Model reevaluation based on graph transformation rules

M. Gaide[1], D. Marcheix[2], A. Arnould[3], X. Skapin[3], H. Belhaouari[3] and S. Jean[2]

[1]ISAE-ENSMA Poitiers, Université de Poitiers, LIAS, Poitiers, France
[2]Université de Poitiers, ISAE-ENSMA Poitiers, LIAS, Poitiers, France
[3]Université de Poitiers, Univ. Limoges, CNRS, XLIM, Poitiers, France

Abstract
In this paper, we extend the scope of naming problem studies to encompass rule-based graph transformation modeling systems. We propose a novel persistent naming method that capitalizes on the formalized operations of generalized maps and graph transformation rules. It enables a unique and homogeneous characterisation of entities across all dimensions.

CCS Concepts
• Computing methodologies → Shape modeling;

Keywords *: Topology-based modeling; Graph transformation rules; Persistent Naming; Reevaluation; Generalized maps;*

1. Introduction

The ability to generate multiple variants of an object during a construction process has become increasingly prevalent across various application domains. Most of the time, tools and operations used to create those variants are dedicated to specific fields and the construction process is often both tedious and time-consuming [QB15; HMV09; MWH*]. In the realm of computer-aided design (CAD), parametric history-based systems have long been utilized, recording the successive operations used during the construction process. This latter can then be reevaluated after performing some slight modifications upon the operations, obviating the need to restart from scratch. Such an approach requires addressing the well-known issue of *persistent naming* [Kri97]. Indeed, adding, deleting, moving operations in the parametric specification or modifying some operation parameters require ensuring that the subsequent operations are still valid even if their own parameters have been affected. This entails using *persistent identifiers* as operations parameters, which make possible to unambiguously characterize entities and find their match at reevaluation. Persistent naming has been studied for decades in the CAD's field [Kri97; CCH96; WZZZ01; WN05; BNB05; MH05; Mar06; Bab10; XJH*16; FH18]. To our knowledge, only one approach has studied this problem in the domain of graph transformations, but it is a preliminary study that considers only a part of topological entities' history [CMS*19].

In this paper, we propose to extend this preliminary study in order to define a full persistent naming mechanism, using a rule-based graph transformation formalism, and more specifically the Jerboa software [BAGB14]. Contrary to other approaches, Jerboa is independent from any specific application field and does not require *ad hoc* operations to be manually coded. Operations are formally defined as rules within the Jerboa interface, thereby facilitating their

rapid development. Furthermore, it guarantees the topological consistency of the underlying geometric model, regardless of the applied operations [ABB*22]. It is based on the generalized maps (or G-maps) topological model [Lie91; DL14]. This model represents a specific class of labelled graph and allows the homogeneous modeling of quasi-manifolds in any dimension. Number of applications already make use of Jerboa and/or G-maps in fields such as plant growth [BTG15], architecture [HMDB09; ALS15], geology [HH00] or physics-based modeling [SBAM].

Our contribution is two-fold. First, we extend the scope of naming problem studies to encompass rule-based graph transformation modeling systems. Second, we integrate the mechanisms of reevaluation for parametric systems into Jerboa. The persistent naming method proposed takes advantage of the rule-based formalization of operations and their ability to precisely describe the history of topological entities, such that these entities are uniquely and homogeneously characterized for all dimensions. Most existing methods require tracking numerous topological entities and consider the persistent naming problem only through the prism of parameters modifications from a parametric specification standpoint [Kri97; CH95; WZZZ01]. Our solution tracks only the entities used in the parametric specification and the ones they originate from. Moreover, not only the naming problem is tackled within the usual framework of parameter edition, but we also take the specification edition (*i.e.* adding, deleting and changing the order of operations) into account.

This paper is organized as follows. Section 2 presents the main concepts necessary to carry out the persistent naming mechanisms within the framework of a system, making use of graph transformation rules. The following sections describe our persistent naming system and how a parametric specification is evaluated or reevaluated through directed acyclic graphs.

