of design metrics related to coupling and cohesion. To investigate this proposal and utilise the SBSE in the best way to derive candidate services for Cancer Care, the following research questions are devised:

- **RQ1**: To what extent can role-based business process models such as the BPMN 2.0 models be mapped to Service Oriented Architectures (SOAs)?
- **RQ2**: In what ways can SOA services be best represented for metaheuristic search?
- **RQ3**: How can the services solution space be effectively and efficiently explored and exploited?

In this paper, a novel metaheuristic search framework for service identification is introduced to derive SOA candidate solutions for the Cancer Care from the Business Process Modelling and Notation (BPMN). This top-down framework aims to ensure business-IT alignment. One of the key objectives of this framework is to provide computationally intelligent support for all activities of service identification. In addition, it aims to enrich the quality of the resulting solutions by exploring and exploiting the search space until finding high-quality solutions that are measured using quantitative design metrics.

Moreover, using an interactive human preference to steer the trajectory of the search is anticipated to enrich the quality of the resulting solutions [11]. However, interactive context is out of the scope of this paper, but this research aims to develop a flexible framework that enables the use of the interactive search in the future.

The remainder of this paper is outlined as follows. Section 2 presents the current work focusing issues of top-down service identification methods. Section 3 introduces the proposed metaheuristic search framework for service identification from Cancer Care business process models. The framework is demonstrated in Section 4 and evaluated in Section 5 using domain experts from KHCC, Jordan. Finally, we conclude the paper and suggest directions for future study in section 6.

II. Background

A. The State-of-the-art in Service Identification

The issue of bridging the gap between the business context and software systems has gained the attention of many researchers and practitioners in the last decade [12]. Investigating the business-IT alignment requires focusing on top-down approaches that aim to derive candidate services from the business process. Therefore, the focus of this research attempt is on methods that start from a high-abstract input type such as the business process, business goals, and business requirements. A semi-automated approach for service identification was presented in [1]. This approach aims to achieve the business-IT alignment by deriving the SOA services from the business process or business goals. Although this SIM automates some service identification activities, the human involvement is required at some stages. In addition, a static clustering algorithm is adopted, which produces one candidate solution only.

In their attempt [3] Yousef (2010) have proposed BPAOntoSOA which is an ontology-based top-down SIM that derives candidate services from the enterprise Business Process Architecture (BPA). This method manages all the phases of the service identification and produces a list of candidate SOA services. However, not all the phases of the service identification are automated, thus the human should handle some activities. In addition, the static clustering algorithm utilised by BPAOntoSOA uses the relationships between business elements as the only quantitative measure of service identification. Which neglects some important factors such as human qualitative feedback during the service identification process and if there are tight relations between services, solutions may not conform to SOA principles.

In their paper, [13] present a research attempt that adopts business process models to derive SOA services. The presented method comprises three phases. Firstly, the preparation phase (i.e., to determine the scope and define the stakeholders). Secondly, the service analysis phase (i.e., utilises the SOA principles to check for IT feasibility). And thirdly, the service categorisation phase, in which the service-types and operations are defined (i.e., the output refinement phase). The main drawback of this method is the lack of automation, which means that the software engineer handles all the mapping activities based on a set of guidelines.

In a further research, [14] adopt a graph clustering approach to identify the services from the BPMs. The clustering algorithm measures the relationships between the activities in order to reduce the coupling and increase the cohesion between the local tasks. In addition, this approach prepares the input data in a specific level of granularity that helps to fulfill the service identification process. However, this SIM adopts a static algorithm only that uses the quantitative measures only to map the business activities to the corresponding services. Thus, the human evaluation is neglected. Moreover, the details of the identification approach are not revealed, which makes it impossible to reproduce the algorithm or repeat the experiment.

In a further research, [6] have proposed a top-down search-based SIM that uses a genetic algorithm (i.e., GA) to identify business services from the enterprise business processes. This SIM automates the phase of service identification and uses a set of quantitative design metrics to measure the resulting solutions. This method focuses only on the automation of the service identification phase, however, the other phases of SOA service derivation are not fully addressed. The other limitation is the lack of using the qualitative assessment of experts to examine the quality of the services, while at the

same time relying on the human to prepare the input elements, such as extracting elements from business process models and preparing the output. The outcome of this method is a list of candidate services not a full SOA solution.

Another research attempt to propose a SIM is presented in [5], in which a top-down SIM to identify services from business process models is presented. This SIM adopts a metaheuristic search algorithm and uses quantitative measures to evaluate the service abstractions, such as coupling, cohesion and reusability. This method produces a list of candidate services that are feasible and conform to the pre-defined business requirement. However, this method addresses a set of activities that covers one phase of the full-service identification process. It considers the clustering of business activities to produce a list of clusters (i.e., candidate services) but not full cycle model. Other activities of the service identification are managed either by the business architect (e.g., preparing the business elements in a CRUD matrix), or by solution architect (e.g., producing a full SOA model out of the resulting services). Moreover, the interactive preference of human experts is not considered during the process of service identification, which depends mainly on the quantitative measures of the design metrics.

B. Conclusion Remarks

This paper presents the set of the most important capabilities that should construct a comprehensive framework for service identification. Deriving high-quality candidate services requires the development of a comprehensive framework that manages all phases of service identification. The automation is another significant aspect to be addressed. In addition, deriving services for Cancer Care makes gives this research a higher level of importance. Although the derivation of candidate services has a great impact on the entire development life-cycle of the SOA applications, the capabilities specified above have not been fully satisfied by any of the works cited in this paper. There is still a need for a comprehensive framework that automates the full process of service identification while achieving the business-IT alignment.

