Mining Correlations of ATL Model Transformation and Metamodel Metrics

Juri Di Rocco*, Davide Di Ruscio*, Ludovico Iovino* and Alfonso Pierantonio*
*DISIM - Unversità degli Studi dell'Aquila - Italy
Email: name.surname@univaq.it

***Abstract*—Model transformations are considered to be the "heart" and "soul" of Model Driven Engineering, and as a such, advanced techniques and tools are needed for supporting the development, quality assurance, maintenance, and evolution of model transformations. Even though model transformation developers are gaining the availability of powerful languages and tools for developing, and testing model transformations, very few techniques are available to support the understanding of transformation characteristics. In this paper, we propose a process to analyze model transformations with the aim of identifying to what extent their characteristics depend on the corresponding input and target metamodels. The process relies on a number of transformation and metamodel metrics that are calculated and properly correlated. The paper discusses the application of the approach on a corpus consisting of more than 90 ATL transformations and 70 corresponding metamodels.**

I. Introduction

In Model Driven Engineering (MDE) model transformations play a key role since they permit to bridge together different abstraction levels by automatically mapping source models to target ones. In [1] model transformations are considered to be the "heart" and "soul" of MDE, and as a such, they require to be treated in a similar way as traditional software artifacts [2]. Thus, dedicated techniques and tools for supporting the development, quality assurance, maintenance, and evolution of model transformations are not an option. Even though model transformation developers are gaining the availability of powerful languages and tools for developing, testing, and chaining model transformations, very few techniques are available for analysing model transformations [2], [3]. In particular, there is still the need for techniques able to support the understanding of common characteristics of model transformations, e.g., what are the main constructs that are typically used when developing transformations, and to what extent the development of model transformations is affected by the complexity of the corresponding metamodels.

In this paper we propose an analysis process contributing to the understanding of model transformations by correlating metrics calculated on both transformations and metamodels they are defined on. The identified correlations permit to draw interesting considerations e.g., what is the typically employed model transformation development style (i.e., declarative vs imperative), how a model transformation is typically structured depending on the considered metamodels (i.e., if they are general purpose or domain specific), and how the complexity of metamodels has an impact on the overall model transformations development. Such considerations can be preparatory to further analysis that are very common in software development [4], e.g., estimating the effort required to develop model transformations by considering the structural characteristics of the source and target metamodels. The proposed process has been applied on a corpus of 91 ATL [5] transformations and 72 corresponding metamodels. To this end we have considered 28 metamodel metrics and 35 transformation metrics. The elements in the corpus have been collected from several academic and open repositories.

The paper is structured as follows: Section II describes the process we have conceived to analyse model transformations and to correlate both transformation and metamodel metrics. Interesting correlations are discussed in Section III. Section IV relates the proposed approach with existing works, whereas Section V concludes the paper and mentions some interesting hints for future works.

II. Measuring transformations

Over the last years, software metrics have been proposed to assess and predict software effort and quality [6] and recent works have proposed the adoption of metrics to measure transformations. Specifically, metrics on transformations have been investigated with the aim of understanding transformations via quantitative evaluations. For instance, in [3] specific metrics have been conceived to measure ATL transformations, and in [7] authors define the meaning of several quality attributes in the context of model transformations and align them to a set of metrics.

The adoption of metrics to measure metamodels has been recently proposed in [8], [9], [10]. In [8] authors apply object-oriented measurements to understand common structural characteristics of metamodels, whereas [9] proposes a measuring mechanism for assessing the quality of metamodels. In [10] we correlate metamodel metrics to understand common characteristics in metamodeling (e.g., abstraction, inheritance, and composition) that might increase the complexity of metamodels and hamper their adoption and evolution in modeling ecosystems [11]. To the best of our knowledge, none of the existing approaches calculate transformation metrics with the aim of correlating them.

