Framework for profiling Critical Path related Algorithms

RP#66

Henri Trenquier

Supervisors:
Dr. Zhiming Zhao & Dr. Arie Taal
MSc Systems and Network Engineering

UNIVERSITY OF AMSTERDAM

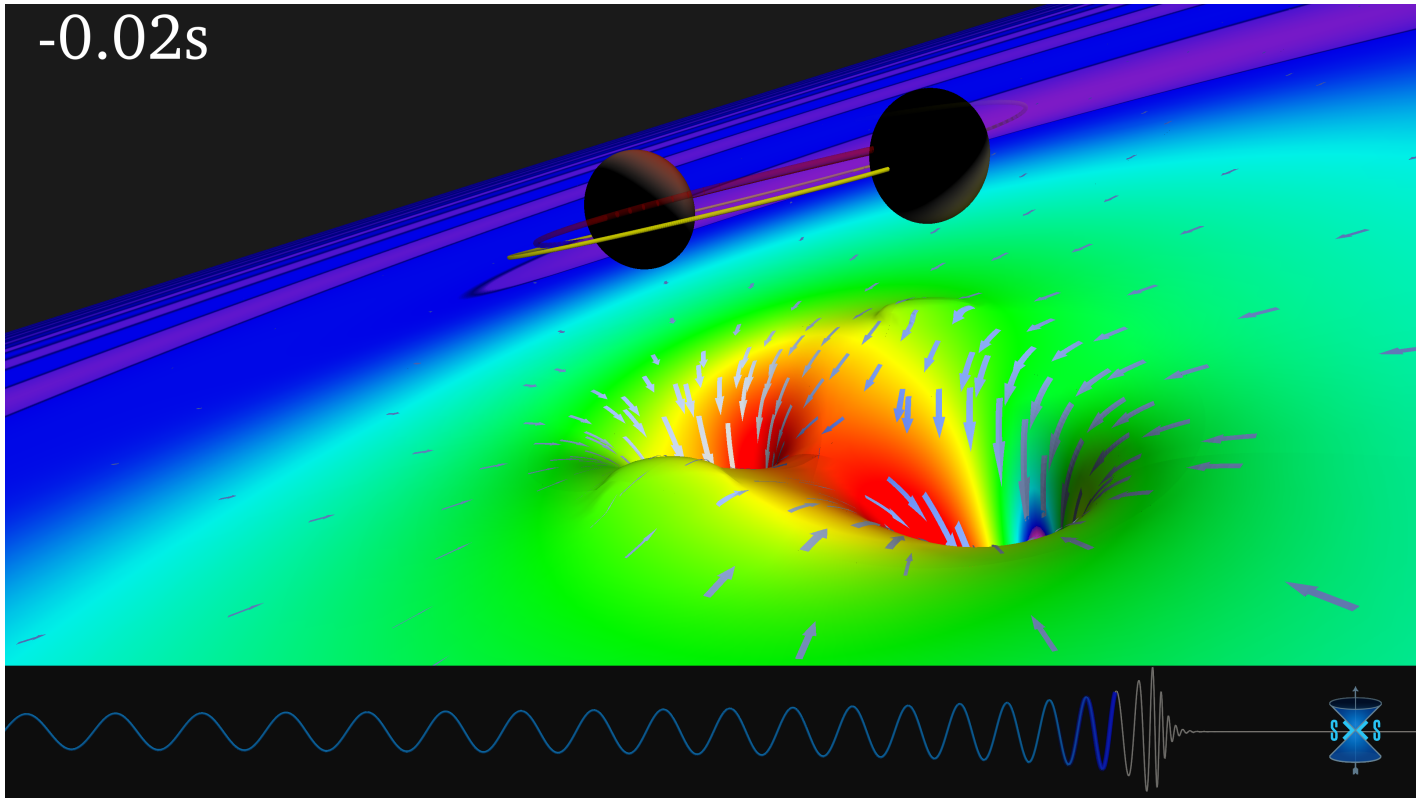
February 6, 2018

System and Network Engineering

Context

LIGO Gravitational wave detection analysis

- Data to be processed
  - Deadline
  - Order
- Time critical problem

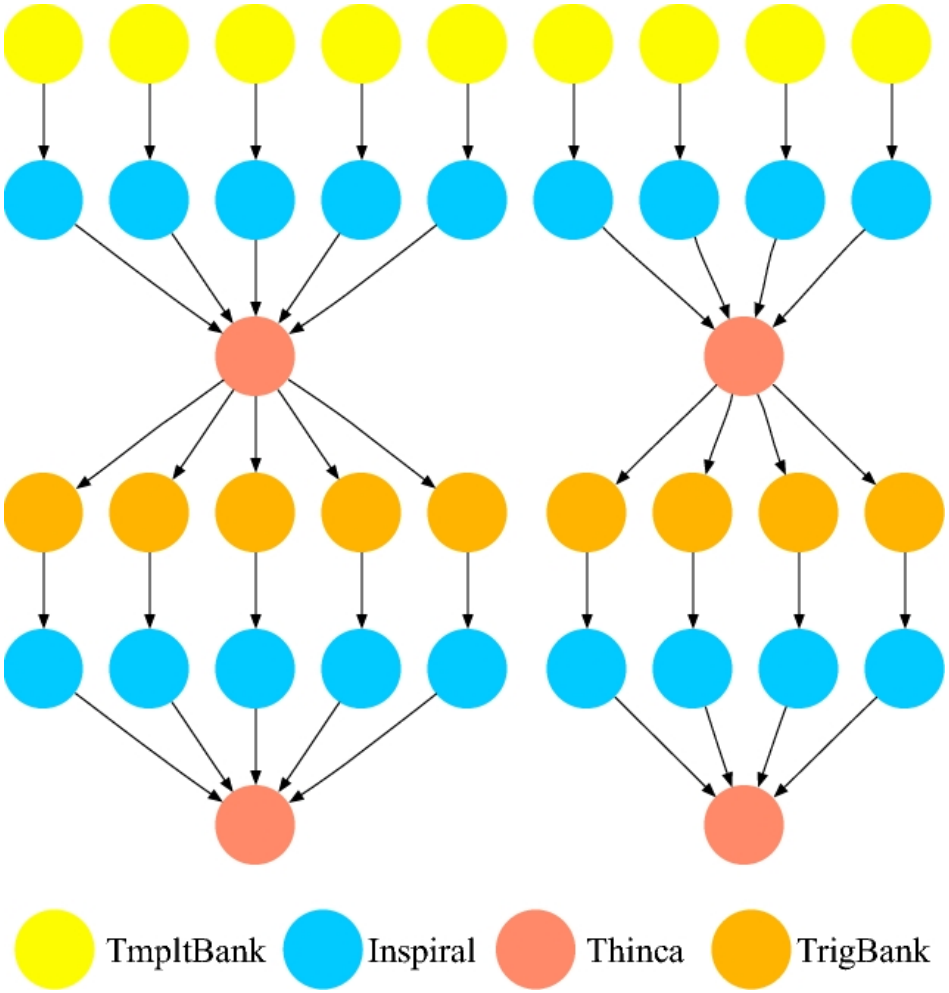


System and Network Engineering

Context

LIGO Gravitational wave detection analysis

- Defines a task graph



System and Network Engineering

Context

Critical Path related Algorithms

- "Deadline-constrained workflow scheduling algorithms for Infrastructure as a Service Clouds" (2013)
  Saeid Abrishami and Mahmoud Naghibzadeh and Dick H.J. Epema
  Cited **311** times.
- Critical path related algorithms i.e. **IC-PCP**