III. The Proposed Framework

This section presents the proposed framework for Cancer Care service identification from BPMN. The first part of this section presents the layers of the proposed framework. This part aims to address the first and second research questions (RQ1 and RQ2) of this paper. The second part shows the components of the search-based algorithm. This part of the section aims to address the third research question (RQ3).

A. The layered framework

The proposed framework comprises three layers and each layer manages a specific task. This design enables a high-level of flexibility to update any part of this framework. For example, change the meta-model of the input business process in the first layer, or utilise different techniques for service identification in the second layer (e.g., search or interactive search). The three layers are (i) the input BPMN preparation layer, (ii) the service identification layer, and (iii) the service refinement layer. Fig. 1 presents the layers of the framework. The outcomes of each layer feed into the lower layer in order to produce candidate solutions for Cancer Care.

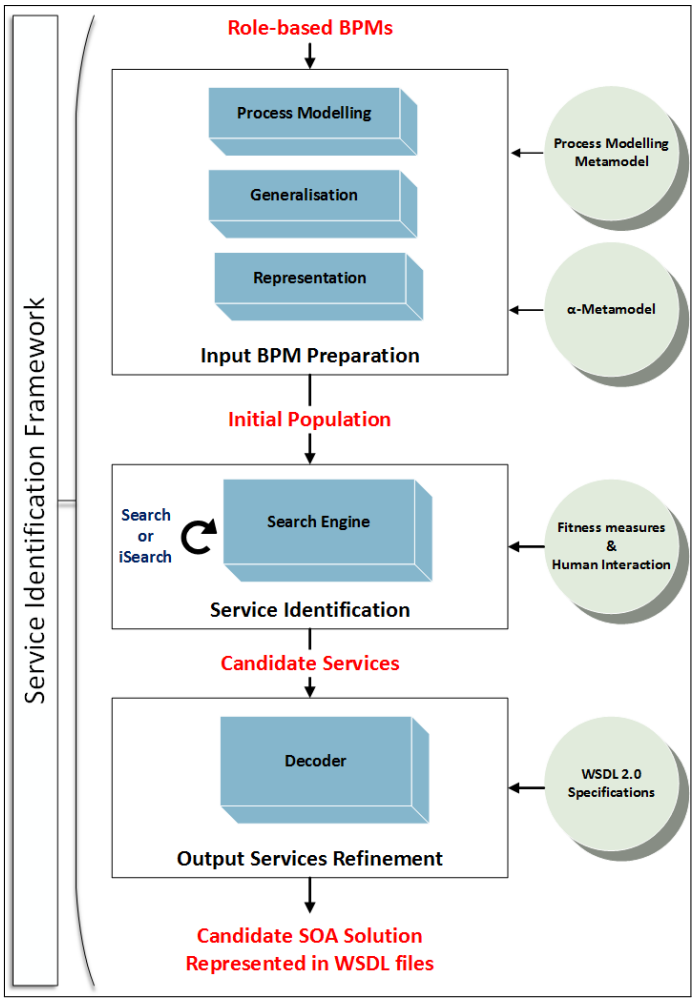

Fig. 1. The layered framework for service identification

(i) The BPM preparation layer: the purpose of this layer is to prepare the input Cancer Care business models (i.e., in BPMN format) in the right granularity level in order to construct the search space. This step is important to enhance the performance of the search and protect the chronological sequence of the business process. Three sub-layers comprise this layer in order to satisfy the objective of the input preparation layer.

1) Business process modelling: in this sub-layer, the BPMN models are traversed to collect information about the business process elements the flow of activities. The collected data include the element type, the role ID, the role name, and the position of the element in the sequence flow.
2) Generalisation: this sub-layer helps to categorise the business process elements into groups based on the semantic

similarities. A tag is assigned to each business process element based on its type. Although BPMN is adopted by this research, this sub-layer is important as it supports using different role-based modelling languages such as role activity diagrams (RAD) as input. The output of this sub-layer is a set of search space elements ($SSEs$) that are the basic building blocks of the search space.

3) Representation: the mechanism to map the resulting $SSEs$ from the previous sub-layer to the candidate services. The representation should be designed to enable the exploration and exploitation of the search space through the corresponding genetic operators (e.g., crossover and mutation). The representation along with the corresponding genetic operators will be discussed in more detail in the second part of this section.

The output of the input data preparation layer is an initial search space that comprises a set of $SSEs$. Creating the abstract $SSEs$ and representing them in a way that maps the BPMs to SOA candidate services has many advantages. It allows a more efficient search because managing a smaller number of elements in the search space (i.e., $SSEs$) is easier than managing all of the raw elements. In addition, the $SSEs$ protect the essential sequential flow of business activities.

(ii) The service identification layer: this layer aims to assign the $SSEs$ to the candidate services such that the distribution of these elements will result in effective and efficient solutions. Grouping $SSEs$ into candidate services is based in the first place on the relationships between these elements. Therefore, it is recommended to allocate $SSEs$ with strong relationships in the same candidate service. However, it is important to show a clear purpose of each candidate service, which encourages to have a fewer number of $SSEs$ in the resulting services. In order to measure the extent to which these two factors are applied in a specific solution, a fitness function will be used. The design metrics that comprise the fitness function examine the relationships between different services (i.e., coupling) and the clear purpose of each candidate service (i.e., cohesion). The fitness function is discussed in the second part of this section. Since the service identification is considered as an optimisation problem, the genetic algorithm will manage to find effective (i.e., with high fitness values) and efficient (i.e., in a short time) solutions. The best situation of a resultant candidate solution is when $SSEs$ inside each service have the maximum intra-relations (i.e., internal relations inside a service) and minimum inter-relations (i.e., relations between services) [1]. This supports the reusability of the resulting services by producing services with high cohesion and low coupling.