Since it is reasonable to claim that the complexity of model transformations is somehow related to that of the source and

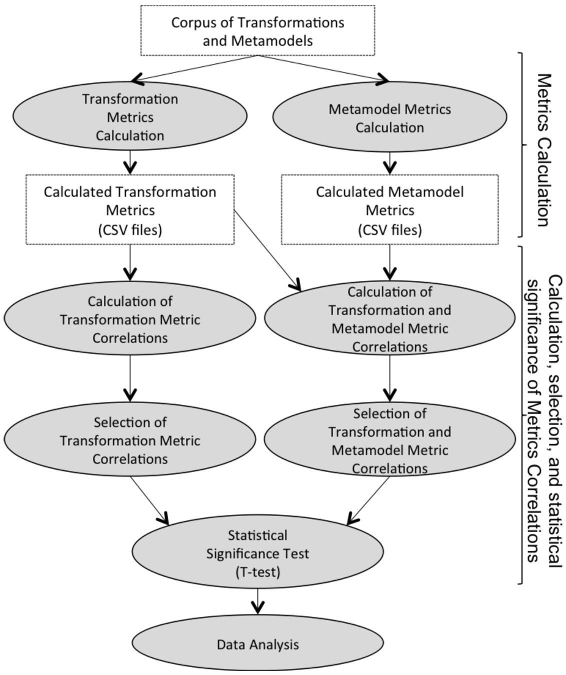

Fig. 1: Proposed analysis process

target metamodels, in our opinion in order to have a complete measurement of model transformations, it is necessary to identify also possible correlations among transformation and metamodels metrics, e.g., to figure out to what extent the number of matched rules of given ATL transformations depends on the number of metaclasses in the source and/or target metamodels. To this end, in this section the measurement process shown in Fig. 1 is presented. In particular, the first step of the process consists of applying a number of metrics on an available corpus of transformations and corresponding metamodels. Afterwards the calculated transformation metrics are correlated among them by using statistical tools. Finally, the collected data are analysed in order to cross link structural characteristics of transformations, e.g., how the different kinds of ATL rules (i.e., matched, lazy, and called) are typically used. Moreover, metamodel metrics are also considered in order to identify possible correlations among transformation and metamodel metrics (e.g., how the number of metaclasses in the target metamodel impacts the structural characteristics of transformations in terms of number of matched rules, helpers, etc.).

In the remaining of this section all the steps of the process shown in Fig. 1 are discussed. The last step of the process is discussed separately in the next section.

A. Metrics Calculation

The first step of the process consists of the application of transformation and metamodel metrics on the considered corpus. To perform our analysis we have created a corpus of transformations and corresponding metamodels that have been collected from the web, forges, repositories and academic projects, i.e., EMFText Zoo, ATLZoo, Github, GoogleCode, for a total of 91 transformations, and 72 related metamodels. Analyzed metamodels are equally distributed in complexity and size, in particular we worked with a range of: 0-5 metaclasses for about 12 metamodels, 5-10 metaclasses for about 23 metamodels, 10-20 metaclasses for about 15 metamodels, more than 20 metaclasses for about 22 metamodels.

Metrics have been calculated by exploiting a model-based tool chain that we have developed [12] and already applied in other similar analysis [10], [13]. Essentially, metrics are calculated by means of ATL transformations whose target models conform to the metric metamodel initially proposed in [14] and subsequently refined in [10]. The involved ATL transformations have been inspired by the transformation in [14] and they mainly consist of couples of lazy rules and helpers for each metric to be calculated. The output of the whole calculation process consists of Comma Separated Values (CSV) files that enable the adoption of statistical tools like IBM SPSS and MS Excel.

As previously mentioned the total number of transformation metrics that have been calculated in our analysis is 35 and some of them are shown in Table I. Concerning the metrics that have been applied on the corresponding metamodels they are in total 28 and some of them are shown in Table II.

B. Calculation, selection and statistical significance of metric correlations

Correlation is probably the most widely used statistical method to detect cross-links and assess relationships among observed data. There are different techniques and indexes to discover and measure correlations. In the following we overview the Pearson's and Spearman's coefficients that we have considered in this paper to measure the correlations among calculated metamamodel metrics [10].

