

Some Thoughts on a Datatype
for Higher Genus Graphs

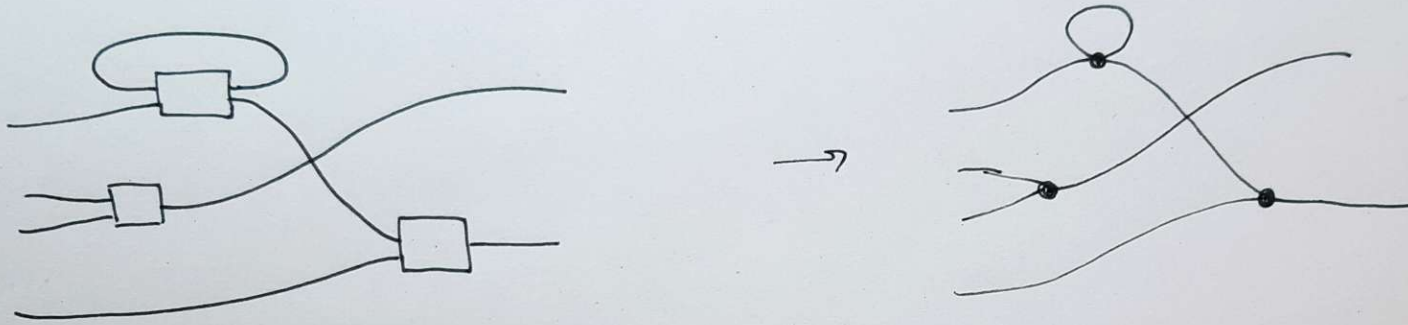
Malin Altenmüller

MSP 101

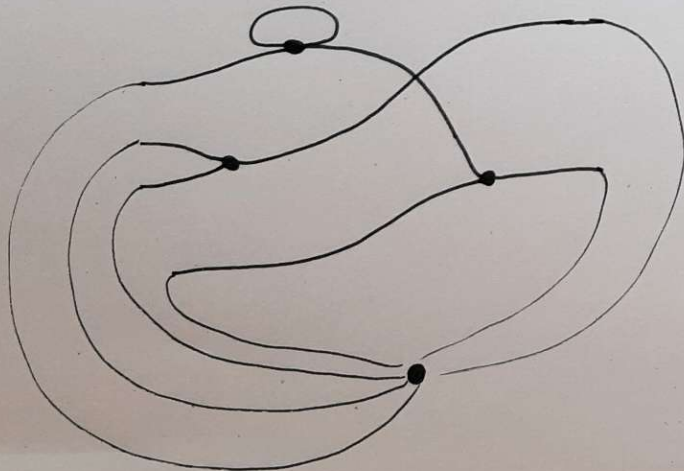
25/02/2021

Graphs model circuits

> wires become edges, boxes become vertices



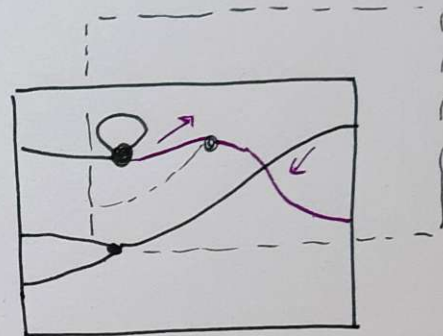
> open graphs for inputs & outputs



Graphs model circuits

Consider the topology of a graph, e.g.:

- > no wires crossing
- > explicit (and non-trivial) crossings



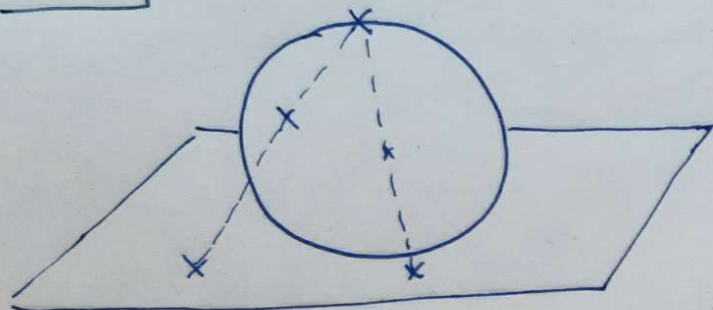
The bigger picture: How to program with graphs?