(iii) The candidate service refinement layer: Web services have been defined as web application components that can be published, found and used on the Web [15]. A standard XML-based interface definition language that is used to describe the functionalities of a web service is called the Web Services Description Language (WSDL) [16]. Constructing the WSDL files is a needed step to produce an SOA solution, therefore, this layer aims to map each element in the resulting candidate services to the corresponding component of a WSDL file. Although there is no direct one-to-one matching between the business process elements and the WSDL components, the elements inside the resulting candidate services are either assigned to a corresponding component (e.g., task is assigned to operation) or used to do a specific functionality (e.g., message flows are assigned to service inputs and outputs). Table I presents the main business process elements and the corresponding WSDL components.

TABLE I
MAPPING THE BPMN ELEMENTS TO WSDL FILE COMPONENTS

BPMN 2.0 Element	WSDL component
Activity elements (e.g., User Task)	Operation
Data Elements Messages	Service inputs and outputs
Events and connection elements	Binding, Interconnections, and communications between services

This layer constructs the full standard structure of the WSDL file and places each component in the right place. The binding section in each WSDL file manages the connections between the resulting candidate services.

B. The search algorithm

Since the service identification can be addressed as an optimisation problem, a search-based technique will be used. A genetic algorithm (GA) is adopted for this research to perform the search for optimised solutions. The Multi-objective Genetic Algorithm (MoGA) is an enhancement of the Single-objective Genetic Algorithm which is a promising technique that attained good results with a variety of problems in software engineering. One of the most widely applied MoGA techniques is the Non-dominated Sorting Genetic Algorithm II (NSGA-II) [17]. An enhancement version of NSGA-II is the NSGA-III [18]. This algorithm is designed to manage many objectives (i.e., more than two and could reach up to 20 objectives) by changing the selection mechanisms.

Although the MoGA techniques have shown promising results with different problems in software engineering, as well as in the field of service identification [5], [19], the objectives of this research suggest the adoption of a single-objective GA with a weighted-sum fitness function. This selection of a single-objective GA can be justified by two reasons. Firstly, the simplicity of the algorithm helps to maintain a focus on the context of bridging the gap between the business and the services rather than focusing on a large number of variables that are needed for a MoGA. Secondly, the single-objective GA can show more sensitivity to the human interaction, which paves the way to use the human interactive preference in the future to steer the trajectory of the search. A fitness function that includes a small number of design metrics is anticipated to be more sensitive to the interactive preference values. Which satisfies one of the main objectives of this research. As a consequence, just two design metrics are selected to create the objective fitness function (i.e., coupling and cohesion).

The GA contains two main components; the representation along with the genetic operators (i.e., selection, crossover, and mutation) and the fitness function. The GA can be implemented in different ways, for example, one way was the mechanism introduced by Goldberg [20]. Fig. 2 shows an activity diagram of the search-based GA adopted by this research. An integer-based representation is adopted, in which the SSEs have static positions, and an integer number that represents the service ID is assigned to each SSE. The genetic operators are applied to the integer service IDs such that changing the integer value that is assigned to a certain SSE indicates that the SSE belongs to a new candidate service that has the corresponding ID.

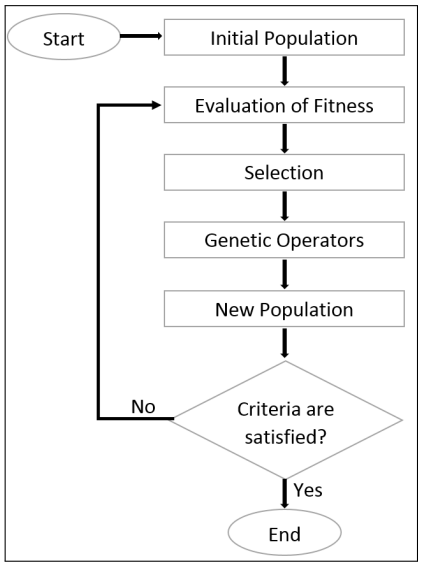

Fig. 2. The genetic algorithm activity diagram [20]

1) The Representation: This research adopts an integer representation approach that is inspired by [21] to represent the service identification problem. Chromosomes basically comprise only integer values, where the position of the gene within the chromosome represents the SSE, and the genes integer value denotes their service assignment. Each position in the vector is assigned to one SSE. Fig. 3 shows an example use of the representation technique in which SSE_1 (i.e., the first gene) belongs to the 2nd service, whereas, SSE_2 belongs to the 6th service. The resulting solution presented in this example is represented by the vector (2, 6, 6, 2, 1, 5, 3, 4, 1, 3). Since the chromosome representation assigns an SSE to each gene within the chromosome, the length of the chromosome consequently equals the number of $SSEs$ in the search space. The positions of the elements in the first row are static and will not change. Instead, the genetic operators (i.e., crossover and mutation) would be applied to the gene position (i.e., the second row).