The *Pearson's correlation coefficient* [15] was developed by Karl Pearson from a related idea introduced by Francis Galton in the 1880s. It is widely used in the sciences as a measure of the degree of linear dependence between two variables. In particular, the Pearson correlation coefficient is appropriate when it is possible to draw a regression line between the points of the available data (e.g., see the diagrams A and B in Fig. 2).

The *Spearman's correlation coefficient* [16] was used by Charles Spearman in the 1900s in the psychology domain. This coefficient is better than Pearson to manage situations when there is a monotonic relationship between the considered variables. For instance, in the cases shown in the diagrams C and D in Fig. 2, the Pearson coefficient would wrongly identify a very low correlations among the considered data. This is due to the fact that the assumption of linear relationships required by Pearson is not satisfied. Contrariwise, Spearman's correlation index would perform better in cases of monotonic relationships as in the diagrams C and D in Fig. 2[1]. It is also important to note that the assumption of a monotonic relationship is less restrictive than a linear relationship (an assumption that has to be met by the Pearson correlation). For this reason, we

[1] All scattered plot diagrams shown in this paper use date logarithmic scale for empathize the correlation

Acronym	Name	Description
B	Number of Bindings	Number of bindings in all output pattern
IP	Number of Input Pattern	The metric number of input pattern elements measures the size of the input pattern of rules. Note that since called rules do not have an input pattern, the metric number of input model elements does not include called rules.
OP	Number of Output Pattern	The metric number of output pattern elements measure the size of the output pattern of rules.
TR	Number of Transformation Rules	A measure for the size of a model transformation is the number of transformation rules it encompasses. In ATL, there are different types of rules, viz., matched rules, lazy matched rules, unique lazy matched rules, and called rules.
MR	Number of Matched Rules (Excluding Lazy Matched Rules)	Number of matchad rule exzcluding lazy matched rule. If this matrics are equals to number of transformation rule the transformation are defined *completely declarative*
LR	Number of Lazy Matched Rules (Including Unique)	Number of lazy rule including unique
CR	Number of Called Rules	Number of Called Rules
RWF	Number of Rules with a Filter Condition on the Input Pattern	Number of rules with a filter condition on the input pattern. The input pattern has a condition. This implies that not all model elements in the source model may be transformed.
RWD	Number of Rules with a Do Section	ATL allows the definition of imperative code in rules in a do block. This can be used to perform calculations that do not fit the preferred declarative style of programming. To measure the use of imperative code in a transformation, we defined number of rules with a do section
RWU	Number of Rules with a Using clause	ATL allows the definition of local variable in a rule. This can be used to perform calculations that do not fit the preferred declarative style of programming. To measure the use of imperative code in a transformation, we defined number of rules with a using clause
AH	Attribute helper	Total number of attribute helpers in the transformation
H	Number of Helper	Total number of helpers in the transformation
HWC	Number of Helpers with Context	Number of helper with context in the transformation
HNC	Number of Helpers without Context	Number of helper without context in the transformation
CRT	Number of Calls to ResolveTemp()	The resolveTemp() function is used to look-up references to non-default output elements of other rules. Therefore, it is to be expected that model transformations with a large number of calls to the resolveTemp() function are harder to understand.

TABLE I: 15 out of 35 metrics for measuring ATL transformations

Acronym	Name	Description
AMC	Number of Abstract MetaClass	Number of metaclasses that cannot be instantiated in models
CMC	Number of Concrete MetaClass	Number of metaclasses that can be directly instantiated
MC	Number of Total MetaClass	Number of metaclasses in the metamodel (MC = AMC + CMC)
SF	Number of Structural Features	Number of attributes and references in the metamodel

TABLE II: 4 out of 28 metrics used for measuring metamodels

use Spearman only for highlighting curvilinear correlations. Both Pearson's and Spearman's correlation indexes assume values in the range of -1.00 (perfect negative correlation) and +1.00 (perfect positive correlation). A correlation with value 0 indicates that the two considered variables are not correlated. In order to assess the strength of correlations it is possible to consider the guide that Evans [17] suggests for the *absolute value* of the correlation indexes, i.e., [*0.0*,*0.19*] very weak, [*0.20*,*0.39*] weak, [*0.40*,*0.59*] moderate, [*0.60*,*0.79*] strong, and [*0.80*,*1.0*] very strong.