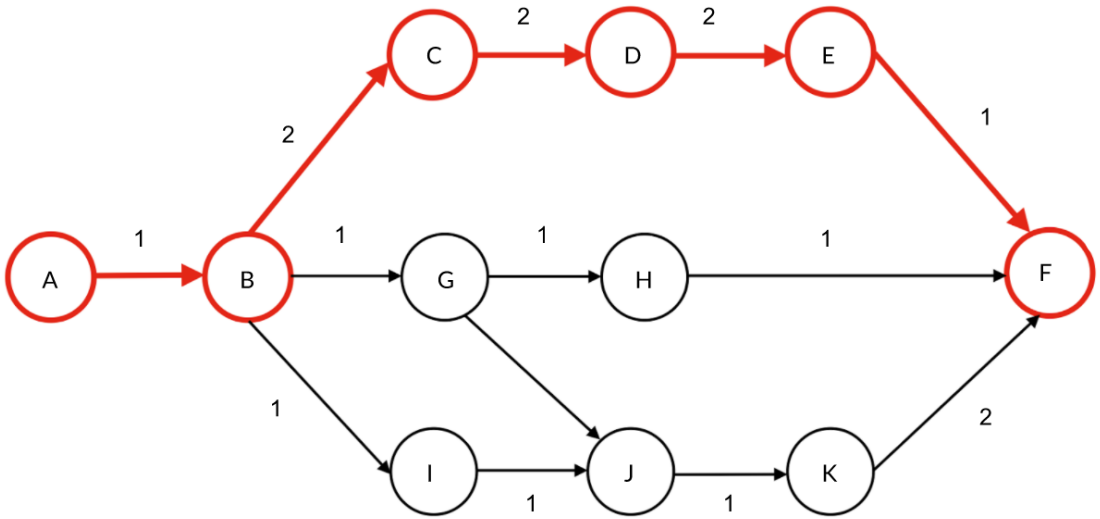


Figure: Critical Path on a Directed Acyclic Graph

System and Network Engineering

Complexity of the problem

Multi-parameter problem:

Complexity of the problem

Multi-parameter problem:

- Time constraints

System and Network Engineering

Complexity of the problem

Multi-parameter problem:

- Time constraints
- Performance model

System and Network Engineering

Complexity of the problem

Multi-parameter problem:

- Time constraints
- Performance model
- Cost constraints

Complexity of the problem

Multi-parameter problem:

- Time constraints
- Performance model
- Cost constraints
- Unlimited task graphs

System and Network Engineering

Research Question

Can we profile critical path related algorithms by the means of a framework ?

- *How relevant is a framework for profiling critical path related algorithms ?*
- *What are the challenges for such a framework ?*

System and Network Engineering

Motivation

Why a framework ?

- Compare algorithms performances
- Compare implementations of the same algorithm
- Manipulate and plot output data

What is the best implementation for the "Inspiral" (LIGO) task graph ?