Adopting this representation method efficiently utilises integers to represent service IDs, which is a problem representation that naturally takes the form of grouping elements together [22], [23]. A key advantage of this representation technique is that it implements the check for constraints that guarantee the feasibility of the resulting SOA solutions. Therefore, resulting candidate services do not lack $SSEs$, and in addition, each SSE can appear in one service only. This significant advantage reduces the execution time and leads to an improvement in the performance of the search process. According to the representation method of adoption, the associated genetic operators are formulated as follows:

SSE_1	SSE_2	SSE_3	SSE_4	SSE_5	SSE_6	SSE_7	SSE_8	SSE_9	SSE_{10}
2	6	6	2	1	5	3	4	1	3

Fig. 3. Representation example

(a) Selection: the tournament selection mechanism is used by this research. A number of individuals are selected at random from the population for later breeding and the fittest individual among these selected individuals is chosen as a parent. This technique is selected because it is (i) coding efficient [24] (ii) has the capability to manage either maximisation or minimisation problems without performing any structural changes [25], and (iii) has the ability to create more diverse populations by providing a uniform probability for all population individuals to be in the new generation [23], [26].

(b) Crossover: A single point crossover operator is applied to represent the recombination. This function takes two parents and generates two offspring from them. This operator exchanges the allocation of $SSEs$. It is based on switching the mapping of two individuals around a pivot point that is selected at random [27]. Fig. 4 presents an example on the single point crossover operator. The position for the crossover point is selected randomly. The first child inherits the ID numbers from the first position to the crossover point in the first parent, i.e., p = 0, 1, 2, 3. It inherits the reset of IDs from the second parent where p=4, 5, 6, 7. The second child inherits the remaining service IDs. From two parents, the crossover produces two children.

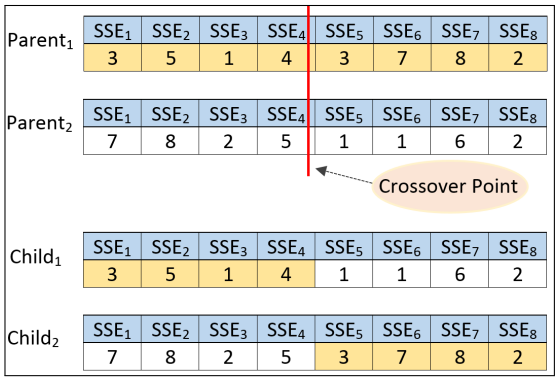

Fig. 4. Crossover

(c) Mutation: The mutation operator is based on switching the mapping of two $SSEs$. Two positions are selected at random, and the service IDs at these positions are swapped [28]. Fig. 5 illustrates an example of applying the mutation

operator. Note that the two service IDs at positions '1' and '5' were swapped.

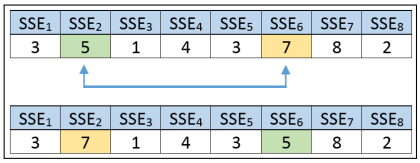

Fig. 5. Mutation

2) The Fitness Function: the effective factors in deriving candidate services that are assessed using fitness function are coupling and cohesion. The choice of these two metrics is due to their strong influence on the granularity level as well as the reusability of the services [1], [5]. The fitness values highlight the quality of the desired solution by applying mathematical calculations to quantitatively assess the inner relations inside each service (i.e., the cohesion), and the degree of the relationship between one service and other services (i.e., the coupling value), see [5]. The first design metric is inspired by the coupling factor (CpF) [1]. Ideally, the resulting services are stand-alone i.e., they do not need to be connected to other services to fulfil any task. These services score a CpF of zero. The second design metric adopted is cohesion to reflect the extent that an SOA service has a clear purpose (i.e., service abstracts the underlying business), which can be achieved using the cohesion metric. The cohesion of Service (ChS) metric has been adopted by [29] to measure the intra-relations of a service. Thus, if the service performs only one function, the cohesion scores a ChS of 1.0. Good solutions are obtained through the minimization of CpF, and the maximization of ChS. A single-objective fitness function that aggregates values of all design metrics in a single value is implemented [30], [31]. The goal is to maximise the weighted sum of all the quality metrics [32].

a) Coupling: the coupling metric measures the relative degree of interdependence among services [33], by quantifying the extent to which the services within a solution are interconnected [34]. The conversation between the services is implemented using send/receive messages. The number of messages and the size of them are important factors in representing the degree of coupling between these services [1]. Although this is a simple count, it is very useful in identifying real-world problems by demonstrating the effectiveness of the general strategy [5]. Based on this, the coupling factor of a service X, denoted by $CpF(x)$ is formulated in (1).

$$CpF(x) = \frac{I_x}{SSE_x + Ix} \tag{1}$$

I_x represents the number of interactions with other services, and SSE_x is the number of the $SSEs$ inside the service. For example, if a candidate service comprises six $SSEs$, and calls three other services, then the coupling factor $CpF(x) = 3/(6+3) = 0.33$. If a candidate service comprises a number of $SSEs$ but does not initiate any interactions (i.e., stand-alone service), the coupling factor will be zero.

b) Cohesion: this metric indicates the degree of strength of relationships between the operations of a service [5], [35]. Within a service, each SSE is strongly connected to other $SSEs$ if their chronological relationships have a strong dependency [36]. The impact of grouping different tasks in one candidate service is minimising the cohesion of that service as it no longer focuses on a single functionality. Nevertheless, the strongest cohesion can be achieved when a service focuses on one conceptual task [37]. According to this description, the internal dependency (i.e., cohesion) metric helps to optimise the correlation of $SSEs$ such that each candidate service encapsulates the more relevant $SSEs$. The formula for Cohesion of a Service x, denoted by $ChS(x)$, is presented in (2).