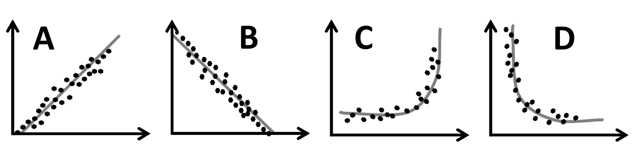

Fig. 2: Examples of scattered plots

Just because two variables are related, it does not necessarily mean that one directly causes the other. In particular, when performing statistical analysis that take into account and relate members of given populations, it is necessary to assess that the produced outcome is statistically significant or not. To this end, the *significance level* of a statistical hypothesis refers to the probability that the random sample that has been chosen is not representative [18]. Thus, the lower the significance level, the more confident one can be in replicating the performed results [19]. In our measurements, to establish if the identified (Pearson's and Spearman's) correlation coefficients are statistically significant we make use of the T-test [20]. Usually the significance value can be established by the experimenter but typically this value is set as 0.05 or 0.01. In our analysis we have considered the T-test significance value as 0.05 and we have calculated it on the correlated metrics by using the T-test implementation available in MS Excel.

1) ATL Model transformation metric correlations: Once the transformation metrics have been calculated, the most correlated one are identified and selected. In particular, we have calculated the Pearson's correlation indexes for all the values of the transformation metrics. The results of this operation is a correlations matrix as the one shown in Table III. For instance, the number of output patterns of a transformation (OP) is strongly correlated with the number of bindings (B) as testified by their Pearson's correlation index having value 0.88.

2) ATL Model transformation and metamodel metric correlations: The interesting part of our analysis relies on correlating model transformation and metamodel metrics. To this

	B	IP	OP	TR	MR	LR	CR	RWF	RWD	RWC	AH	H	HWC	HNC	CRT
B	1,000														
IP	0,413	1,000													
OP	0,883	0,340	1,000												
TR	0,478	0,938	0,319	1,000											
MR	0,475	0,925	0,299	0,983	1,000										
LR	0,094	0,512	0,142	0,524	0,386	1,000									
CR	0,365	0,152	0,264	0,231	0,164	-0,061	1,000								
RWF	0,392	0,818	0,208	0,862	0,852	0,461	0,140	1,000							
RWD	0,247	0,329	0,165	0,372	0,397	-0,090	0,344	-0,024	1,000						
RWC	0,461	0,112	0,384	0,150	0,163	-0,117	0,267	0,191	-0,022	1,000					
AH	0,036	0,127	0,013	0,155	0,147	0,071	0,114	0,179	0,025	-0,032	1,000				
H	0,183	0,345	0,218	0,303	0,260	0,374	0,009	0,362	-0,079	-0,009	0,680	1,000			
HWC	0,191	0,302	0,237	0,239	0,197	0,374	-0,061	0,303	-0,112	-0,019	0,523	0,938	1,000		
HNC	0,070	0,264	0,064	0,291	0,270	0,183	0,160	0,312	0,035	0,020	0,685	0,629	0,321	1,000	
CRT	0,540	-0,053	0,602	-0,022	-0,029	-0,083	0,202	-0,041	0,010	0,436	0,159	0,119	0,070	0,169	1,000

TABLE III: Pearson Correlation values related to transformation metrics

end a correlation matrix based on the Spearman's index has been calculated and a fragment of it is shown in Table IV. For instance, according to the calculated matrix, the number of output patterns (OP) of a model transformation is strongly related with the number of metaclasses (MC) contained in the output metamodel.

