ISSN (Print): 0974-6846 ISSN (Online): 0974-5645

# Multi-Objective Optimization of Feature Model in Software Product Line: Perspectives and Challenges

### Asad Abbas, Isma Farah Siddiqui and Scott Uk-Jin Lee\*

Department of Computer Science and Engineering, Hanyang University, South Korea; asadabbas@hanyang.ac.kr, isma2012@hanyang.ac.kr, scottlee@hanyang.ac.kr

#### **Abstract**

Software Product Line (SPL) is process for developing families of software with reusability of features categorized as common and variable features. Feature Model (FM) is developed to manage these features. Common features are easy to manage, however variable features are hard to manage because of complex relations and constraints between features. Optimization is required to manage the variabilities for best selection of features and product configurations. To this end, different Multi-Objective Evolutionary Algorithms have been proposed to get the optimal solutions of feature model. In this paper we have compared among three main optimization algorithms i.e. IBEA, NSGA-II and MOCell. Our comparison is based on previous research correctness solutions for product's configuration with five objective functions on different feature models from SPLOT and LVAT repositories. The goal of this comparison is to find the current research prospective and challenges of multi-objective optimization in FM.

**Keywords:** Feature Model, Optimization of Feature Model, Software Product Line

### 1. Introduction

Software Product Line (SPL) is extensively used in current software development industry because of reusability of components, time to market; reduce cost, high productivity and quality of product<sup>1-4</sup>. SPL is an effective approach that share common and variable resources for the development of software families. Domain engineering and application engineering are two major processes in SPL. All common and variable resources of SPL are developed in domain engineering process and produce multiple products by reusability of those resources in application engineering<sup>5.6</sup>. Common features (resources) are used for every product of SPL and variable features are used by the requirement of each product. Common features are easy to manage and reuse however, variable features are hard to handle as these features differentiates all products of SPL<sup>7,8</sup>.

Feature Model is most efficient and best approach to manage the commonalities and variability of SPL. FM

is structured in tree structure that contains all common and variable features. It differentiates variable feature by different relationships such as alternative, optional and OR<sup>9,10</sup>. Variable features are hard to manage because of relationship (include or exclude) with other features<sup>11</sup>. Commonalities and variability information is not enough to develop products of SPL. Therefore, binding information among features is also important for products derivation i.e. when, how and where specific features are required. Feature binding information can be analyzed from different perspectives such as, which features can be bound (product requirements), when these features are required (features relationship) and how features can be bound (designing techniques) for final product derivation<sup>12</sup>.

Feature Oriented Domain Analysis (FODA) approach is mostly used to manage the variability of FM. Cardinality based FM is based on FODA that acquires attribute information of variable features. Feature attributes contains functional and non-functional information of SPL fam-

ily<sup>13</sup>. Attributes can be textual or numeric that determines when and where these features are required in product derivation<sup>14</sup>. Another approach for variability management of FM is goal model which is extensively used in recent years. Goal model is used to manage the functional and non-functional attributes of every feature in FM and products configuration of SPL. Goal model represents the intentional variability model to provide functional and non-functional aspects of domain engineering<sup>15</sup>.

Selection of best features for product configuration according to user requirements is based on relationships and attributes values of every feature 16.17. Optimal configuration is best paradigm to select the features on the user requirements. Therefore, optimization empowers the user preference to select best features for product derivation<sup>18</sup>. Single objective optimization is simple as it concentrates on single objective given by user. However, multi-objective optimization is complex as it achieves more than one objective for single product derivation and all objectives need to satisfy user's requirement at a same time. Furthermore, it is nearly impossible that all objectives satisfy a single solution set of features. Trade-off between objectives is required to get close solutions that satisfy the all objectives<sup>20</sup>. In recent research different approaches and algorithms are used for optimization of SPL feature model. However, three algorithms IBEA, NSGA-II and MOCell are most effective and efficient for optimization of SPL in current research. In this paper we briefly unfold the comparative study of achievements and current research on challenges faced in Feature Model optimization. The comparison is done by examining the efficiency and correctness of product configuration results by each algorithm. Comparison of these algorithms is based on the correct optimization of FM with five objectives for each algorithm on same FMs.

### 2. Foundations

Software Product Line framework is adopted that is part of ISO/IEC #26550 standard in Figure 1.