$$ChS(x) = \frac{1}{|SSE_x| * |SSE_x - 1|} \sum_{i=1,j=1}^{n} \delta_{ij}$$
$$\delta_{ij} = \begin{cases} 1, & SSE_i \text{ calls } SSE_j \\ 0, & \text{Otherwise} \end{cases} \tag{2}$$

I_x represents the number of elements inside the service, n is the number of potential connections inside the service, and δ_{ij} represents the existence of a relationship between SSE_i and SSE_j. A service contains three elements $S = \{SSE_1, SSE_2, SSE_3\}$, SSE_1 requests data from SSE_2, and SSE_2 requests data from SSE_3. The total number of elements is three, whereas the number of connections between elements is two. Thus, the cohesion of service is calculated as follows: $ChS(x) = (1/(|3| * |3-1|)) * 2 = (1/6) * 2 = 1/3$.

c) Weighted sum fitness function: The fitness function adopted by the search-based service identification method combines the coupling and cohesion design metrics in a single fitness value. The goal is to maximise the weighted sum of the fitness value. This can be achieved by finding the best allocation of $SSEs$ in the candidate services that achieves high-cohesion and low-coupling. Examples of using similar approaches are found in [38], [39]. The fitness value will be calculated using the formula presented by (3).

$$Fitness(x) = (Weight_{Coupling} * (1 - CpF(x))) + (Weight_{Cohesion} * ChS(x)) \tag{3}$$

If the $CpF(x) = 0.2$ and $ChS(x) = 0.70$. Considering that coupling and cohesion have equal weights (i.e., 0.5 each), the fitness value is calculated as follows:

$$Fitness(x) = 0.5 * (10.2) + 0.5 * 0.70 = 0.4 + 0.35 = 0.75.$$

IV. Demonstration

Experiments in this section aim to demonstrate the service identification framework using a case study from the Cancer Care. Demonstrating the proposed framework is required to validate the service identification framework and examine its effectiveness and efficiency. In addition, it is useful to

examine the feasibility of the resulting solutions. To show that the search algorithm is useful to derive Cancer Care services, the first step is to create these services manually by experts and then use the search to automate the service identification process. By doing this, the comparison between the two techniques (manual and search) can be examined in terms of effectiveness (i.e., fitness values).

A. Experimental Design

With regard to the manual service identification process, seven domain experts from the King Hussein Cancer Centre (KHCC) with different levels of experience were recruited to participate in deriving the candidate SOA solutions from the BPMN models. The allocation of $SSEs$ in candidate services is based on the relationships and interactions between these elements and relies upon the knowledge and experience of the participant domain experts. This experiment helps to show the capability of the proposed framework to derive achievable service solutions. Understanding the interactions between different $SSEs$ relies upon the implicit knowledge and experience of the domain experts.

With regard to the search experiment, Initial parameter values have been derived from the literature of evolutionary computing or used by previous research studies [20], [22], [40]. Using a set of empirical trail-and-error experiments, the following parameters are found useful to perform the test: selection size is 7, crossover probability is 0.8, mutation probability is 0.03, population size is 100, number of generations is 500, and a generational replacement strategy is adopted. To examine reliability, each search is run 50 times to provide average population fitness curves. The search engine tool is implemented in Java, and the experiments are conducted on a standard desktop PC running the Microsoft Windows operating system.

B. The Case Study

The case study is an important empirical method for evaluating the resulting artefacts of the service identification framework. Selecting the right case study is a significant task as this supports the generalisation of the developed service identification framework to be used within other domains.

The Cancer Care and Registration (CCR) case study [3], [41] has been adopted to demonstrate the effectiveness of the metaheuristic search framework and its associated service identification methods. The CCR case study represents a real-world example for the KHCC in Jordan and has been assessed to be sufficient enough to carry out investigating the effectiveness of this framework [41]. This case study has been validated and improved by previous research attempts [3], [41], [42]. A traceability mechanism has been introduced to support tracing the CCR process elements for the service identification activities in this framework. And more specifically, this case study provides a comprehensive representation of the CCR process model activities such as the roles, the activities within the roles, and the interactions between the roles. In addition, the CCR case study has been examined to reveal a large scale and complexity that is sufficient enough to test the framework. Finally, the flexibility provided by the clear roles of the CCR process models supports constructing the $SSEs$ at different levels of granularity.

C. Results

A sample solution is presented in Table II

TABLE II
SAMPLE SOA SOLUTION PRODUCED USING THE SEARCH

#	Abstract Functions	Main Activities
1	Patient and Imaging Dept.	Perform test and save the results
2	Patient and Radiotherapy Dept.	Begin treatment and check if imaging test is needed
3	Patient and Inpatient care specialists and nurses	Perform surgery and update patient's file
4	Receptionist and Medical Records	Register patient's details and save it in the library
5	Patient and Lab	Perform test and save the results
6	Inpatient care specialist and nurses And clerk clerk	Check patient's financial state, review doctor's orders and diagnoses patients and review old tests
7	Patient and Receptionist	Register patient's details and book appointments
8	Patient general reception	Request and book appointments, inform the patient about the appointments and register patient's details
9	Patient, doctor, registrar, receptionist and medical records clerk	Collect patient data, produce reports, book appointments with doctors, update patient's record, review patient's history, generate statistical reports and analyse collected data

Table III shows the population average fitness, coupling, and cohesion values after arriving at a fitness plateau using the search method. The average fitness value is 0.761 (Coupling=0.152, and Cohesion=0.673). The average fitness value is 0.498 (Coupling=0.211, and Cohesion=0.208). Since the aim is to maximise fitness, it is observed that fitness values achieved by the search-based solutions (i.e., average is 0.761) are higher than fitness values achieved manually (i.e., 0.498). This reflects the superior coupling values between the services produced using the search, as well as the high cohesion in each of these candidate services when compared to the manual services. The low-coupling potentially enables more stand-alone functionality. Moreover, the high cohesion values achieved for the services reflect a clarity in their purpose. Fitness value for coupling and cohesion are aggregated within a maximization function, and therefore, the higher the fitness, the better the solution.

