

AVMf

**An Open-Source
Implementation of the
Alternating Variable Method**

**Gregory M. Kapfhammer
Phil McMinn**

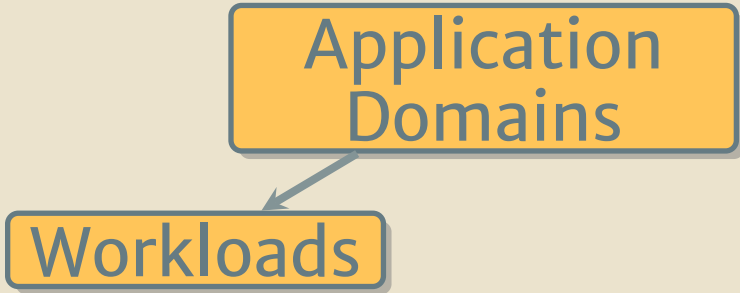
SSBSE 2016

October 9, 2016

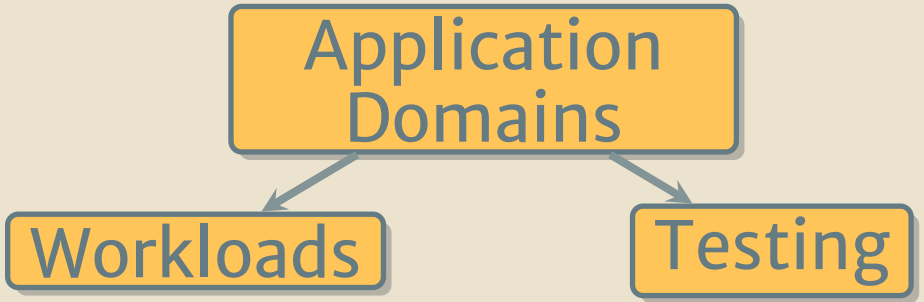
AVM is Everywhere

Application
Domains

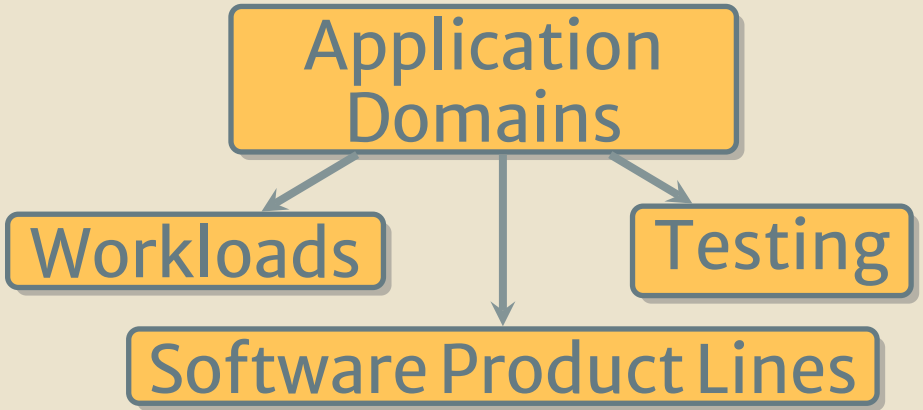
AVM is Everywhere



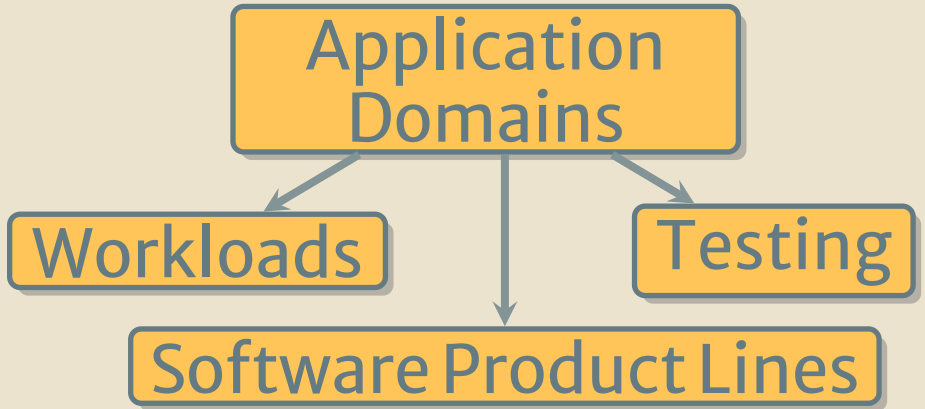
AVM is Everywhere



AVM is Everywhere



AVM is Everywhere



AVM is used in varied domains

Exploring AVM

Input Vector

Exploring AVM

Input Vector

$$\vec{X} = (x_1, x_2, \dots, x_n)$$

Exploring AVM

Input Vector

$$\vec{X} = (x_1, x_2, \dots, x_n)$$

Exploring AVM

Input Vector

$$\vec{x} = (x_1, x_2, \dots, x_n)$$

Objective Function

Exploring AVM

Input Vector

$$\vec{X} = (x_1, x_2, \dots, x_n)$$

Exploring AVM

Input Vector

$$\vec{X} = (x_1, x_2, \dots, x_n)$$

Exploratory Moves

Exploring AVM

Input Vector

$$\vec{X} = (x_1, x_2, \dots, x_n)$$

Exploratory Moves

Positive or negative “direction”?

Exploring AVM

Input Vector

$$\vec{X} = (x_1, x_2, \dots, x_n)$$

Pattern Moves

Exploring AVM

Input Vector

$$\vec{X} = (x_1, x_2, \dots, x_n)$$

Pattern Moves

Improve objective value?

Exploring AVM

Input Vector

$$\vec{X} = (x_1, x_2, \dots, x_n)$$

Pattern Moves

Improve objective value?

Yes! pattern or No! explore

Exploring AVM

Input Vector

$$\vec{X} = (x_1, x_2, \dots, x_n)$$

Pattern Moves

Improve objective value?