#### 2.1 Software Product Line Processes

Domain engineering process consists of further multiple processes that are used for domain development of SPL as shown in Figure 1. Domain engineering describes the complete scope with all requirements of SPL<sup>20</sup>. However application engineering is further development of each

product by selecting the features from domain engineering according to user's requirements. Application engineering is the final process of products derivation in such way each product differentiate with other products on at least one variation point i.e. variability<sup>21</sup>.

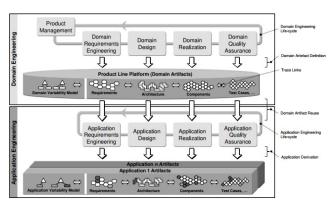


Figure 1. SPL framework<sup>20</sup>.

#### 2.3 Attribute Feature Model

Figure 2 shows a sample FM and identifies possible set of relationships among features. The relationships serves as constraints in FM. Main objective of FM is to get the compact picture of SPL with commonalty and variability of products. Outcome of FM is the systematic management of common and variable features<sup>22</sup>.

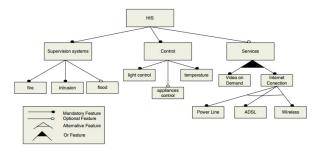


Figure 2. Feature Model<sup>22</sup>.

In Figure 2, possible relationship among features are described as below<sup>5,23</sup>:

- **Mandatory**: Core part of every product and must be selected for all products.
- Optional: Based on the requirement of user, optional feature may or may not be select for product derivation.
- **Alternative:** Only one feature can be selected from number of features.
- **Or-relation**: At least one feature must be select from OR group relation.

 Attribute Feature Model is an extension of FM that represents the functional and non-functional attribute values of every feature such as COST and Deterministic Time (DTIME) as shown in Figure 3.

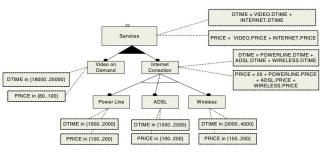


Figure 3. Attribute feature model<sup>22</sup>.

Selecting suitable feature for product derivation according to user requirements is based on these functional and non-functional attribute values. Optimization of FM to select the best features combination for product development is applicable on attribute FM. In Multi-Objective Optimization every feature should consist of more than one attribute values to select the best features combination 24,25.

# 3. Feature Model Multi-Objective Optimization

Beat feature selection for product derivation problem was first described in Filtered Cartesian Flattening<sup>25</sup>. Main objective for feature selection is to improve and maximize the accuracy that is proposed by user e.g. cost of product should not increase than a defined cost. This problem indicates an optimal feature selection with resource constraints. Multi-Objective Optimization satisfy multiple objectives simultaneously such as minimization of product cost and memory consumption and maximization of performance<sup>26</sup>. Constraint violation may occur during optimization because of complex relationships (include or exclude) and constraints (alternative, optional, OR groups) in FM. For correct and efficient product configuration according to end user's requirements, optimization methods and approaches needs to avoid any constraint violation. In current research, different algorithms have been proposed for optimized feature selection such as, Indicator Based Evolutionary Algorithm (IBEA)27, Nondominated Sorting Genetic Algorithm (NSGA-II)29 and

Multi-Objective Cellular Genetic Algorithm (MOCell)<sup>30</sup> have been proposed for Multi-Objective Optimization. Therefore, we compared these three algorithms from current research for Multi-Objective Optimization of FM.

### 3.1 Indicator Based Evolutionary Algorithm (IBEA)

IBEA proposed by Zitzler in 2004 for multi-objective optimization is based on fitness value of objectives<sup>27</sup>. IBEA calculate fitness value of every objective and compare with other solution sets and find the dominated solution based on the indicator value. IBEA is efficient that favors user objective for optimization<sup>28</sup>.

### 3.2 Non-dominated Sorting Genetic Algorithm (NSGA-II)

NSGA-II is effective to get best optimized feature selection based on Genetic Algorithm (GA). GA is appropriate for single objective optimization however; NSGA-II gets optimal solutions for multi-objective problems. All possible solutions were compared to find non-dominated solutions for minimization or maximization of objective functions. Non-dominated solutions are pareto-front solutions that cannot be dominated by any other solution set<sup>29</sup>.

### 3.3 Multi-Objective Cellular Genetic Algorithm (MOCell)

MOCell was proposed in 2009 is multi-objective algorithm based on canonical Genetic Algorithm (cGA). MOCell initiates optimization with empty pareto-front and take two individual solution sets and make comparison on the fitness values to get non-dominated solutions. Comparison continues until termination criteria get ful-filled<sup>30</sup>.