2. Main Concepts

Generalized maps (or G-maps) [Lie91; DL14] allow the representation of manifold geometric objects (with or without boundaries), based on a cellular *n*-dimensional topological structure. The representation of an object as a G-map comes intuitively from its decomposition into topological cells (vertices, edges, faces, volumes, and so on). The lowest topological entity in a G-map is called a *dart*. A dart is represented as a subgraph called an *orbit*. For example, a volume orbit is noted $\langle 0, 1, 2 \rangle$, a volume face $\langle 0, 1 \rangle$, a volume edge $\langle 0, 2 \rangle$ and a volume vertex $\langle 1, 2 \rangle$.

Jerboa's [BAGB14] rules describe the transformation of a G-map *G* into another G-map *H* by describing a subgraph to match in *G* and the subgraph replacing it in *H*. These subgraphs respectively represent the left and right members of a rule. A rule's member is made up of nodes describing a G-map's orbit (see Fig.1). Perfor-
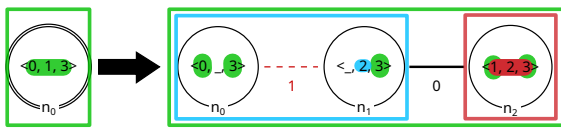

Figure 1: *Face triangulation rule*

ming a syntactical analysis of rules allows to characterize the topological changes affecting an orbit throughout an application. Thus, it is possible to automatically detect topological changes of an orbit and to track it automatically throughout a construction. For example, in Fig.1, the rule splits the face orbit $\langle 0, 1, 3 \rangle$ in the green box, creates the vertex orbit $\langle 1, 2, 3 \rangle$ in the red box and modifies the volume vertex orbit $\langle 1, 2 \rangle$ in the blue box.

3. Persistent Name

Since Jerboa's graph transformation rules use darts as topological parameters, it follows that each persistent name must represent a unique dart. As it happens, rules make it possible to determine unambiguously which node filters or creates any dart. Therefore, a persistent name, noted *PN*, is a list of all the right nodes involved in a darts' history, coupled with a rule application's number. For example, $PN_3 = [1n_3; 2n_4; 3n_0]$ in Fig.2d. Indeed, the dart 35 used to designate the face to be colored was created through the nodes n_3 of 1-create rule, n_4 of 2-extrude rule, and n_0 of 3-triangulation rule.
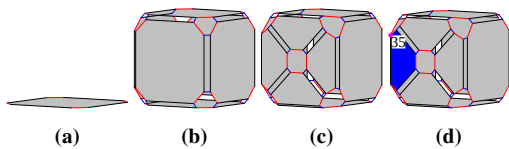
4. Evaluation

Figure 2: *Evaluation. (a) 1-create(pos); (b) 2-extrude(PN$_1$, vec); (c) 3-triangulate(PN$_2$); (d) 4-color(PN$_3$)*

The evaluation is based on a Directed Acyclic Graph (or DAG) to compute each persistent name before any reevaluation can take
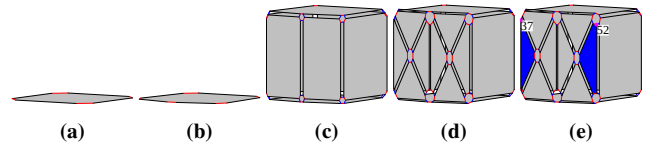

Figure 3: *Reevaluation. (a) 1-create(pos); (b) ADD1-insert(3); (c) extrude(PN$_1$, vec); (d) triangulate(PN$_2$); (e) 4-color(PN$_3$)*

place. The DAG is built bottom-up by a backward traversal through the persistent name. The role of an evaluation DAG is to trace the history of a topological parameter back to the first created orbits it originates from, thus allowing matching a corresponding topological parameter at reevaluation. In order to accurately represent the
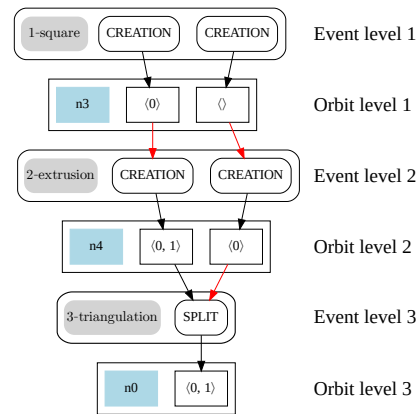

Figure 4: *PN$_3$'s evaluation DAG*

history of an orbit, two types of arrows are used in an evaluation DAG (and later in the reevaluation DAG). A black arrow , or *trace*, represents the evolution of an orbit throughout the modeling process. A red arrow, or *origin*, represents another orbit the tracked orbit originates from. For example, The DAG in Fig. 4 represents the history the topological parameter designated as PN_3 in Fig.2d which is the face being colored. This history contains the trace and origin orbits as well as their changes (creation, split, modification, and so on).