System and Network Engineering

Prototype Architecture

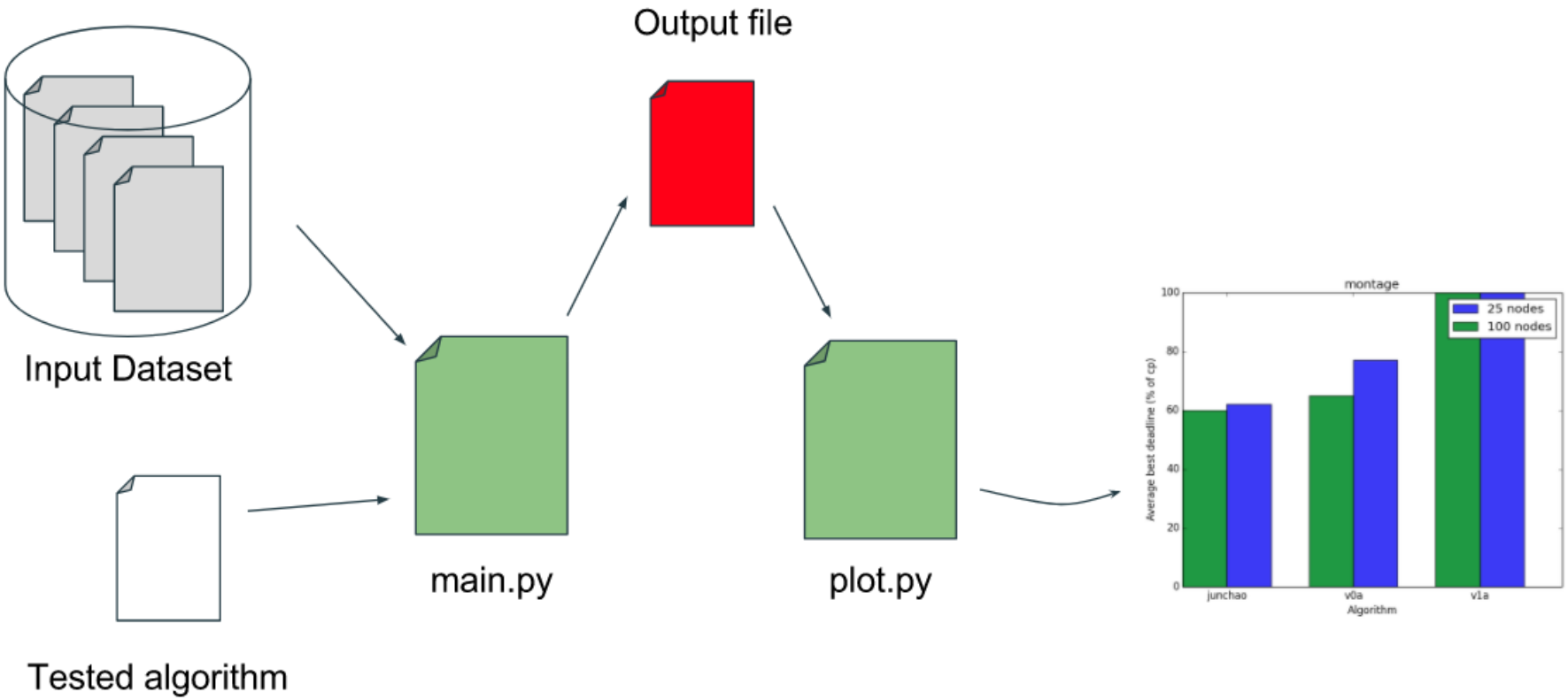


Figure: Prototype Architecture

System and Network Engineering

Prototype Inner Working

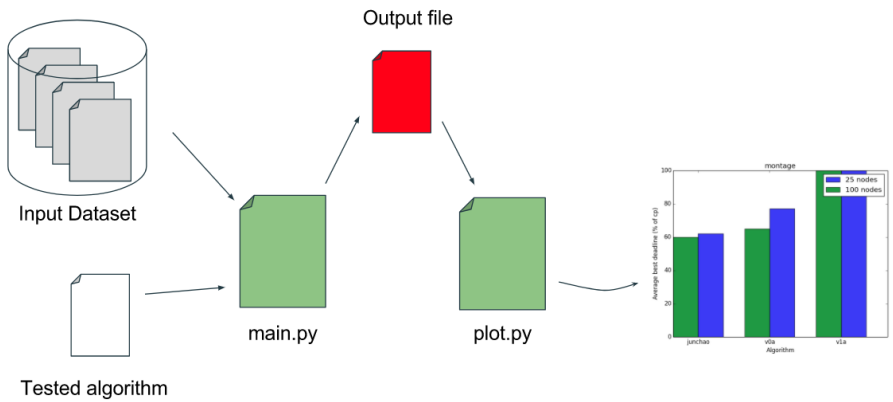


Figure: Prototype Architecture

1. Step 1: *main.py*
   - Browse input task-graph dataset
   - Run algorithm on dataset with specific parameters
   - Write the results in an output file
2. Step 2: *plot.py*
   - Browse output file lines
   - Structure the data
   - Plot the structured data

System and Network Engineering

File Formats

- Input task graph format
  - Topology

```
digraph dag {
    0 -> 3  [weight=0.0];
    0 -> 48 [weight=0.0];
    1 -> 101    [weight=5.0];
    2 -> 101    [weight=5.0];
    3 -> 4  [weight=6.0];
    3 -> 6  [weight=9.0];
```

  - Performance Model

```
0,12,9,8,7,5,11,12,19,12,15,15,13,8,11,12,17,10,16,7,7,16,15,18,17,14
0,24,18,16,14,10,22,24,38,24,30,30,26,16,22,24,34,20,32,14,14,32,30,36,34,28
0,36,27,24,21,15,33,36,57,36,45,45,39,24,33,36,51,30,48,21,21,48,45,54,51,42
```