Yes! pattern or No! explore

Exploring AVM

Input Vector

$$\vec{X} = (x_1, x_2, \dots, x_n)$$

Exploring AVM

Input Vector

$$\vec{X} = (x_1, x_2, \dots, x_n)$$

Exploratory and
Pattern Moves

Exploring AVM

Input Vector

$$\vec{X} = (x_1, x_2, \dots, x_n)$$

Exploratory and
Pattern Moves

Consider all input vector variables

Exploring AVM

Input Vector

$$\vec{X} = (x_1, x_2, \dots, x_n)$$

Exploratory and
Pattern Moves

Consider all input vector variables

Exploring AVM

Input Vector

$$\vec{X} = (x_1, x_2, \dots, x_n)$$

Exploratory and
Pattern Moves

Revisit each x_i in the input vector

Exploring AVM

Input Vector

$$\vec{X} = (x_1, x_2, \dots, x_n)$$

Exploratory and
Pattern Moves

Restart for local optimum

Exploring AVM

Input Vector

$$\vec{X} = (x_1, x_2, \dots, x_n)$$

Exploratory and
Pattern Moves

Continue until termination condition

AVM Innovations

Search
Algorithms

AVM Innovations

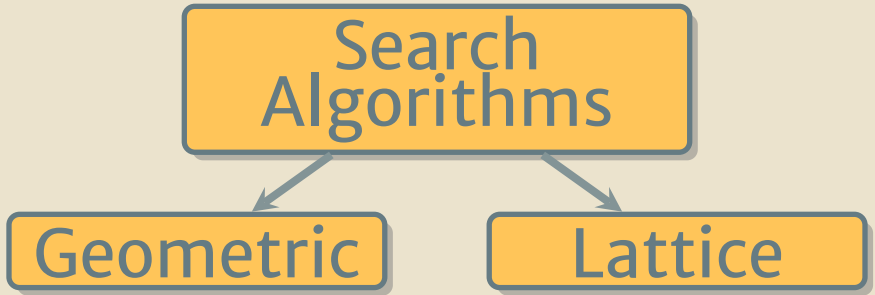
Search
Algorithms

```
graph TD; A[Search Algorithms] --> B[Geometric]
```

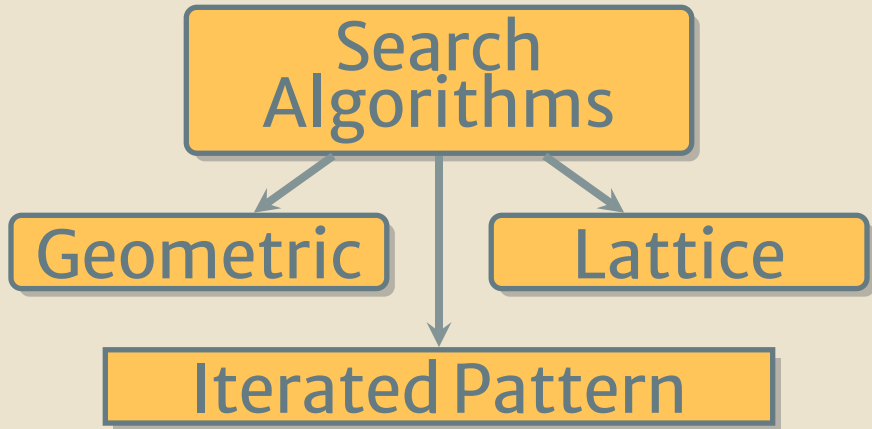
The diagram consists of two yellow rounded rectangular boxes with dark blue borders. The top box contains the text 'Search Algorithms' in a dark blue font. A grey arrow points from the bottom-left corner of this box to the top-left corner of a second, smaller yellow rounded rectangular box below it, which contains the text 'Geometric' in a dark blue font.