Procedure comparison between IBEA, NSGA-II and MOCell is elaborated in Table 1. In this section we present MOCell, our proposed multi-objective algorithm based on the cGA model.

# 4. Product Configuration Optimization: Comparative Study

In comparative study we consider seven FMs from current research. Among these, three are small FMs from

Table 1. Procedure comparison of IBEA, NSGA-II and MOCe
---

Algorithms	Population	Basic Operators	Domination	Objective
IBEA	Main(One)+archive	Crossover, Mutation and Environmental Selection (Based on Indicator)	Quality indicator decide the domination of solutions i.e. Epsilon, HyperVolume etc.	Favor Indicator value defined by user i.e. user reference
NSGA-II	Main	Crossover, Mutation and Tournament Selection (Objectives)	Closest distance from each objective functions is favorable i.e. higher fitness value.	Discrete solutions i.e. overall dominating
MOCell	Main(One)+archive	Crossover, Mutation, Tournament Selection and Random feedback	Select the solutions according to crowding distance and ranking	Absolute domination same as NSGA-II

SPLOT repository and four are large FMs from LVAT repository as shown in Table 2.

Table 2. Feature Models Repository

Repository	Feature Model	#Features	Cross Tree Constraints (CTCs)
SPLOT	E-Shop	290	21
	Web Portal	43	6
	JCS	12	13
LVAT	Linux	6888	343944
	uClinux	1850	2468
	eCos	1244	3146
	Fiasco	1638	5228

Our comparative study is based on five objective functions adapted from previous researches to examine optimization challenges for FM.

The five objective functions are:

• **Correctness:** Correctness of product configurations i.e. constraint violation and relationship in FM.

**Table 3.** Result comparison from current research

•	Richness of Features:	Minimization	of dese-
	lected features.		
•	<b>Defects:</b> Minimize the d	lefects that are l	known in

- Defects: Minimize the defects that are known in FM.
- **Reusability of Features:** Minimize the features that were not part of any product.
- **Cost:** Minimize the total cost.

Table 3 shows the comparative study with correctness of product configurations of small FM from SPLOT by using IBEA, NSGA-II and MOCell. Correctness results with five objective functions represent the quality of optimized product configurations. Results indicate that IBEA performance for E-Shop and Web portal is 100% that means no constraint violation occurs. However for JCS, the correctness results are 86% and 98% 33.34.

At this point of discussion, we are convinced to identify the first challenge in current research of multiobjective optimization of SPL feature model.

 Challenge 1: Determine a General Algorithm for optimization of FM that satisfies all constraints and relationships to configure correct products.

Algorithms	Feature Models	%Correctness with 5 Objectives				
		[31]	[29]	[32]	[33]	[34]
	E-Shop	52%	37%	66.8%	100%	100%
IBEA	Web Portal	73%	98%	-	100%	100%
	JCS	-	-	-	86%	98%
	E-Shop	1%	22%	2.4%	12%	12%
NSGA-II	Web Portal	8%	20%	-	28%	28%
	JCS	-	-	-	27%	27%
	E-Shop	1%	-	-	14%	14%
MOCell	Web Portal	4%	-	-	41%	41%
	JCS	-	-	-	34%	34%

Table 3 shows that IBEA is more efficient algorithm for optimization of FM than NSGA-II and MOCell. However, IBEA also not fully confirm with no constraint violation in JCS FM optimization.

Complexity of optimization is increased as objective functions are increased. The increment also impact on search space as well as number of relationships and constraints between features. Trade-off between objectives creates complexity with large number of objectives. Considering this, the second and third challenges are identified to target important aspect of multi-objective optimization, given below as:

- **Challenge 2:** For multi-objective optimization, determination of number of objectives supported by algorithms to get correct product configurations without constraint violations even on a single point.
- Challenge 3: Determine maximum number of features and constraints in FM that can be correctly optimized with Multi-Objective Evolutionary Algorithms (MOEAs).

Table 4 is based on the comparative study of optimization in large FMs from LVAT repository. Linux X86 has 6888 total number of features and large number of constraints i.e. 343944. Results show 0% correctness with IBEA on Linux X86 i.e. constraint violations occurs and thus not feasible for large FM. However, uClinux has comparatively less number of features and constraints and the product configurations are 100% correct.