  - VM prices

```
5,2,1
```

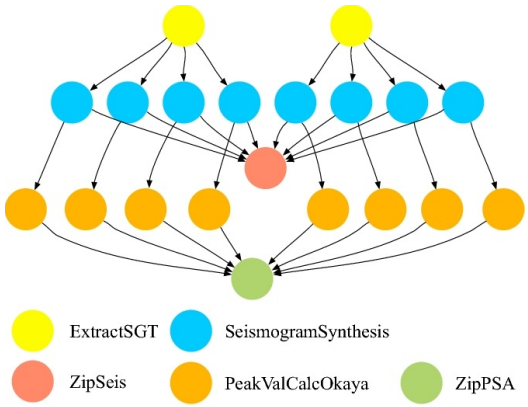
- Output file format

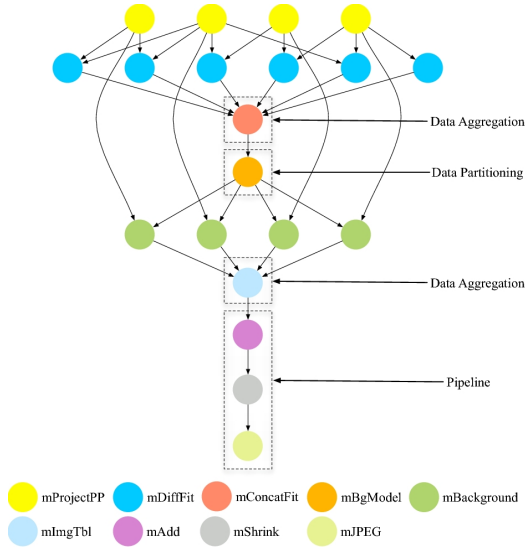
```
1.30.5;1;0;[22, 23, 24, 27, 29, 30];0.0302591323853
1.30.5;2;0;[22, 23, 24, 27, 29, 30];0.029815196991
1.30.5;3;0;[22, 23, 24, 27, 29, 30];0.0295469760895
1.30.5;4;0;[22, 23, 24, 27, 29, 30];0.0297148227692
1.30.5;5;0;[22, 23, 24, 27, 29, 30];0.0296459197998
1.30.5;6;0;[22, 23, 24, 27, 29, 30];0.0298428535461
1.30.5;7;0;[22, 23, 24, 27, 29, 30];0.0298299789429
```