V. EVALUATION

Domain experts have confirmed that the resulting services are valid and adhere to SOA principles. This shows that the search-based framework is capable of producing valid SOA candidate solutions that are accepted by experts. The outcomes of the search method achieve higher fitness values in comparison with the fitness values obtained manually.

TABLE III
POPULATION AVERAGE FITNESS VALUES AT A FITNESS PLATEAU USING THE SEARCH AND THE MANUAL METHODS

Solution	Search	Manual
1	0.761	0.52
2	0.75	0.461
3	0.762	0.59
4	0.758	0.53
5	0.758	0.47
6	0.77	0.436
7	0.765	0.492
Average	0.761	0.498

The larger number of configurations that are available in the search landscape allows for the production of more candidate solutions with better fitness values. In contrast, the landscape for solutions produced by domain experts is limited and relies on the knowledge and experience of the domain experts. The capability of the search method to find a better optimisation amongst a large population is much higher than the capability of a domain expert to find such an optimisation, which explains the large difference between the fitness values obtained using the two methods. To compare the means of the two populations produced by two independent methods (i.e., the search and manual), a test for the distribution of the data should be conducted to confirm that the sample data have been drawn from a normally distributed population. Due to the small sample size, a Shapiro-Wilk test for normality is performed at an alpha level of 0.05. For both the search and manual data samples, the p-values are greater than the alpha level (i.e., 0.961 for search, and 0.804 for manual). In addition, the skewness and kurtosis are within acceptable limits of ± 2 [43]–[45]. This implies that the data are normally distributed and that indicates the possibility of performing a parametric test such as the t-test. Table IV shows some statistics and Table V presents the results of conducting the statistical analysis test on the two samples sets.

TABLE IV
GROUP STATISTICS

Method	Mean	Std. Deviation	Std. Error Mean
Search	0.761	0.006	0.002
Manual	0.498	0.05	0.019

TABLE V
INDEPENDENT SAMPLES T-TEST

t-value	df	Sig. (2-tailed)	Cohens d	Effect-size r
13.64	12	0.00 (<.05)	7.39	0.965

The $Search_{Fitness}$ group (N = 7) is associated with a fitness value (M= 0.761, SD=0.006). By comparison, the $Manual_{Fitness}$ group (N = 7) is associated with a numerically smaller fitness value (M=0.498, SD=0.05). To test the hypothesis that the $Search_{Fitness}$ in comparison with $Manual_{Fitness}$ is associated with statistically significantly different mean fitness values, an independent samples t-test is performed. The independent samples t-test is associated with a statistically significant effect in comparison with the two methods; the Fitness t(12) = 13.64, p ¡ 0.05. Thus, the $Search_{Fitness}$ samples are associated with a statistically significant larger mean fitness value than the $Manual_{Fitness}$ samples. Cohen's d is estimated at 7.39 with effect-size r estimated at 0.965, which is a large effect based on Cohens guidelines [46]. Based on the results and analysis, the search method outperforms the manual method. The search is a fully automated comprehensive top-down framework that comprises all the phases of the service identification. This fully-automated framework supports the agility, reuse and composability of SOA services [47], [48]. In addition, it is concluded that the representation method, with the associated genetic operators, has implemented the service constraints such that the resulting solutions are feasible and conform to the SOA principles of low coupling and high cohesion.

VI. CONCLUSIONS

In this paper, different challenges that face service identification methods have been identified, not least the cognitive difficulties of manual service identification by software engineers. To address these challenges, a novel metaheuristic search framework has been developed using a genetic algorithm to support the software engineer for the totality of service identification activities. Moreover, this framework has provided computationally intelligent support for the identification of services from business processes models applied to a designated Cancer Care at the King Hussein Cancer Center. The comprehensive framework takes a set of BPMN models as input, frames them at an appropriate granularity level, and then formulates a search space with traceable building blocks, or 'search space elements' ($SSEs$). In a subsequent step, a genetic algorithm explores and exploits the search space to arrive at optimal candidate services that adhere to SOA principles such as coupling and cohesion.

It is concluded that role-based business process models such as BPMN 2.0 can indeed be mapped to Service-Oriented Architectures, and the mapping has been realised by a comprehensive framework that derives Cancer Care services from BPMN models. The representation of the search space elements ensures the feasibility of resulting candidate service solutions.

Experiments show that the search-based technique for service identification proposed in this paper has been configured and parameters tuned to achieve promising exploration and exploitation of the search space to find effective solutions. In comparison with the services produced manually by domain experts, the search technique has produced more effective service solutions.