	B	IP	OP	TR	MR	LR	CR	RWF	RWD	H	HWC	HNC	CRT	
MC	0,450	**0,692**	0,450	0,450	0,410	0,300	0,200	0,520	0,300	-0,010	-0,090	0,160	0,090	INPUT
AMC	0,330	0,470	0,320	0,410	0,380	0,270	0,180	**0,640**	0,280	0,070	-0,030	0,220	0,000	
CMC	0,470	0,510	0,490	0,470	0,420	0,290	0,260	0,270	0,340	0,030	-0,040	0,180	0,000	
SF	0,259	0,313	0,205	0,216	0,212	0,121	0,041	0,132	0,277	-0,040	-0,140	0,140	0,050	
MC	0,520	0,542	**0,783**	**0,746**	0,500	0,223	0,369	0,480	0,399	0,180	0,168	0,204	0,131	OUTPUT
AMC	0,485	0,455	0,524	0,454	0,424	0,218	0,368	0,419	0,372	0,226	0,162	0,321	0,158	
CMC	0,523	0,570	0,513	0,560	0,510	0,258	0,351	0,508	0,399	0,191	0,171	0,195	0,143	
SF	**0,806**	0,430	0,428	0,449	0,392	0,221	0,383	0,421	0,368	0,190	0,149	0,203	0,110	

TABLE IV: Spearman Correlation values related to ATL transformation and metamodel metrics

III. Data analysis

In this section we discuss some relevant correlations we have identified as described in the previous section. Because of the lack of space it is not possible to discuss all the identified correlations that include the metrics shown in Table I and Table II. However, interested readers can refer to the spreadsheet available onlinecontaining all the obtained results. By considering the correlations of both transformation and metamodel metrics (see Section III-A), interesting considerations can be drawn about how structural characteristics of metamodels affect the structure of the corresponding model transformations. Even by considering correlations involving only ATL metrics it is possible to obtain insights e.g., about how the constructs of the ATL language are typically used by developers (see Section III-B).

A. How metamodel characteristics affect model transformations

By exploiting the matrix obtained by correlating transformation and metamodel metrics, in this section we discuss how metamodels affect the development of model transformations. The discussion is based on the correlation matrix shown in Table IV and by considering the most interesting correlations having value greater than 0.60 (thus strong or even very strong).

1) How transformation rules are influenced by target metamodels: This aspect can be investigated by considering the correlation between the number of metaclasses (MC) in the target metamodel (OUT MC) and the number of transformation rules (TR). Such two values are correlated because of the Spearman's index having value 0.746 and the T-test significance value 0,000229581.

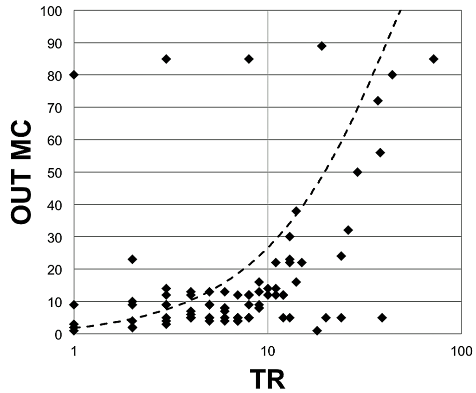

Fig. 3: How transformation rules are influenced by the number of metaclasses in target metamodel

The plot in Fig. 3 represents how these two values are influenced each other. We note that both the number of the metaclasses (MC) in the target metamodel and the number of transformation rules (TR) tend to increase together in the figure, potentially indicating some causality. This is generally true since the transformation development is typically output-driven when the developer tries to cover all the metaclasses of the target metamodel. We can also state that the common concentration in the corpus is in the range between 1 and 20 metaclasses, and 1 and 15 transformation rules, indicating the declarative style of transformation as common choice of developers.

2) How the total number of transformation input patterns are influenced by the source metamodels: As anticipated in the previous sections the metric IP (number of input patterns of the considered transformations) is related with the metric MC (number of metaclass) in the source metamodel. This is confirmed by the Spearman's correlation of 0.692 and significance value 0.0001. In the plot shown in Fig. 4 the distribution is less clear than the previous case but the trend is similar: the values of MC and IP seem to increase together. Again this confirms the use of the declarative style for developing the ATL transformations included in our corpus.