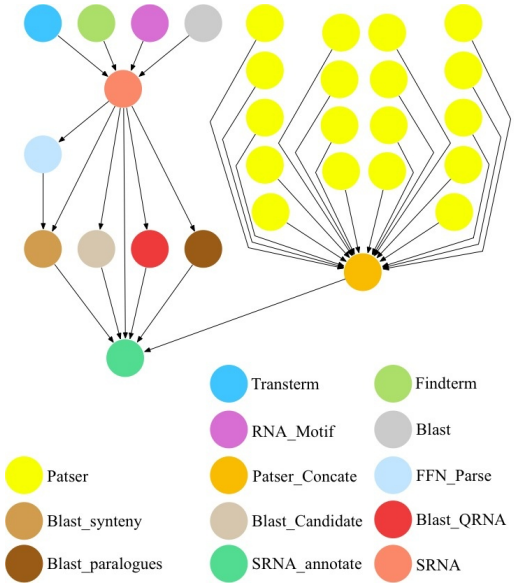
System and Network Engineering

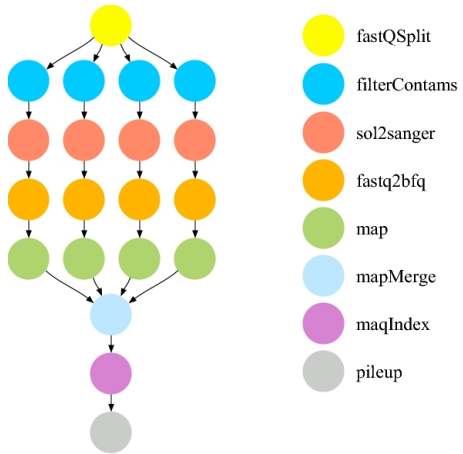
Input Data sets

Other topologies : Cybershake, Montage, Sipht, Epigenomics





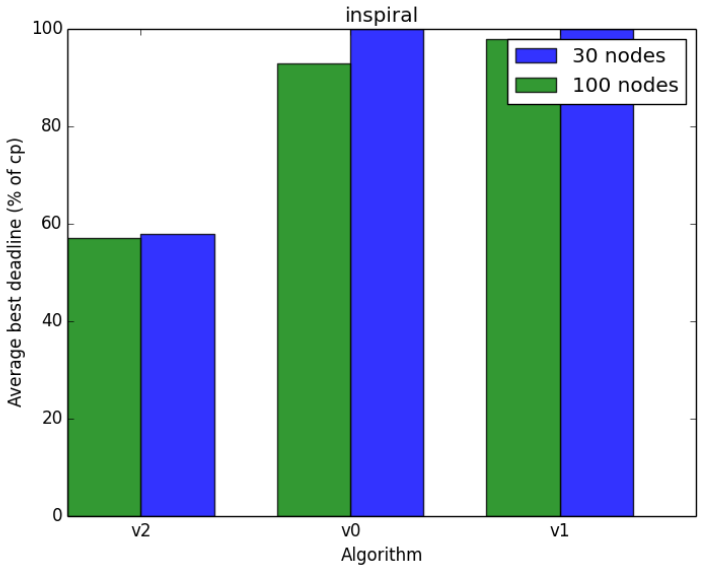


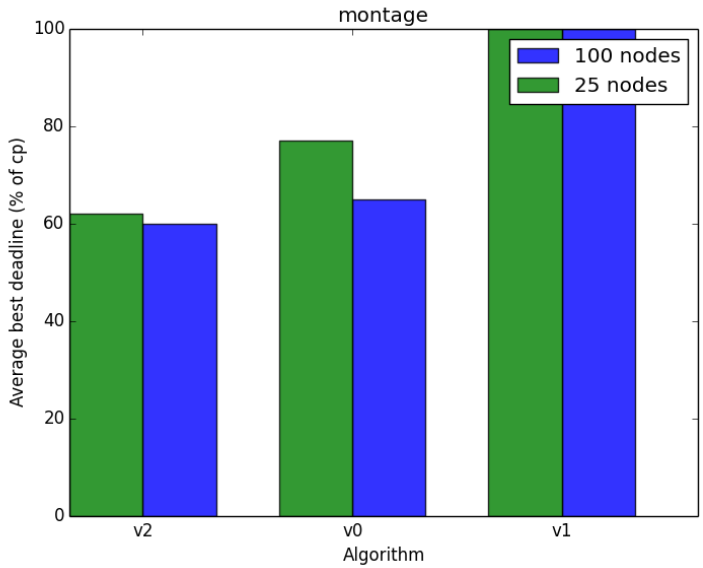


System and Network Engineering

Experiment #1

- Assumption : "If the algorithm succeeds for a shorter deadline, then it will succeed for a longer one"
- Input datasets: Montage & Inspiral
- KPI : shortest deadline