Geometric

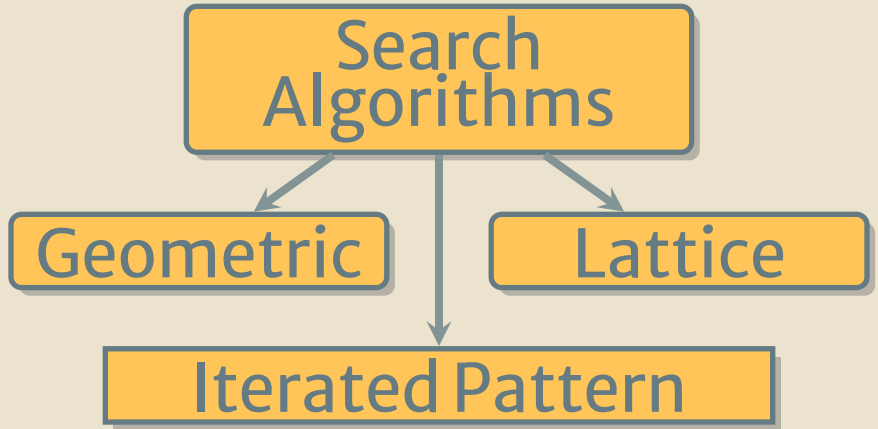
AVM Innovations



AVM Innovations

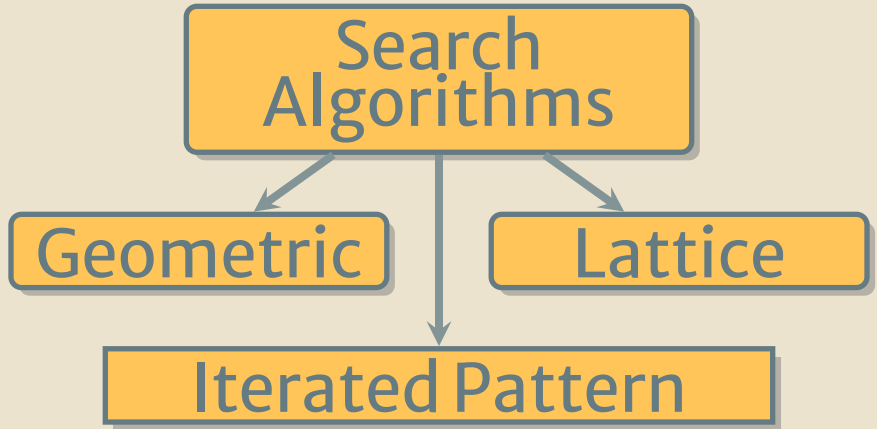


AVM Innovations



Better search for many landscapes

AVM Innovations



Provably faster for unimodal

AVM Innovations

Representations

AVM Innovations

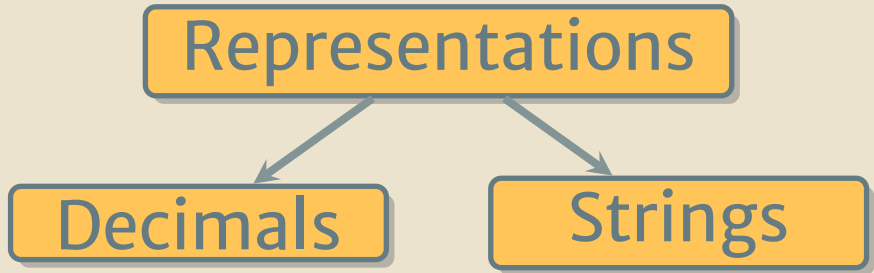
Representations



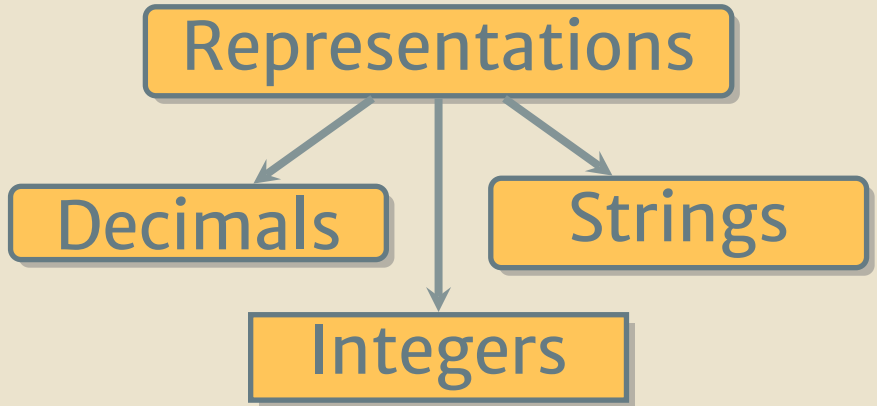
```
graph TD; A[Representations] --> B[Decimals]
```

Decimals

