A first step towards the inference of geological topological operations

R. Pascual[1] and H.Belhaouari[2] and A.Arnould[2] and P. Le Gall[1]

[1]MICS, CentraleSupélec, University Paris Saclay, France
[2]XLIM UMR CNRS 7252, University of Poitiers, France

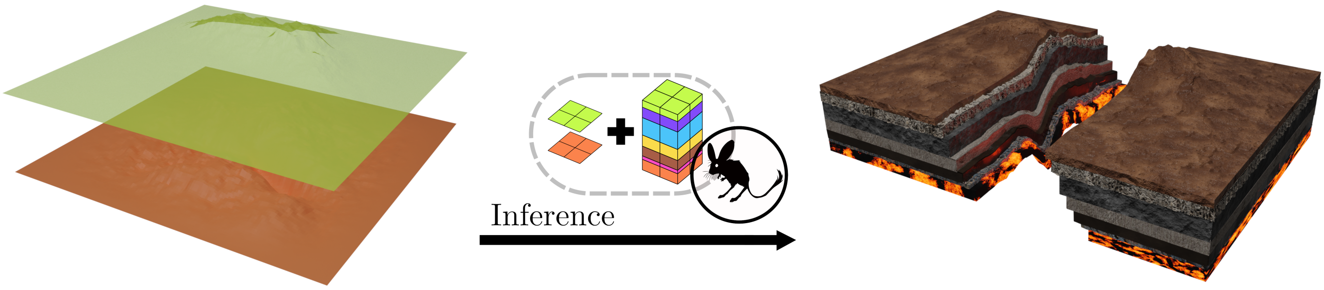

Figure 1: *Application of an inferred operation to assist procedural generation of a complex scene*

Abstract

Procedural modeling enables building complex geometric objects and scenes in a wide panel of applications. The traditional approach relies on the sequential application of a reduced set of construction rules. We offer to automatically generate new topological rules based on an initial object and the expected result of the future operation. Non-expert users can thereby develop their own operations. We exploited our approach for the modeling of the geological subsoil.

CCS Concepts

*• **Computing methodologies** → **Computer graphics; Mesh models;** Symbolic and algebraic manipulation; • **Human-centered computing** → Interaction design process and methods; • **Software and its engineering** → Automatic programming;*

1. Introduction

In geometric modeling, the development of dedicated tools often requires expertise in both computer graphics and the specific application field. This article proposes an approach to help non-computer scientists develop their operations by inferring them from a representative example and reconstructing them as rules in a procedural modeling approach. We infer the topological part of an operation (leaving aside any geometric modification). This inference is realized without prior knowledge, such that the obtained operation can then be applied in a broader context. We will illustrate our approach within the field of geology.

2. Topological structure and modeling operations

To ease the design of new operations, we rely on a homogeneous data structure called generalized maps or G-maps [DL14], considered as graphs [PLGAB22]. A node of the graph encodes part of all topological cells, i.e., a vertex, an edge, a face, and a volume simultaneously. We write modeling operations with the help of the Jerboa framework [BALGB14] where an operation is defined as a rule. The application of an operation requires finding an occurrence of the left pattern and replacing it with the right pattern. Jerboa provides scheme rules that generalize the operation via a parameter abstracting a topological cell. For instance, we can define operations that modify a vertex, a face, or a surface. Both Jerboa rules and G-maps offer a highly specialized framework that allows us to deduce topological operations. We use a reverse-engineering approach to check which variables produce a valid generalization based on the intuition that a Jerboa rule generalizes an operation via a cell.

3. Inference workflow

We now discuss which steps are automated and which ones require user interaction. Everything can be realized in a dedicated modeler freely available online that we developed with Jerboa (https://

DOI: 10.2312/egp.20221005

tinyurl.com/inferencegeology/). First, the user builds two instances of an object as a representative example of the operation before and after its application. Secondly, the user selects the parts of the object that should be taken into account to build the operation, delimiting its scope. Thirdly, the user specifies a mapping of some preserved elements. This step ensures that the deduced operation properly preserved unmodified elements within its scope. The last step is the topological folding algorithm and exploits user-specified information to deduce a compatible operation. Before running the topological folding algorithm, the user can specify a topological cell that should correspond to the rule's variable. If the topological folding algorithm provides an output, the user obtains an operation applicable in a broader context after generating the associated code via Jerboa.