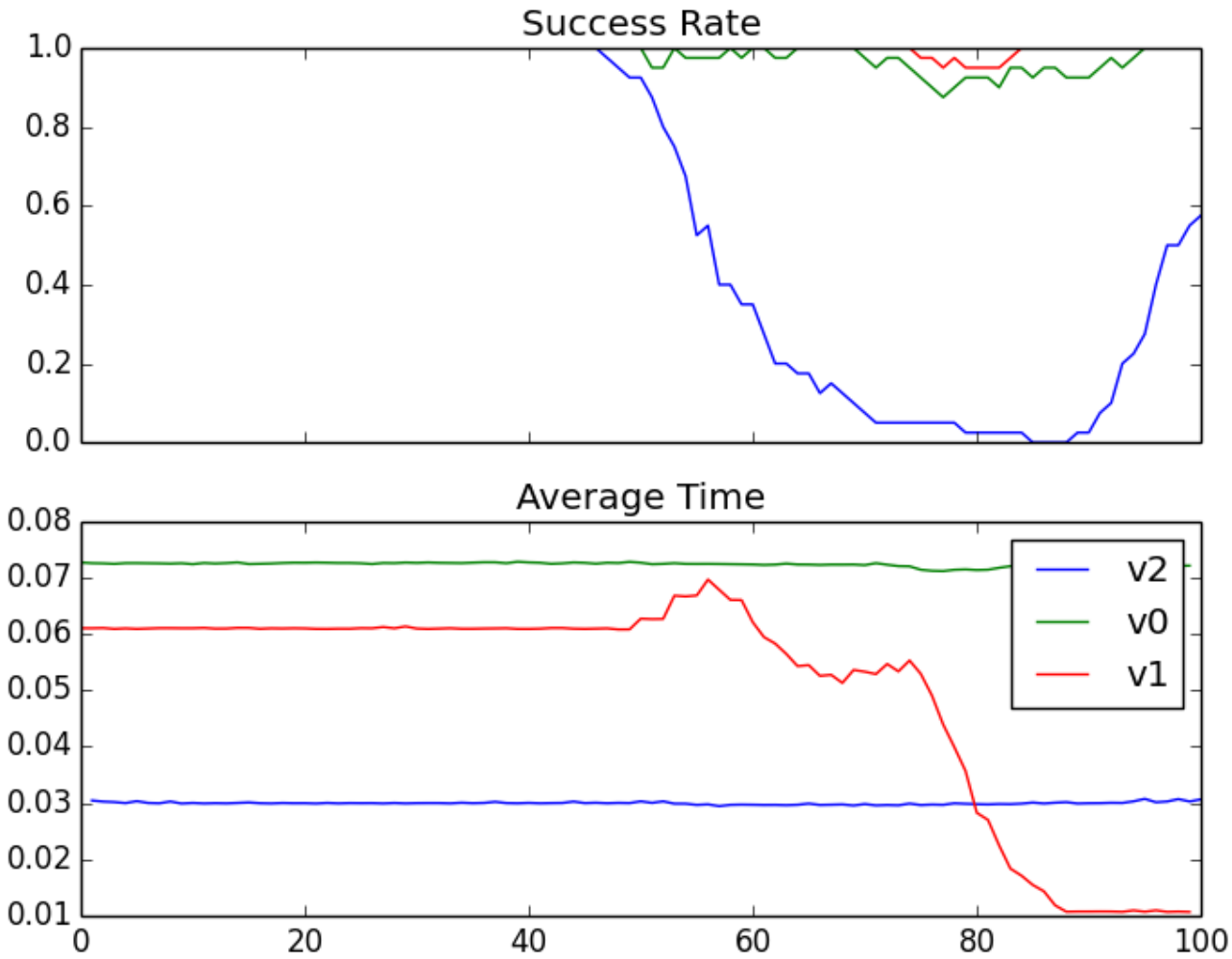




System and Network Engineering

Experiment #2