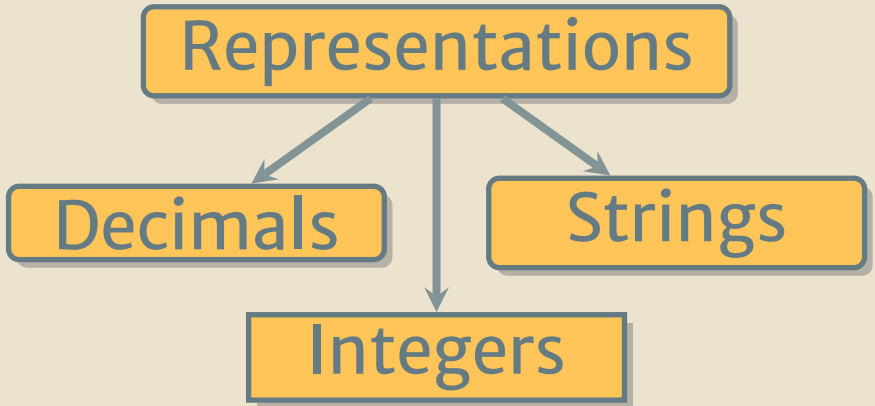
AVM Innovations



AVM Innovations



AVM Innovations



Handle real-world programs

Missing Features



Data?

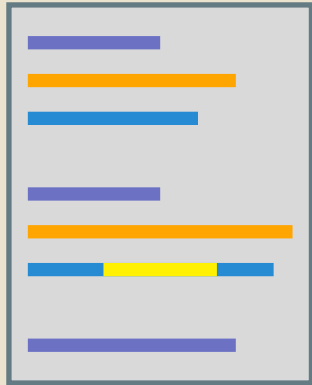


Method?

Missing Features



~~x~~Data?



~~x~~Method?

Key Challenge

Prior AVMs
lack provably
faster methods!

Tools Using AVM

Test
Generation

Tools Using AVM

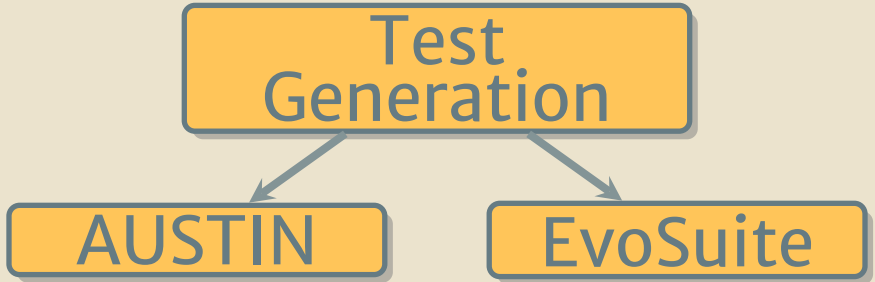


```
graph TD; A[Test Generation] --> B[AUSTIN]
```