3) How the total number of transformation output patterns are influenced by the target metamodels: According to the calculated matrix, the Spearman's correlation index between the value of the metrics OP (number of output patterns) in the transformation rules and the number of metaclasses (MC) in the target metamodels has value 0.783. However, even though the correlation index is high the T-test significance value is 0.419 thus we are not sure if the correlation is actually existing or if it depends on the corpus that we have used. In such cases it is necessary to add new transformations in order to extend the corpus. As discussed in Section V this represents a task that we intend to do in the future.

4) How the structural features in the target metamodel influence the number of bindings: The Spearman correlation

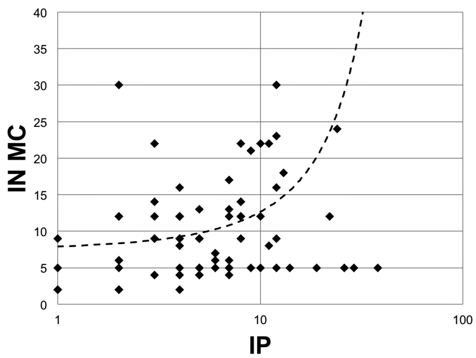

Fig. 4: How the total number of input patterns are influenced by the source metamodels

Cluster	Number of transformations	MR	LR	CR	RWF	RWD	RWU	HWC	HNC	CRT
DSL2DSL	37	86,06%	11,50%	2,44%	47,39%	17,42%	2,44%	69,66%	30,34%	38,54%
GPL2GPL	17	91,00%	0,00%	9,00%	71,50%	18,50%	3,00%	26,32%	73,68%	39,58%
DSL2GPL	15	90,00%	9,58%	0,42%	39,58%	18,75%	1,25%	47,20%	52,80%	21,88%
GPL2DSL	22	91,77%	5,06%	3,16%	75,32%	19,94%	0,95%	69,61%	30,39%	0,00%

TABLE V: How general purposes and domain specific metamodels affect the complexity of the model transformations

index for the number of structural features (SF) of the target metamodel and the number of bindings (B) written in the rules of the transformations is 0.808 and the T-test significance value is 0.07. We note that both SF in the output metamodels and B tend to increase together in Fig. 5 potentially indicating some causality.

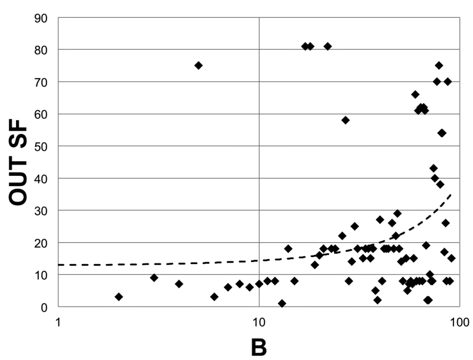

Fig. 5: How the structural features in the target metamodel influence the number of bindings

5) How general purpose and domain specific metamodels affect the complexity of model transformations: In order to understand how the complexity of model transformations depends on the kind of corresponding metamodels we partitioned our metamodel corpus by distinguishing general purpose languages (GPL) and domain specific languages (DSL). For instance, the Java metamodel is in the GPL part, whereas the BPMN metamodel is in the DSL part. The transformations in our corpus were partitioned as shown in Fig. 6. We have considered such a distribution to calculate the values of the representative metrics shown in Table V.

According to Table V the use of filter conditions in the rules of `GPL2GPL` and `GPL2GPL` transformations is higher than in the other cases. This might depend on the fact that in such kinds of transformations

MR	LR	CR	HWC	HNC
89.6 %	6.9%	3.5%	65.0%	35.0%

TABLE VI: Use of rules and helpers in ATL transformations

only parts of metamodels and hierarchies are considered and consequently filter conditions in rules are required.

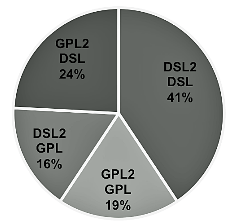

Fig. 6: Distribution of transformations

The use of the imperative *"do"* block is higher in the `DSL2GPL` and `GPL2DSL` than the other cases. This represents the fact that typically imperative constructs are used when the input and output metamodels are completely different. The use of the *"clause"* construct is very limited and it is one of the less used constructs of ATL.