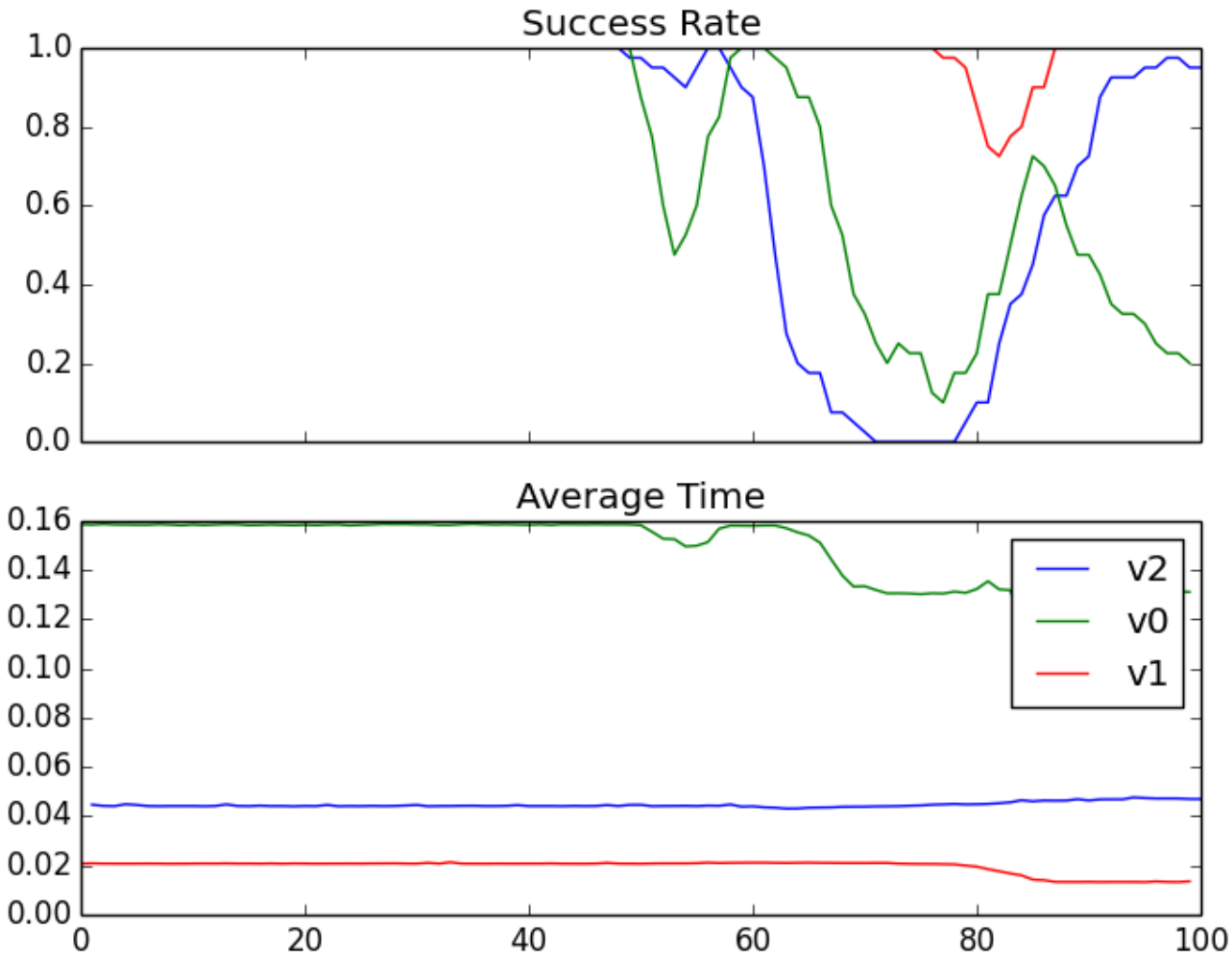
- Input datasets: **Inspiral**
- KPI : Time and Success rate