Overall, the findings of this paper suggest that the metaheuristic search-based framework provides an effective and novel technique to derive Cancer Care services from BPMN models. Evaluation by domain experts confirmed that the

resulting services are feasible and useful solutions that the domain experts might not have arrived at had they been identifying the services manually. Indeed, statistical analysis shows candidate services produced by the search-based framework are superior to the services produced manually by domain experts at KHCC with respect to metrics for coupling and cohesion.

In future research, we plan to extend the impact of this work in two areas. Firstly, we plan to expand the generalisability of the research by examining the performance of the framework in other health-care problem domains and beyond. Secondly, it would be worthwhile to investigate how domain expert insight and preferences might be incorporated within the computationally intelligent search to combine qualitative evaluation of the domain expert with the quantitative fitness selection in the genetic algorithm.

ACKNOWLEDGEMENT

The authors would like to thank our colleagues from the King Hussein Cancer Centre who provided insight and expertise that greatly assisted this research. We would also like to show our gratitude to Dr Asem Mansur for supporting this project, Mrs Einas Jazairy and Mr Mohannad Abdulmalik for facilitating the experimental sessions. Our great thanks as well to all the domain experts for taking time out of their busy schedule to participate in the experiments.

REFERENCES

[1] A. T. Zadeh, M. Mukhtar, S. Sahran, and Z. Lotfi, "Automated service identification framework (asif)." *Journal of Theoretical & Applied Information Technology*, vol. 83, no. 3, 2016.

[2] R. Yousef, O. Adwan, and M. A. Abushariah, "Extracting soa candidate software services from an organizations object oriented models," *Journal of Software Engineering and Applications*, vol. 7, no. 09, p. 770, 2014.

[3] R. M. Yousef, "Bpaontosoa: A semantically enriched framework for deriving soa candidate software services from riva-based business process architecture," Ph.D. dissertation, University of the West of England, Bristol, 2010.

[4] R. Yousef, M. Odeh, D. Coward, and A. Sharieh, "Translating rad business process models into bpmn models: A semi-formal approach." *IJWA*, vol. 3, no. 4, pp. 187–196, 2011.

[5] P. Jamshidi, S. Mansour, K. Sedighiani, S. Jamshidi, and F. Shams, "An automated service identification method," Technical Report, TR-ASER-2012-01. Automated Software Engineering Research Group, Shahid Beheshti University, Tech. Rep., 2012.

[6] A. Kazemi, A. Rostampour, P. Jamshidi, E. Nazemi, F. Shams, and A. N. Azizkandi, "A genetic algorithm based approach to service identification," in *Services (SERVICES), 2011 IEEE World Congress on*. IEEE, 2011, pp. 339–346.

[7] L. G. Azevedo, F. Santoro, F. Baião, J. Souza, K. Revoredo, V. Pereira, and I. Herlain, "A method for service identification from business process models in a soa approach," in *Enterprise, Business-Process and Information Systems Modeling*. Springer, 2009, pp. 99–112.

[8] M. Harman and B. F. Jones, "Search-based software engineering," *Information and software Technology*, vol. 43, no. 14, pp. 833–839, 2001.

[9] C. L. Simons, J. Smith, and P. White, "Interactive ant colony optimization (iaco) for early lifecycle software design," *Swarm Intelligence*, vol. 8, no. 2, pp. 139–157, 2014.

[10] M. Harman, "Software engineering meets evolutionary computation," *Computer*, no. 10, pp. 31–39, 2011.

[11] A. Ramirez, J. R. Romero, and C. Simons, "A systematic review of interaction in search-based software engineering," *IEEE Transactions on Software Engineering*, 2018.

[12] M. Odeh and R. Kamm, "Bridging the gap between business models and system models," *Information and Software Technology*, vol. 45, no. 15, pp. 1053–1060, 2003.

[13] K. Klose, R. Knackstedt, and D. Beverungen, "Identification of services-a stakeholder-based approach to soa development and its application in the area of production planning." in *ECIS*, vol. 7, 2007, pp. 1802–1814.

[14] Y. Kim and K.-G. Doh, "Formal identification of right-grained services for service-oriented modeling," in *International Conference on Web Information Systems Engineering*. Springer, 2009, pp. 261–273.

[15] O. Patashnik. (2016, Feb.) Xml web services. [Online]. Available: http://www.w3schools.com/xml/xml_services.asp

[16] Z. Zheng, Y. Zhang, and M. R. Lyu, "Investigating qos of real-world web services," *IEEE transactions on services computing*, vol. 7, no. 1, pp. 32–39, 2014.

[17] K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan, "A fast and elitist multiobjective genetic algorithm: Nsga-ii," *IEEE transactions on evolutionary computation*, vol. 6, no. 2, pp. 182–197, 2002.

[18] K. Deb and H. Jain, "An evolutionary many-objective optimization algorithm using reference-point-based nondominated sorting approach, part i: Solving problems with box constraints." *IEEE Trans. Evolutionary Computation*, vol. 18, no. 4, pp. 577–601, 2014.

[19] S. Khoshnevis and F. Shams, "Automating identification of services and their variability for product lines using nsga-ii," *Frontiers of Computer Science*, vol. 11, no. 3, pp. 444–464, 2017.

[20] D. E. Goldberg, "Genetic algorithm," *Search, Optimization and Machine Learning*, pp. 343–349, 1989.

[21] M. Bowman, L. C. Briand, and Y. Labiche, "Solving the class responsibility assignment problem in object-oriented analysis with multi-objective genetic algorithms," *IEEE Transactions on Software Engineering*, vol. 36, no. 6, pp. 817–837, 2010.