- > Datatype for Graphs?
- > When are two graphs the same?
- > applying rewriting

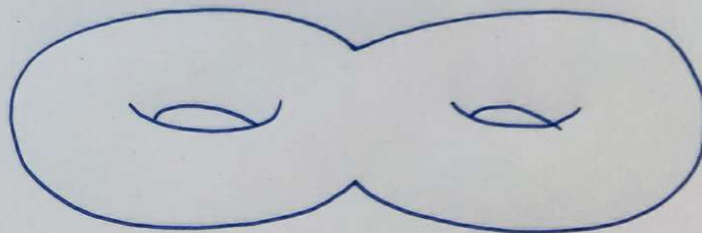
Surfaces

- > closed surface: compact topological space without boundary
- > orientable: consistent normal vector

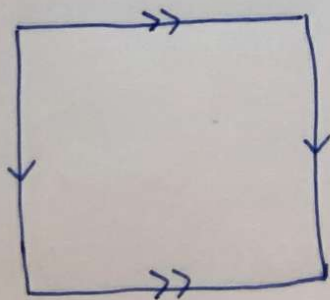
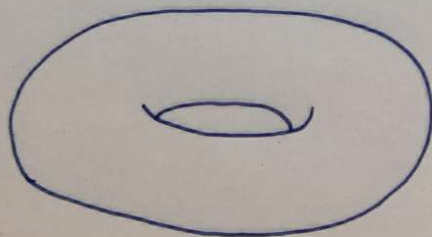
genus 0



genus 2



genus 1

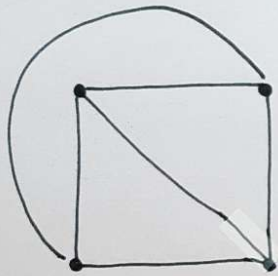


more holes
→ higher genus



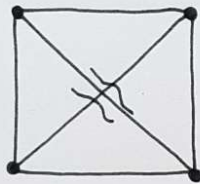
Graphs & Maps

- > here : graphs are closed, multiple edges & self-loops allowed, connected
- > map = drawing of a graph onto some surface

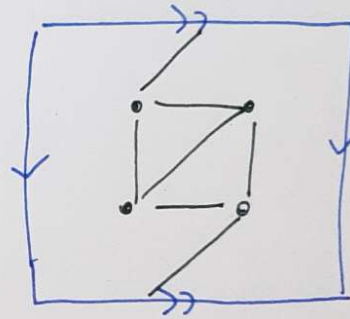


genus 0

same graph



genus 1



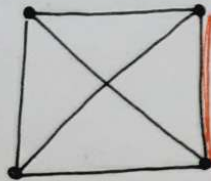
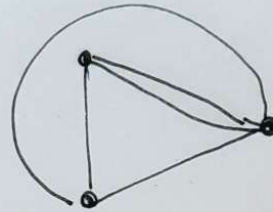
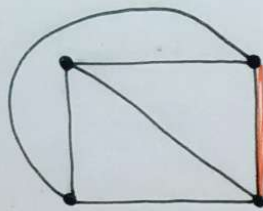
- > represent map by rotations



from now on
all graphs are
maps

Edge contraction

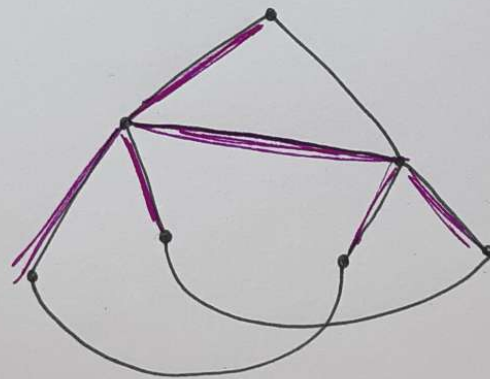
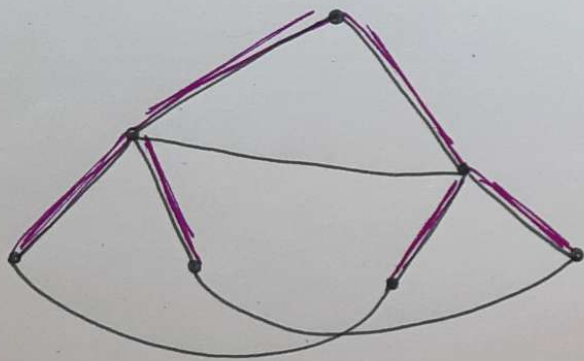
> Contracting a plane edge doesn't change the genus