Finally, we report that the use of *resolveTemp* seems to be never used in `GPL2DSL` transformations and equally distributed in other cases. In particular, it is used in only 22 transformations resulting to be one of the most complex construct of the language.

B. How developers use the ATL language

By considering only transformation-specific metrics and thus the correlation matrix in Table III it is possible to do a number of considerations. In particular, the number of output pattern (OP) and the number of bindings (B) tend to increase together and both have an impact on the complexity of the transformation. The Pearson index for them has value 0.883 and the T-test significance value is 0.000000009. Such a correlation can be explained as the fact that the more the transformation is complex and verbose, including a large number of OP, the more the number of bindings increases (and vice versa). As expected, the number of transformation rules is linked to the number of input patterns. In fact the Pearson's correlation index for TR and IP is 0.938. Unfortunately, the T-test significance value for this result is 0.426 and consequently, we need to extend our corpus to be sure that the identified correlation is actually existing.

The calculated ATL specific metrics permit to draw considerations also related to the *declarativeness* and the *imperativeness* of transformations. This is particularly important since ATL developers are typically encouraged to use the declarative style of the language. In particular, there are some metrics like the number of matched rules (MR) indicating how declarative constructs have been used in contrast to the number of *do blocks* indicating the imperative use of ATL. The value of such metrics applied on our corpus are shown in Table VI. According to such values most of the transformations available in our corpus are developed in a declarative way. In fact 86.6% of the transformation rules are matched ones. Moreover, helpers with contexts (HWC) are more than those without contexts (HNC), which are typically used as variables in transformations developed in an imperative way.

IV. Related Work

In [3] authors introduce metrics to measure the quality of ATL transformations without considering metamodel aspects. In [21] authors have focused on transformation model measurements in order to better understand transformations via a quantitative evaluation, like the declarative factor of modules and rules. One of the most inspiring papers on the argument of this work is [2] that proposes an approach consisting of three main building blocks: *i)* metrics are used to acquire quick insights into transformations; *ii)* two different ways are proposed to visualize dependencies between the components of a transformation; *iii)* coverage analysis is proposed to visualize the coupling of transformations with their corresponding metamodels. In [22] an analogous approach is proposed to measure model repositories. By means of specific metrics, in [23] authors investigate factors that might have impact on the execution performance of model transformations. In [24] Van Amstel et al. propose a set of six quality attributes for evaluating the quality of model transformations. In [25] authors discuss how model transformations can improve the quality of models by means of metrics.

Differently to what has been presented in this paper, all the previously mentioned works propose the adoption of metrics to measure quality attributes of transformations without considering metamodel aspects. Williams et al. in [8] discuss metrics related to a large metamodel collection exposing how metamodels are commonly structured, and how they evolve over time. A similar approach for understanding structural characteristics of metamodels and their relationships has been presented in [10].

V. Conclusions and Future Work

In this paper we proposed an approach to analyze model transformations by considering also the corresponding metamodels. The approach relies on the correlation of different metrics and has been applied on a corpus of 90 transformations and 70 corresponding metamodels. In the future we intend to extend our corpus in order to aim at increasing the significance value for those correlations that currently are below the threshold. Moreover, we intend to extend the work by including in the analysis further kinds of artifacts typically involved in any MDE approach. We intend also to investigate how it is possible to apply metrics to perform early estimations of the development costs of model transformations by analyzing the corresponding metamodels. The significance of the obtained results will be assessed by considering also the adoption of alternative statistical approaches [26]. The main goal will be to qualitatively investigate actual causalities, or to validate the identified correlations.