Table 4. Large Feature Models from LVAT

Algorithm	Feature Models	%Correctness with 5 Objectives		
		[35]	[33]	[34]
IBEA	Linux X86	0%	0%	0%
	uClinux	31%	100%	100%
	eCos	2%	91%	100%
	Fiasco	100%	-	-

It is also elaborated in Table 4 that foreCosFM33,35, the correctness of product configuration is 2% and 91% that clearly shows the constraint violation. Similarly, uClinux<sup>35</sup> optimization correctness with IBEA is 31%. Constraints violation can be resolved if find the location of violation occurrence in optimal configuration results to get 100% correctness, because of complex relationships manual searching of violations is not possible. This assumption leads us to argue for fourth challenge for Multi-Objective Optimization, given as:

• Challenge 4: Determine criteria to identify the location of violation occurrence in large FM.

### 5. Discussion

In this paper we compared the study of three well-known Multi-Objective Evolutionary Algorithms (MOEAs) that are IBEA, NSGA-II and MOCell for optimization of small and large FMs with less and high constraints. Correctness comparisons shows that IBEA is more efficient and effective algorithm for optimization of large and small FMs. IBEA gives the preference to user objectives on the basis of indicator value that specifies the direction of optimization. However, NSGA-II and MOCell prefer the non-dominated solutions by the comparison of solution sets with each other.

### 6. Conclusion

In this paper we demonstrate the processes of SPL and FM to manage the commonalities and variabilities for product configurations. Attribute Feature Model is used to demonstrate the functional and non-functional attribute values of every feature to determine the product configurations according to user's requirement. Optimization is imperative to get best feature combinations to develop an optimum product. We compared three state-of-art algorithms and results and determined the research challenges related to the correctness of optimal solutions and constraint violations in FM. We also identify IBEA is most suitable optimization algorithm with less constraint violations and high correctness than NSGA-II and MOCell.

### 7. References

- 1. Boonon P, Muenchaisri P. An approach to clustering Feature Model based on adaptive behavior for dynamic software product line. 2014 IEEE International Conference on Information Science and Applications (ICISA); 2014 May. p. 1-4.
- Nie K, Zhang L, Geng Z. Product line variability modeling based on model difference and merge. IEEE 36th Annual Computer Software and Applications Conference Workshops (COMPSACW); 2012 Jul. p. 509-13.

- 3. Zheng L, Zhang C, Wu Z, Liu M. Managing resource repository of a Software Product Line with Feature Model. IEEE International Conference on Computational Intelligence and Software Engineering, CiSE; 2009 Dec. p. 1–4.
- 4. Loesch F, Ploedereder E. Optimization of variability in software product lines. IEEE 11th International Software Product Line Conference (SPLC); 2007 Sep. p. 151–62.
- Abbas A, Wu Z, Siddiqui IF, Lee SUJ. An approach for optimized feature selection in software product lines using union-find and Genetic Algorithms. Indian Journal of Science and Technology. 2016 May; 9(17):1–8.
- 6. Mohan K, Ramesh B, Sugumaran V. Integrating Software Product Line engineering and agile development. IEEE Software. 2010; 27:3:48–55.
- Ripon S, Azad K, Hossain SJ, Hassan M. Modeling and analysis of product-line variants. Proceedings of the 16th International Software Product Line Conference, ACM; 2012 Sep; 2: 26–31.
- 8. Sinnema M, Deelstra S, Nijhuis J, Bosch J. Managing variability in software product families. Proceedings of the 2nd Groningen Workshop on Software Variability Management; 2004 Dec.
- Holdschick H. Challenges in the evolution of modelbased software product lines in the automotive domain. Proceedings of the 4th International Workshop on Feature-Oriented Software Development, ACM; 2012 Sep. p. 70–3.
- Czarnecki K, Grunbacher P, Rabiser R, Schmid K, Wąsowski A. Cool features and tough decisions: A comparison of variability modeling approaches. Proceedings of the Sixth International Workshop on Variability Modeling of Software-Intensive Systems, ACM; 2012 Jan. p. 173–82.
- Lee K, Kang KC, Kim M, Park S. Combining featureoriented analysis and aspect-oriented programming for product line asset development. IEEE 10th International Software Product Line Conference (SPLC); 2006 Aug. p. 10.
- 12. Lee J, Kang KC. Feature binding analysis for product line component development. International Workshop on Software Product-Family Engineering. Springer Berlin Heidelberg; 2003 Nov. p. 250–60.
- Lee SUJ. An effective methodology with automated product configuration for Software Product Line development. Mathematical Problems in Engineering. 2015; 2015: 11 pages.
- 14. Sinnema M, Deelstra S. Classifying variability modeling techniques. Information and Software Technology. 2007; 49(7):717–39.
- 15. Noorian M, Bagheri E, Du W. Capturing non-functional properties through model interlinking. IEEE 27th Canadian Conference on Electrical and Computer Engineering (CCECE); 2014 May. p. 1–6.

