On linear layouts with SAT

Mirco Haug

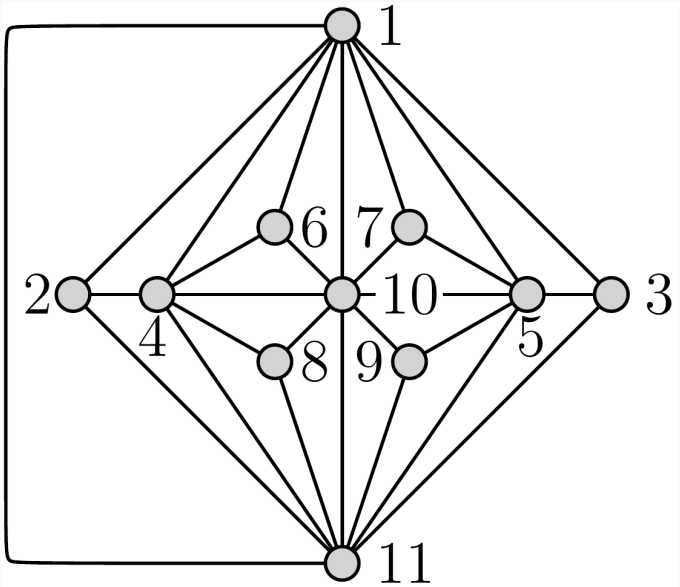
July 10, 2019

Eberhard Karls Universität Tübingen

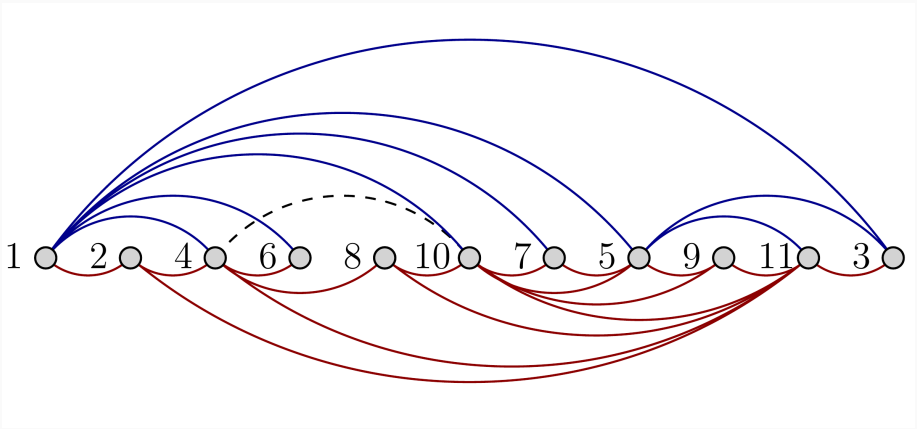
Contents

Motivation

What to do?