> doing this repeatedly : contract plane subgraph

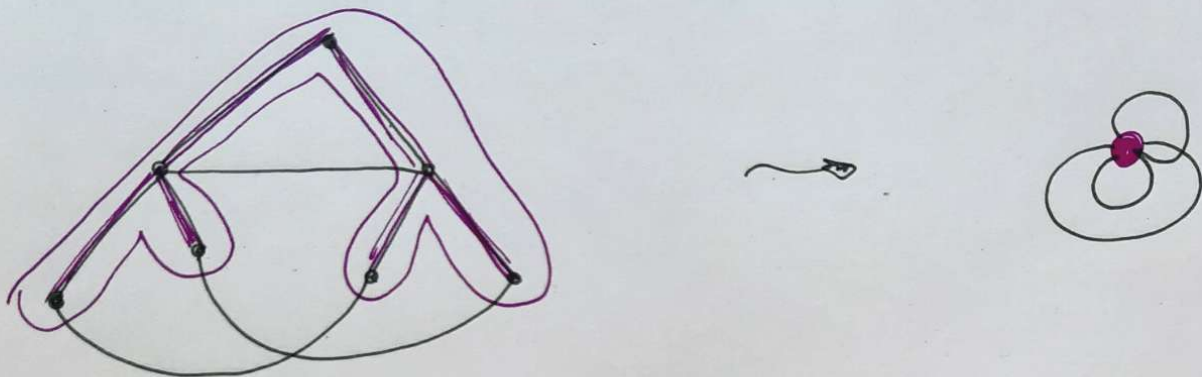
Spanning Trees

- > can find spanning trees in any graph
- > graph = spanning tree + cross edges

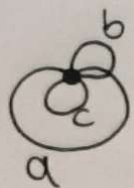


On cross-edges

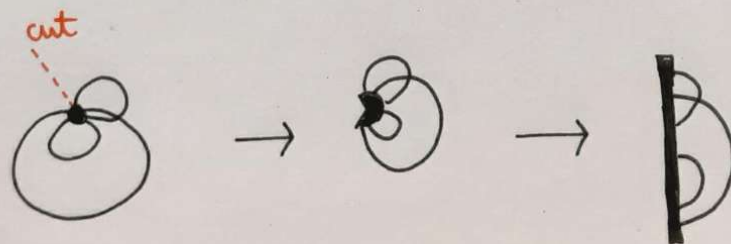
> contracting a spanning tree



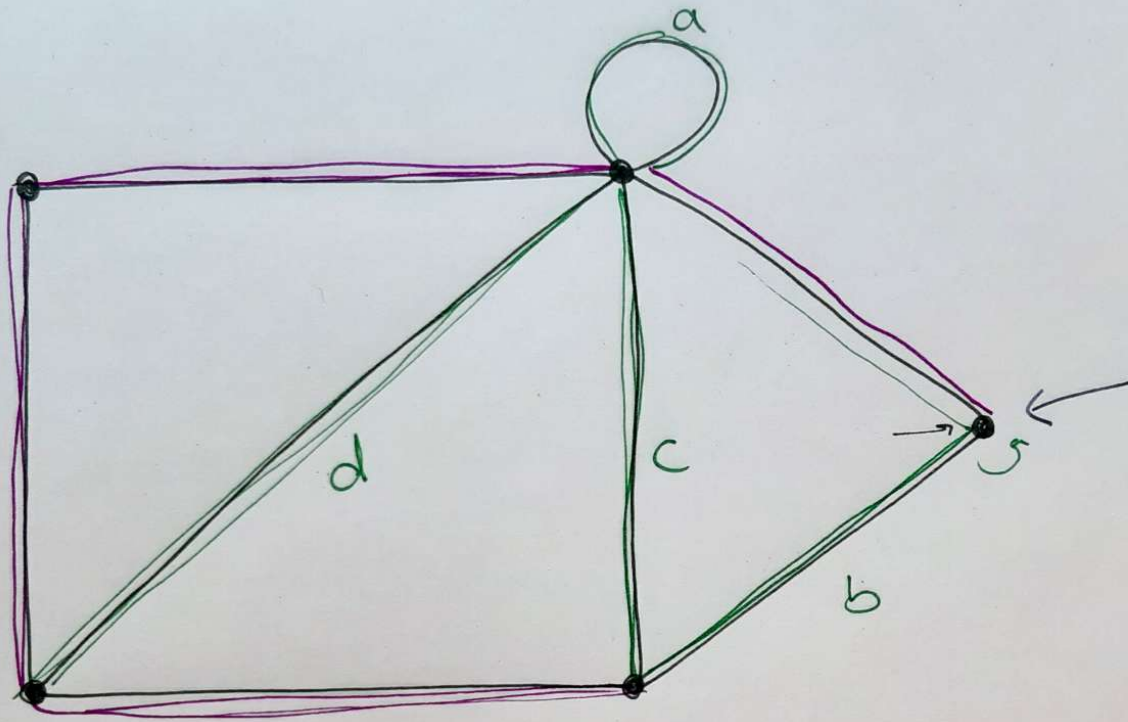
> word representation



accbab



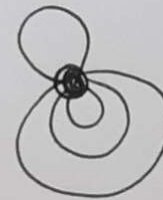
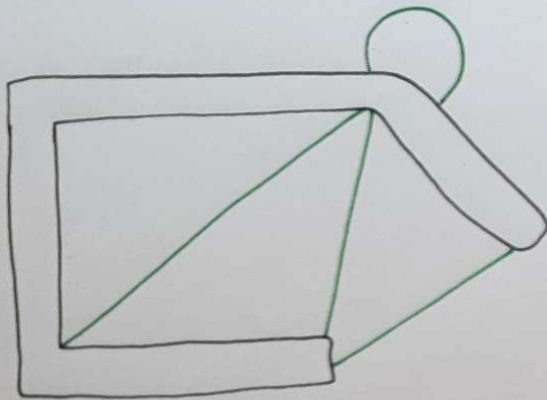
The plane case



~~graph~~

The plane case

The word corresponding to the cross edges:
element of a context-free grammar