References

[1] Sendall, S., Kozaczynski, W.: Model Transformation: The Heart and Soul of Model-Driven Software Development. IEEE Softw. **20** (2003) 42–45

[2] van Amstel, M., van den Brand, M.: Model Transformation Analysis: Staying Ahead of the Maintenance Nightmare. In: Theory and Practice of Model Transformations. Volume 6707. Springer Berlin Heidelberg (2011) 108–122

[3] Vignaga, A.: Metrics for Measuring ATL Model Transformations. Technical report (2009)

[4] Boehm, B., Abts, C., Chulani, S.: Software development cost estimation approaches A survey. Annals of Software Engineering **10** (2000) 177–205

[5] Jouault, F., Allilaire, F., Bézivin, J., Kurtev, I.: ATL: A model transformation tool. Science of Computer Programming **72** (2008) 31–39

[6] Fenton, N.E., Pfleeger, S.L.: Software Metrics: A Rigorous and Practical Approach. 2nd edn. PWS Publishing Co., Boston, MA, USA (1998)

[7] van Amstel, M., Lange, C., van den Brand, M.: Metrics for analyzing the quality of model transformations. Procs. of the 12th ECOOP Workshop QAOOSE (2008) 41–51

[8] James, W., Athansios, Z., Nicholas, M., Louis, R., Dimitios, K., Richard, P., Fiona, P.: What do metamodels really look like? Frontiers of Computer Science (2013)

[9] Ma, Z., He, X., Liu, C.: Assessing the quality of metamodels. Frontiers of Computer Science **7** (2013) 558–570

[10] Di Rocco, J., Di Ruscio, D., Iovino, L., Pierantonio, A.: Mining Metrics for Understanding Metamodel Characteristics. MiSE '14, ACM (2014) 55–60

[11] Di Ruscio, D., Iovino, L., Pierantonio, A.: Evolutionary Togetherness: How to Manage Coupled Evolution in Metamodeling Ecosystems. ICGT'12, Springer-Verlag (2012) 20–37

[12] Basciani, F., Di Rocco, J., Di Ruscio, D., Di Salle, A., Iovino, L., Pierantonio, A.: MDEForge: an extensible Web-based modeling platform. In: CloudMDE Workshop at MODELS '14. (2014)

[13] Basciani, F., Di Ruscio, D., Iovino, L., Pierantonio, A.: Automated Chaining of Model Transformations with Incompatible Metamodels. In: Procs. International Conference on Model Driven Engineering Languages and Systems (MODELS '14). Volume 8767., Springer International Publishing (2014) 602–618

[14] van Amstel, M.F., van den Brand, M.: Quality assessment of ATL model transformations using metrics. In: Procs. of the MtATL '10 Workshop. (2010)

[15] Lee Rodgers, J., Nicewander, W.A.: Thirteen ways to look at the correlation coefficient. The American Statistician **42** (1988) 59–66

[16] Spearman, C.: The proof and measurement of association between two things. The American journal of psychology **15** (1904) 72–101

[17] Evans, J.S.B., Over, D.E.: Rationality and reasoning. Psychology Press (2013)

[18] Steel, R.G., Torrie, J.H., et al.: Principles and procedures of statistics. Principles and procedures of statistics. (1960)

[19] Rice, W.R.: Analyzing tables of statistical tests. Evolution (1989) 223–225

[20] Student: The probable error of a mean. Biometrika (1908) 1–25

[21] Tolosa, J.B., Sanjuán-Martínez, O., García-Díaz, V., Lovelle, J.M.C., et al.: Towards the systematic measurement of ATL transformation models. Software: Practice and Experience **41** (2011) 789–815

[22] Vépa, E., Bézivin, J., Brunelière, H., Jouault, F.: Measuring model repositories. In: Procs. of MSM '06. (2006)

[23] Van Amstel, M., Bosems, S., Kurtev, I., Pires, L.F.: Performance in model transformations: experiments with ATL and QVT. In: Procs. ICMT '11. Springer (2011) 198–212

[24] Amstel, M.F., Lange, C.F., Brand, M.G.: Using Metrics for Assessing the Quality of ASF+SDF Model Transformations. ICMT '09 (2009) 239–248

[25] Saeki, M., Kaiya, H.: Measuring Model Transformation in Model Driven Development. In: CAiSE Forum. Volume 247. (2007)

[26] Courtney, R., Gustafson, D.: Shotgun correlations in software measures. Software Engineering Journal **8** (1993) 5–13