4. Topological folding algorithm

The folding algorithm is intuitively built as a graph traversal finding consistent relabelling functions, i.e., finding elements corresponding to the same node or arc in the rule (see Algorithm 1 for a simplified version). More details can be found in [PBALG21]. The algorithm input consists of three elements: 1) the graph that contains both partial G-maps linked along with preserved darts, 2) the topological cell used to parameterize the operation, and 3) a dart that describes where the user would click on the object to apply the operation. When visiting a link, the function *arc_expansion* (line 8) links the dimensions that can be expended. For these dimensions, we add an arc to the graph. The other arc extremity is deduced from the input graph. When visiting a dart, the function *build_label* (line 9) considers the darts associated with a node and examines the arcs between them. If this induced subgraph is isomorphic (up to relabelling) to the initial orbit, we deduce the relabeling function and label the node accordingly. The function *add_node* (lines 3 and 10) marks the darts associated with a node as visited and adds it to the rule.

Algorithm 1: Topological folding algorithm

Input: A graph G encoding the preservation relation between two partial G-maps, an orbit type $\langle o \rangle$, and a dart a of G.
Output: A graph $\mathcal{S}$ that encodes the Jerboa rule with $\langle o \rangle$ as variable, given that the operation is applied at the dart a.

1 $Q \leftarrow \emptyset,\ \mathcal{S} \leftarrow \emptyset$ // empty queue and empty 'rule' graph
2 $h \leftarrow Node(G, \langle o \rangle, a)$ // build the hook node
3 $add\_node(\mathcal{S}, h)$ // add h to the 'rule' graph
4 $enqueue(Q, h)$
5 **while** $Q \neq \emptyset$ **do**
6 $m \leftarrow dequeue(Q)$
7 **foreach** $d \in [\![0, n]\!] \setminus label(m)$ **do**
8   $v \leftarrow arc\_expansion(G, m, d)$ // extend arcs
9   $build\_label(G, v)$ // deduce the relabeling function
10   $add\_node(\mathcal{S}, v)$
11   $enqueue(Q, v)$
12 **return** S

5. Application to geology

The modeling of the subsoil usually starts from a set of surfaces. For instance, soil horizons are surfaces parallel to the soil splitting the subsoil based on intrinsic properties. The question is then how to fill the volume between the surfaces. Indeed, several interlayers might need to be added based on stratigraphic information retrieved from core samples. In Figure 2, we present cases identified by a geologist. An application of the inferred operation (d) to a more complex scene is illustrated in the teaser (Figure 1), after the manual addition of all relevant geometrical information to the inferred rules.

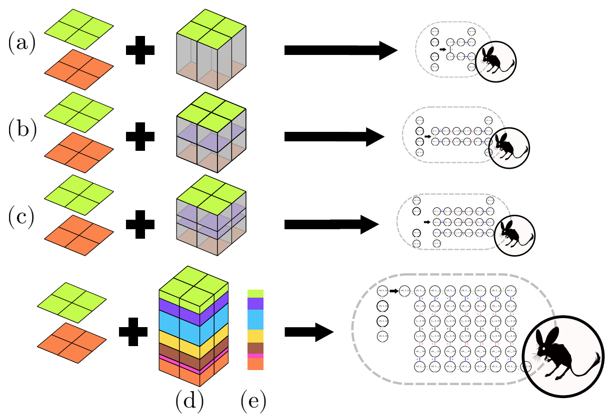

Figure 2: *Several possibilities for the layering operation: (a) no inter-layer, (b) one inter-layer, (c) two inter-layers, (d) six inter-layers based on the information described by (e), which is typically obtained by geologists from a core sample.*

6. Conclusion

We presented a method to infer topological operations from their description on a representative example. Our approach is highly facilitated by the regularity of the underlying model, namely generalized maps, and the genericity of graph-based rules in Jerboa as a solution to express modeling operations. Our algorithm essentially tries to reverse the process of Jerboa's rule instantiation by folding a graph that encodes the parts of the object modified by the operation. We used our method to reconstruct operations in geology.

References

[BALGB14] Belhaouari H., Arnould A., Le Gall P., Bellet T.: Jerboa: A Graph Transformation Library for Topology-Based Geometric Modeling. In *Graph Transformation (ICGT 2014)* (Cham, 2014), Giese H., König B., (Eds.), vol. 8571 of *Lecture Notes in Computer Science*, Springer International Publishing, pp. 269–284. doi:10.1007/978-3-319-09108-2_18. 1

[DL14] Damiand G., Lienhardt P.: *Combinatorial Maps: Efficient Data Structures for Computer Graphics and Image Processing*. CRC Press, 2014. 1

[PBALG21] Pascual R., Belhaouari H., Arnould A., Le Gall P.: Inferring topological operations on G-maps formalism: application to iterated function systems. working paper or preprint, 2021. URL: https://hal.archives-ouvertes.fr/hal-03491856. 2

[PLGAB22] Pascual R., Le Gall P., Arnould A., Belhaouari H.: Topological consistency preservation with graph transformation schemes. *Science of Computer Programming 214* (2022), 102728. doi:https://doi.org/10.1016/j.scico.2021.102728. 1