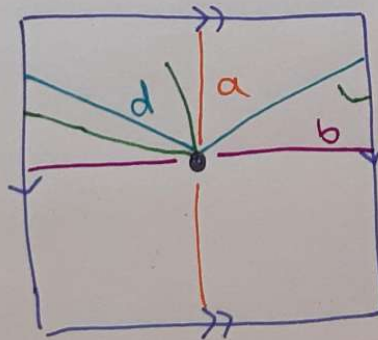
aabcdc

Now: How to go higher genus?

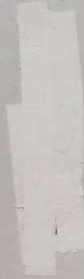
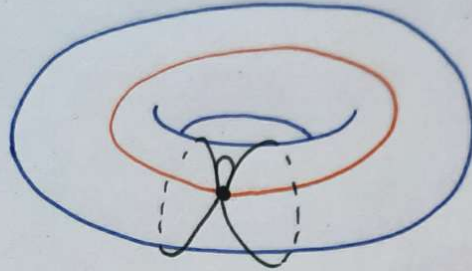
1) Multiple Stacks

- > one stack in the plane case, how about two for toroidal graphs?
- > multi-stack pushdown automata [1]
 - subclass of context-sensitive languages
 - for studying of nested words, recursive sequential programs

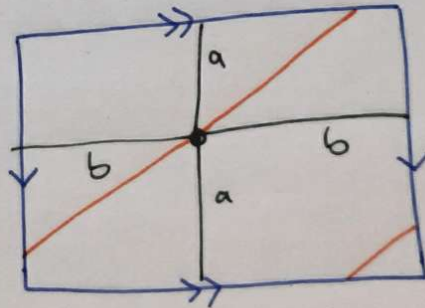
abcdcadb



2) One edge at a time

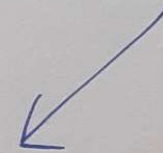
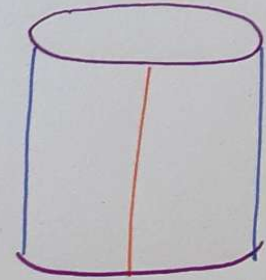
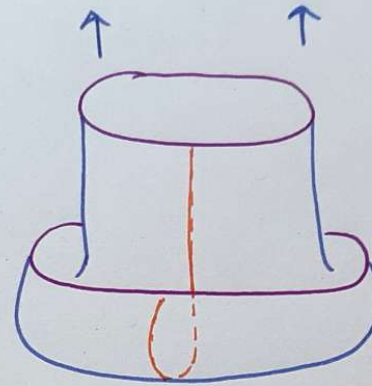
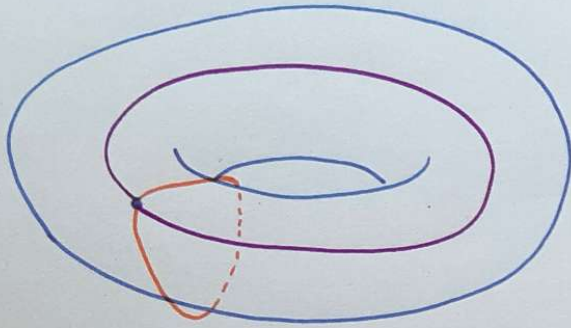