[22] T. Back, *Evolutionary algorithms in theory and practice: evolution strategies, evolutionary programming, genetic algorithms*. Oxford university press, 1996.

[23] A. E. Eiben, J. E. Smith *et al.*, *Introduction to evolutionary computing*. Springer, 2003, vol. 53.

[24] T. Blickle and L. Thiele, "A comparison of selection schemes used in evolutionary algorithms," *Evolutionary Computation*, vol. 4, no. 4, pp. 361–394, 1996.

[25] R. Abd Rahman, R. Ramli, Z. Jamari, and K. R. Ku-Mahamud, "Evolutionary algorithm with roulette-tournament selection for solving aquaculture diet formulation," *Mathematical Problems in Engineering*, vol. 2016, 2016.

[26] T. Bäck, D. B. Fogel, and Z. Michalewicz, *Evolutionary computation 1: Basic algorithms and operators*. CRC press, 2000, vol. 1.

[27] R. Malhotra, N. Singh, and Y. Singh, "Genetic algorithms: Concepts, design for optimization of process controllers," *Computer and Information Science*, vol. 4, no. 2, p. 39, 2011.

[28] P. Guo, X. Wang, and Y. Han, "The enhanced genetic algorithms for the optimization design," in *Biomedical Engineering and Informatics (BMEI), 2010 3rd International Conference on*, vol. 7. IEEE, 2010, pp. 2990–2994.

[29] H. Jain, N. Chalimeda, N. Ivaturi, and B. Reddy, "Business component identification-a formal approach," in *Enterprise Distributed Object Computing Conference, 2001. EDOC'01. Proceedings. Fifth IEEE International*. IEEE, 2001, pp. 183–187.

[30] K. Praditwong, M. Harman, and X. Yao, "Software module clustering as a multi-objective search problem," *IEEE Transactions on Software Engineering*, vol. 37, no. 2, pp. 264–282, 2011.

[31] I. Vanderfeesten, J. Cardoso, J. Mendling, H. A. Reijers, and W. van der Aalst, "Quality metrics for business process models," *BPM and Workflow handbook*, vol. 144, pp. 179–190, 2007.

[32] M. Kessentini, W. Kessentini, H. Sahraoui, M. Boukadoum, and A. Ouni, "Design defects detection and correction by example," in *2011 19th IEEE International Conference on Program Comprehension*. IEEE, 2011, pp. 81–90.

[33] M. P. Papazoglou and W.-J. Van Den Heuvel, "Service-oriented design and development methodology," *International Journal of Web Engineering and Technology*, vol. 2, no. 4, pp. 412–442, 2006.

[34] W. Khlif, N. Zaaboub, and H. Ben-Abdallah, "Coupling metrics for business process modeling," *WSEAS Transactions on Computers*, vol. 9, no. 1, pp. 31–41, 2010.

[35] H. A. Reijers and I. T. Vanderfeesten, "Cohesion and coupling metrics for workflow process design," in *International Conference on Business Process management*. Springer, 2004, pp. 290–305.

[36] P. Jamshidi, S. Khoshnevis, R. Teimourzadegan, A. Nikravesh, and F. Shams, "An automated method for service specification," in *Proceedings of the Warm Up Workshop for ACM/IEEE ICSE 2010*. ACM, 2009, pp. 25–28.
[37] W. Khlif, L. Makni, N. Zaaboub, and H. Ben-Abdallah, "Quality metrics for business process modeling," in *Proceedings of the 9th WSEAS international conference on Applied computer science*. World Scientific and Engineering Academy and Society (WSEAS), 2009, pp. 195–200.
[38] C. L. Simons, I. C. Parmee, and R. Gwynllyw, "Interactive, evolutionary search in upstream object-oriented class design," *IEEE Transactions on Software Engineering*, vol. 36, no. 6, pp. 798–816, 2010.
[39] O. Seng, J. Stammel, and D. Burkhart, "Search-based determination of refactorings for improving the class structure of object-oriented systems," in *Proceedings of the 8th annual conference on Genetic and evolutionary computation*. ACM, 2006, pp. 1909–1916.
[40] T. Kim, K. Lee, and J. Baik, "An effective approach to estimating the parameters of software reliability growth models using a real-valued genetic algorithm," *Journal of Systems and Software*, vol. 102, pp. 134–144, 2015.
[41] F. Abu Rub, M. Odeh, I. Beeson, D. Pheby, and B. Codling, "Modelling healthcare processes using role activity diagramming," *International Journal of Modelling and Simulation*, vol. 28, no. 2, pp. 147–155, 2008.
[42] M. Ahmad, "Semantic derivation of enterprise information architecture from riva-based business process architecture," Ph.D. dissertation, University of the West of England, 2016.
[43] W. M. Trochim and J. Donnelly, *Research methods: The concise knowledge base*. Atomic Dog Publishing Cincinnati, OH, 2005.
[44] F. Andy, "Discovering statistics using spss for windows: Advanced techniques for the beginner," 2000.
[45] F. J. Gravetter and L. B. Wallnau, *Essentials of Statistics for the Behavioral Sciences (PSY 200 (300) Quantitative Methods in Psychology)*. Boston: Cengage Learning, 2010.
[46] J. Cohen, "A power primer." *Psychological bulletin*, vol. 112, no. 1, p. 155, 1992.
[47] M. Papazoglou, *Web services: principles and technology*. Pearson Education, 2008.
[48] T. Erl, *SOA design patterns*. Pearson Education, 2008.