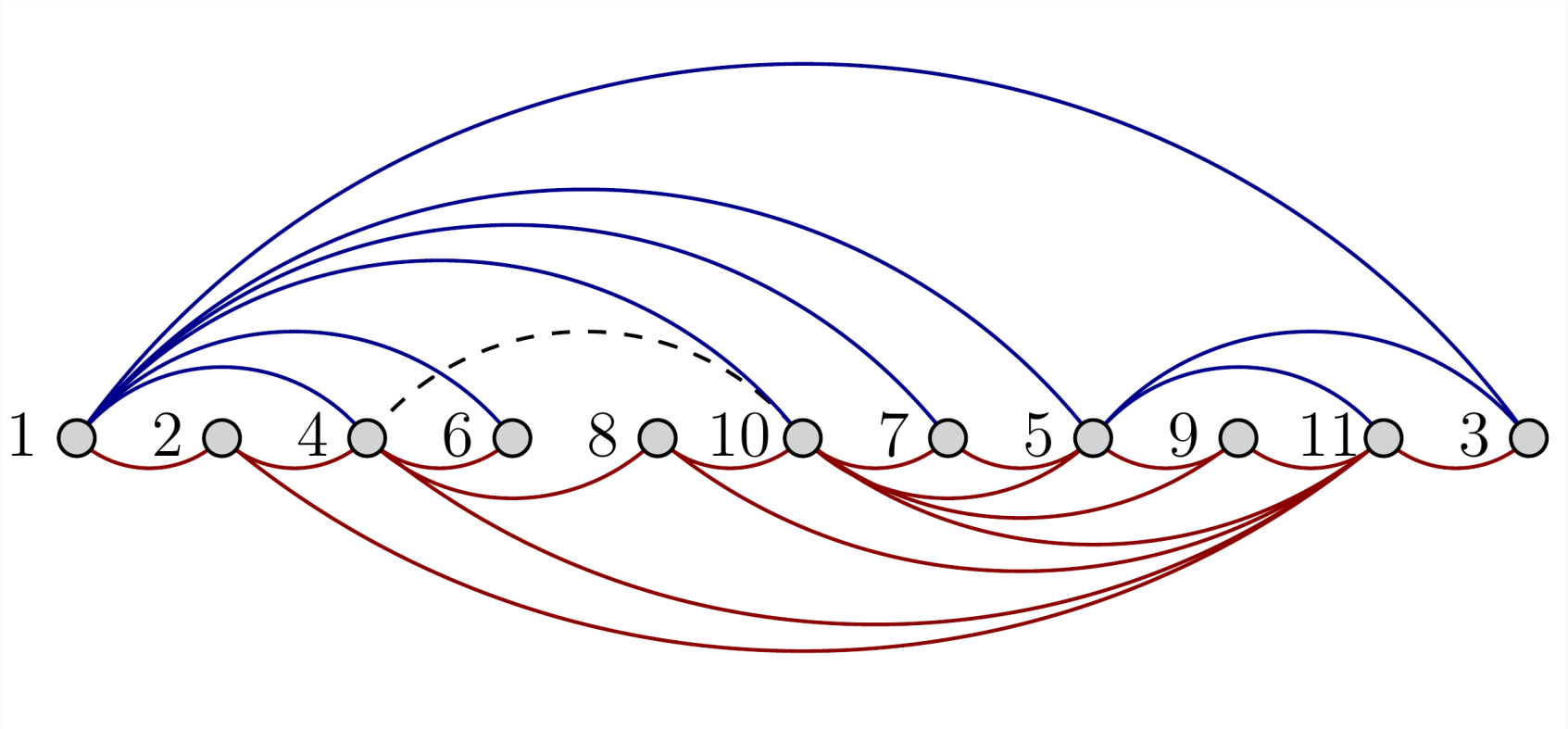


$\Rightarrow$



The bookembedding problem

- No crossings
- Multiple pages / Colors
- Node order
- Edge assignments


1
2
4
6
8
10
7
5
9
11
3

Approach

- Combinatoric explosion
- SAT solvers
- http://be.cs.arizona.edu
- Rather static
- Flexibility needed
- Old proof sketch of Yannakakis[?]

Theory

Standart book embedding[?]

- Node order $\sigma(n_1, n_2)$
- Edge assignment $\phi_{P1}(e_1)$

Constraints

- EDGES_ON_PAGES
- EDGES_SAME_PAGES
- EDGES_DIFFERENT_PAGES
- EDGES_TO_SUB_ARC_ON_PAGES
- EDGES_FROM_NODES_ON_PAGES
- NODES_PREDECESSOR
- NODES_REQUIRE_ABSOLUTE_ORDER
- NODES_REQUIRE_PARTIAL_ORDER
- NODES_FORBID_PARTIAL_ORDER
- NODES_CONSECUTIVE