U U U U U



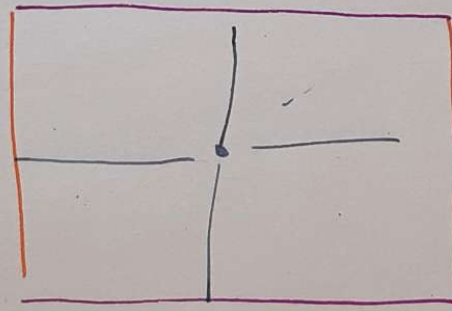
3) Cutting higher genus surfaces

12a

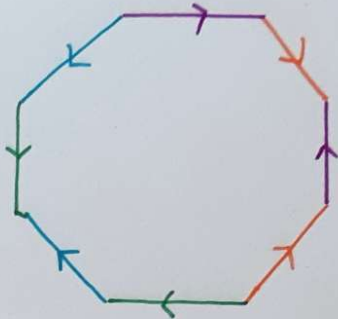
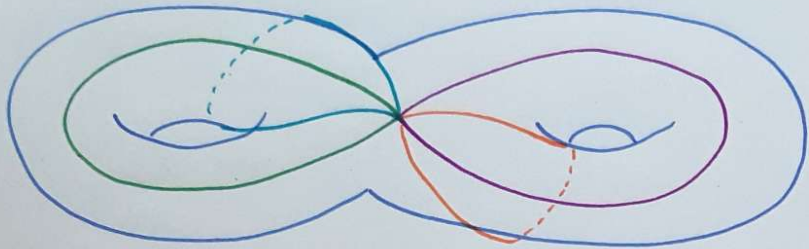


torus

abcabc

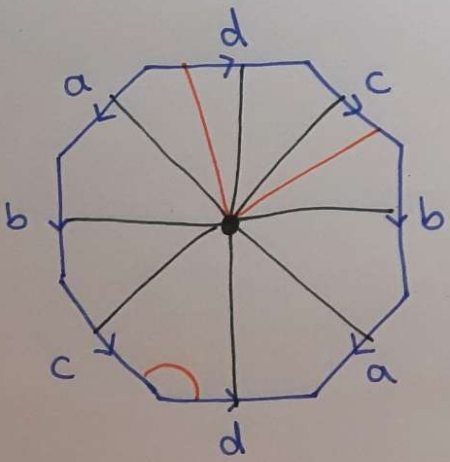


double torus



ababcdcd

different way of cutting



abcdeabcde

abcdabcd

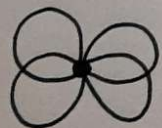
4) Forbidden minors [2]

> minor relation: genus preserving edge contraction

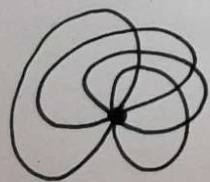
> forbidden minor for the sphere:



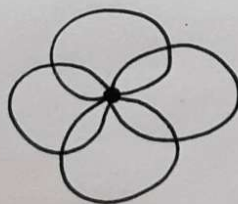
> forbidden minors for the torus:



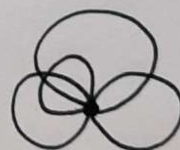
ababcdcd



abcdabcd



abcadcbd



abacdbcd

131 for the double-t.
14118 for the 3 - t.
⋮

Some open questions

- > Topological information in multi-stack approach?
- > How to go from forbidden minors towards the graph type?
- > Known problem to topologists?
- > Any similar issues in different/other areas?

Thank you for listening!

Some references

[1] La Torre / Madhusudan / Parlato :

A Robust Class of Context-Sensitive Languages (LICS '07)

[2] Courcelle / Dussaux :

Map genus, forbidden maps, and monadic second order logic
(Electronic Journal of Combinatorics '02)

Diestel : Graph Theory (Springer Graduate Texts in mathematics)

↑
or your
favourite graph theory
book