System and Network Engineering

Experiment #2

- Input datasets: **Montage**
- Experiment #1 assumption : False



System and Network Engineering

Discussion

- Performance Model : not changeable
- Algorithms' codes have to be adapted
- Solution check is done by the algorithm (not the framework)
- Framework needs a lot of resources
- Output format : Cost ?

System and Network Engineering

Conclusion

Can we profile critical path related algorithms by the means of a framework ?

- Meaningful results
- Assumption proved wrong
- Empirical solution

System and Network Engineering

More motivations

But also...

- Store algorithms and results
- Authoritative source

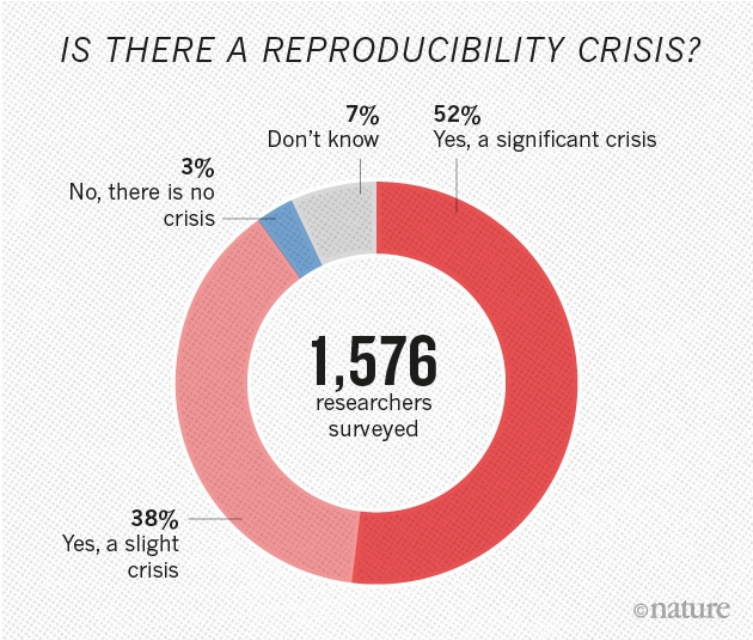


Figure: https://www.nature.com/news/1-500-scientists-lift-the-lid-on-reproducibility-1.19970

System and Network Engineering

Future Work

- Universal standards (challenge)
- Web-application
- Community databases
- New features:

Feature	Note
DAG Generator	Create and visualize new topologies
DAG Parsing	Manage different input topologies format
input checker	Check input algorithm parameters
Solution checker	Make sure the final solution is valid

Table: Framework features

System and Network Engineering