5. Reevaluation

Since topological parameters may change after editing a parametric specification, it is necessary to build reevaluation DAGs from evaluation DAGs in order to update topological parameters. Reevaluation DAGs can designate one, several, or no orbit depending on the editing of the parametric specification.

Reevaluation DAGs are built top-down. While an evaluation DAG represents the orbit's history of a topological parameter, the reevaluation DAG represents the history of this same orbit after editing the parametric specification. Such an edit yields a different DAG at reevaluation (with event levels and/or branches being added, deleted or both). Several matching strategies can then be considered depending on the application's context, such as the
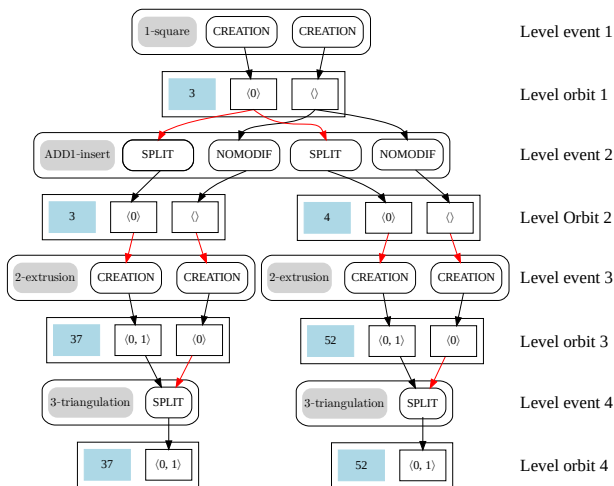
Figure 5: *PN₃'s reevaluation DAG*

DAG in Fig. 5 representing the histories of the faces to color in blue in Fig.3e.

6. Conclusion

This paper introduces our work on extending the scope of naming problem studies to encompass rule-based graph transformation modeling systems. We implement a persistent naming scheme identifying a unique dart's history. We represent any topological parameter's history with an evaluation DAG from a persistent name. We can match one, several or no topological parameter depending on the editing to reevaluate.

More complex operations can make use of several rules brought together in a script. Later works will revolve around widening this reevaluation mechanism to scripts.

References

[ABB*22] ARNOULD, AGNÈS, BELHAOUARI, HAKIM, BELLET, THOMAS, et al. "Preserving consistency in geometric modeling with graph transformations". *Mathematical Structures in Computer Science* 32.3 (2022), 300–347 1.

[ALS15] ARROYO OHORI, KEN, LEDOUX, HUGO, and STOTER, JANTIEN. "A dimension-independent extrusion algorithm using generalised maps". *International Journal of Geographical Information Science* 29.7 (2015), 1166–1186 1.

[Bab10] BABA-ALI, MEHDI. "Systeme de nomination hierarchique pour les systemes parametriques". PhD thesis. 2010. URL: http://http://theses.univ-poitiers.fr/notice/view/5362 1.

[BAGB14] BELHAOUARI, HAKIM, ARNOULD, AGNÈS, GALL, PASCALE LE, and BELLET, THOMAS. "Jerboa: A graph transformation library for topology-based geometric modeling". *International Conference on Graph Transformation*. Springer. 2014, 269–284 1, 2.

[BNB05] BIDARRA, RAFAEL, NYIRENDA, PAULOS J, and BRONSVOORT, WILLEM F. "A feature-based solution to the persistent naming problem". *Computer-Aided Design and Applications* 2.1-4 (2005), 517–526 1.

[BTG15] BOHL, EVANS, TERRAZ, OLIVIER, and GHAZANFARPOUR, DJAMCHID. "Modeling fruits and their internal structure using parametric 3Gmap L-systems". *The Visual Computer* 31 (2015), 819–829 1.