Constraints

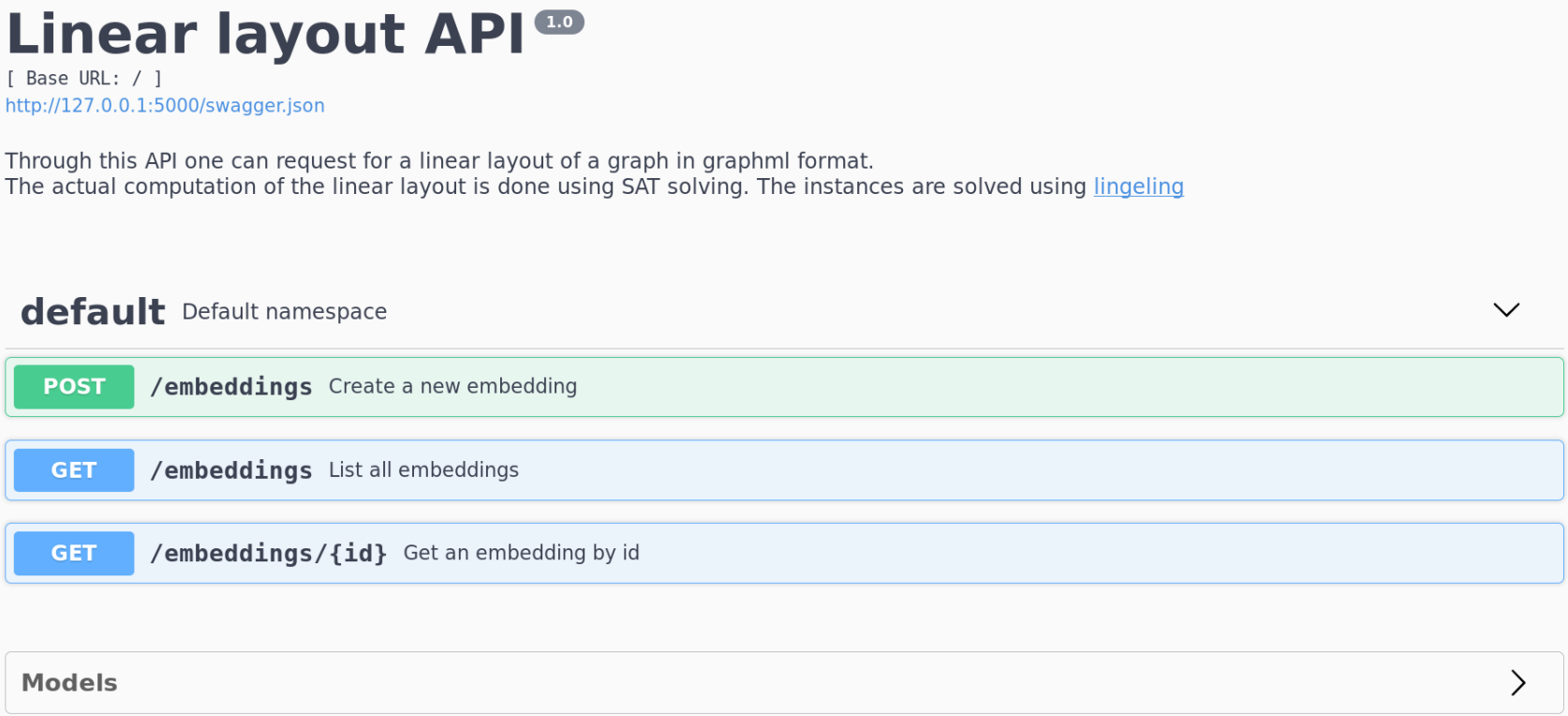
- EDGES_ON_PAGES
- EDGES_SAME_PAGES
- EDGES_DIFFERENT_PAGES
- EDGES_TO_SUB_ARC_ON_PAGES
- EDGES_FROM_NODES_ON_PAGES
- NODES_PREDECESSOR
- NODES_REQUIRE_ABSOLUTE_ORDER
- NODES_REQUIRE_PARTIAL_ORDER
- NODES_FORBID_PARTIAL_ORDER
- NODES_CONSECUTIVE

Implementation

Main classes

- app.py
- model.py
- solver.py

API ⇒ Swagger UI



app.py

- Schema definition
- Deserialize
- Validate

model.py

- Clause generation
- DIMACS Generation
- Parser for lingelinge result

solver.py

- Glue
- Calls lingeling
- Service interface

Demo

Demo

http://algo.inf.uni-tuebingen.de/linearlayouts/index.html#or174

Bibliography