- 16. Rosenmuller M, Siegmund N. Automating the configuration of multi software product lines. VaMoS. 2010; 10:123–30.
- 17. Korner M, Herold S, Rausch A. Composition of applications based on software product lines using architecture fragments and component sets. Proceedings of the WICSA, Companion Volume, ACM; 2014 Apr. p. 13.
- 18. Yamany E, Eid A, Shaheen M, Sayyad A.S. OPTI-SELECT: An interactive tool for user-in-the-loop feature selection in software product lines. 18th International Software Product Line Conference: Companion Volume for Workshops, Demonstrations and Tools, ACM. 2014 Sep; 2014:126–9.
- 19. Karimpour R, Ruhe G. Evolutionary robust optimization for Software Product Line scoping: An explorative study. Computer Languages, Systems and Structures; 2016.
- Metzger A, Pohl K. Software Product Line engineering and variability management: Achievements and challenges. Proceedings of the on Future of Software Engineering; 2014 May. p. 70–84.
- Thum T, Kastner C, Benduhn F, Meinicke J, Saake G, Leich T. Feature IDE: An extensible framework for feature-oriented software development. Science of Computer Programming. 2014; 79:70–85.
- Benavides D, Trinidad P, Ruiz-Cortes A. Automated reasoning on feature models. International Conference on Advanced Information Systems Engineering, Springer Berlin Heidelberg; 2005 Jun. p. 491–503.
- 23. Wang YL, Pang JW. Ant colony optimization for feature selection in software product lines. Journal of Shanghai Jiaotong University (Science). 2014; 19:50–8.
- 24. 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(12):2208–21.
- 25. White J, Doughtery B, Schmidt DC. Filtered cartesian flattening: An approximation technique for optimally selecting features while adhering to resource constraints. SPLC (2); 2008 Oct. p. 209–16.
- Henard C, Papadakis M, Harman M, Le Traon Y. Combining multi-objective search and constraint solving for configuring large software product lines. IEEE/ACM 37th IEEE International Conference on Software Engineering. 2015 May; 1:517–28.
- Zitzler E, Kunzli S. Indicator-based selection in multiobjective search. International Conference on Parallel Problem Solving from Nature, Springer Berlin Heidelberg; 2004 Sep. p. 832–42.
- Strickler A, Lima JAP, Vergilio SR, Pozo AT. Deriving products for variability test of Feature Models with a hyperheuristic approach. Applied Soft Computing, 2016 Dec; 49:1232–42.

- 29. Sayyad AS, Goseva-Popstojanova K, Menzies T, Ammar H. On parameter tuning in search based software engineering: A replicated empirical study. IEEE 3rd International Workshop Replication in Empirical Software Engineering Research (RESER); 2013 Oct. p. 84–90.
- Nebro AJ, Durillo JJ, Luna F, Dorronsoro B, Mocell AE: A cellular Genetic Algorithm for multi-objective optimization. International Journal of Intelligent Systems. 2009; 24(7):726–46.
- 31. Sayyad AS, Menzies T, Ammar H. On the value of user preferences in search-based software engineering: A case study in software product lines. IEEE 35th International Conference on Software Engineering (ICSE); 2013 May. p. 492–501.
- 32. Sayyad AS, Ingram J, Menzies T, Ammar H. Optimum feature selection in software product lines: Let your model and

- values guide your search. IEEE 1st International Workshop on Combining Modeling and Search-Based Software Engineering (CMSBSE); 2013 May. p. 22–7.
- 33. Xue Y, Zhong J, Tan TH, Liu Y, Cai W, Chen M, Sun J. IBED: Combining IBEA and DE for optimal feature selection in Software Product Line engineering. Applied Soft Computing, 2016 Dec; 49:1215–31.
- 34. Tan TH, Xue Y, Chen M, Sun J, Liu Y, Dong JS. Optimizing selection of competing features via feedback-directed evolutionary algorithms. International Symposium on Software Testing and Analysis, ACM; 2015 Jul. p. 246–56.
- 35. Sayyad AS, Ingram J, Menzies T, Ammar H. Scalable product line configuration: A straw to break the camel's back. IEEE/ACM 28th International Conference on Automated Software Engineering (ASE); 2013 Nov. p. 465–74.