[CCH96] CAPOYLEAS, VASILIS, CHEN, XIANGPING, and HOFFMANN, CHRISTOPH. "Generic naming in generative, constraint-based design". *Computer-Aided Design* 28.1 (1996), 17–26 1.

[CH95] CHEN, XIANGPING and HOFFMANN, CHRISTOPH M. "On editability of feature-based design". *Computer-aided design* 27.12 (1995), 905–914 1.

[CMS*19] CARDOT, ANAIS, MARCHEIX, DAVID, SKAPIN, XAVIER, et al. "Persistent naming based on graph transformation rules to reevaluate parametric specification". *Computer-Aided Design and Applications* 16.5 (2019), 985–1002 1.

[DL14] DAMIAND, G. and LIENHARDT, P. *Combinatorial Maps: Efficient Data Structures for Computer Graphics and Image Processing*. A K Peters/CRC Press, 2014 1, 2.

[FH18] FARJANA, SHAHJADI HISAN and HAN, SOONHUNG. "Mechanisms of persistent identification of topological entities in CAD systems: A review". *Alexandria engineering journal* 57.4 (2018), 2837–2849 1.

[HH00] HALBWACHS, YVON and HJELLE, ØYVIND. "Generalized maps in geological modeling: Object-oriented design of topological kernels". *Advances in Software Tools for Scientific Computing*. Springer. 2000, 339–356 1.

[HMDB09] HORNA, SÉBASTIEN, MENEVEAUX, DANIEL, DAMIAND, GUILLAUME, and BERTRAND, YVES. "Consistency constraints and 3D building reconstruction". *Computer-Aided Design* 41.1 (2009), 13–27 1.

[HMV09] HAEGLER, SIMON, MÜLLER, PASCAL, and VAN GOOL, LUC. "Procedural modeling for digital cultural heritage". *EURASIP Journal on Image and Video Processing* 2009 (2009), 1–1 1.

[Kri97] KRIPAC, JIRI. "A mechanism for persistently naming topological entities in history-based parametric solid models". *Computer-Aided Design* 29.2 (1997). Solid Modelling, 113–122. ISSN: 0010-4485 1.

[Lie91] LIENHARDT, PASCAL. "Topological models for boundary representation: a comparison with n-dimensional generalized maps". *Computer-aided design* 23.1 (1991), 59–82 1, 2.

[Mar06] MARCHEIX, DAVID. "A persistent naming of shells". *International Journal of CAD/CAM* 6.1 (2006), 125–137 1.

[MH05] MUN, DUHWAN and HAN, SOONHUNG. "Identification of topological entities and naming mapping for parametric cad model exchanges". *International Journal of CAD/CAM* 5.1 (2005), 69–81 1.

[MWH*] MÜLLER, PASCAL, WONKA, PETER, HAEGLER, SIMON, et al. "Procedural modeling of buildings". *ACM SIGGRAPH 2006 Papers*, 614–623 1.

[QB15] QUATTRINI, RAMONA and BALEANI, ELEONORA. "Theoretical background and historical analysis for 3D reconstruction model. Villa Thiene at Cicogna". *Journal of Cultural Heritage* 16.1 (2015), 119–125 1.

[SBAM] SALAH, FATMA BEN, BELHAOUARI, HAKIM, ARNOULD, AGNÈS, and MESEURE, PHILIPPE. "A general physical-topological framework using rule-based language for physical simulation". *12th International Conference on Computer Graphics Theory and Application (VISIGRAPP/GRAPP 2017)* 1.

[WN05] WANG, YAN and NNAJI, BARTHOLOMEW O. "Geometry-based semantic ID for persistent and interoperable reference in feature-based parametric modeling". *Computer-Aided Design* 37.10 (2005), 1081–1093 1.

[WZZZ01] WU, JUNJUN, ZHANG, TIANBING, ZHANG, XINFANG, and ZHOU, JI. "A face based mechanism for naming, recording and retrieving topological entities". *Computer-Aided Design* 33.10 (2001), 687–698 1.

[XJH*16] XUE-YAO, GAO, JIA-QI, LI, HAO, GUO, et al. "Name and Maintain Topological Faces in Rotating and Scanning Features". *International Journal of Grid and Distributed Computing* 9 (2016), 21–26 1.