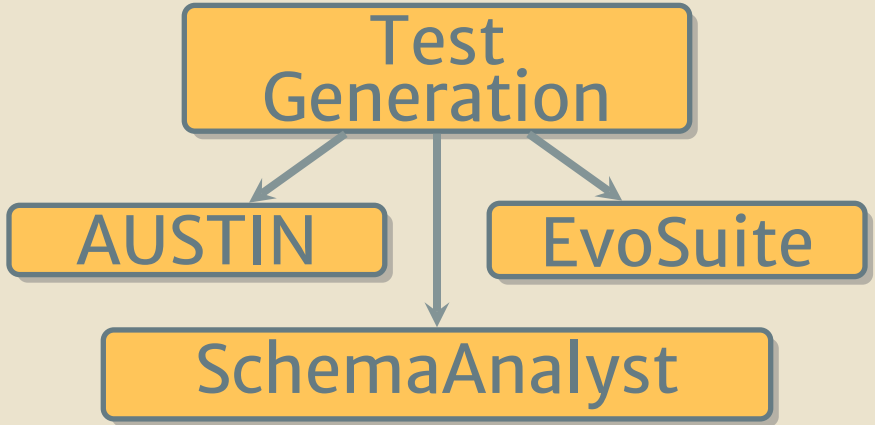
Test
Generation

AUSTIN

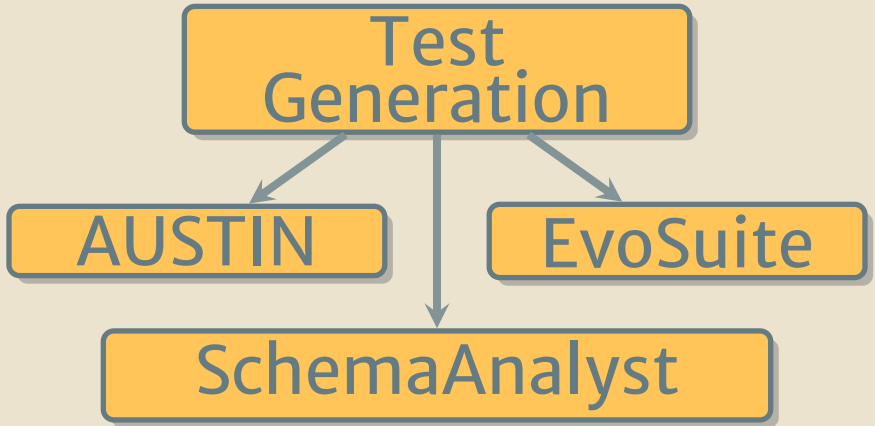
Tools Using AVM



Tools Using *AVM*



Tools Using AVM



AVM is used in many tools

Extracting AVM



Fitness?



Search?

Extracting AVM



Fitness?



Search?

Extracting AVM



Fitness?

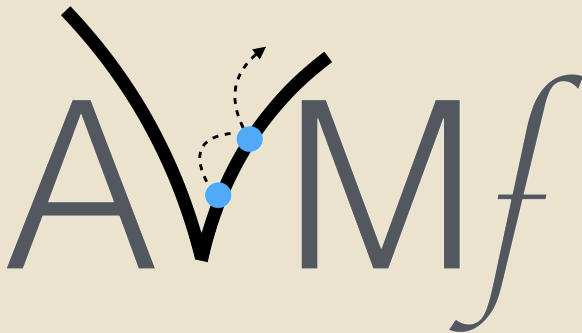


Search?

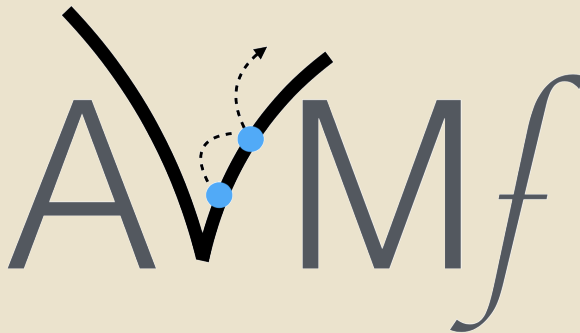
Key Challenge

Hard to
extract AVM
from custom
software!

Rescued by *AVMf*

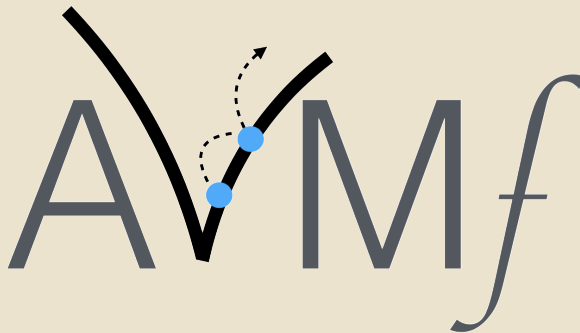


Rescued by *AVMf*



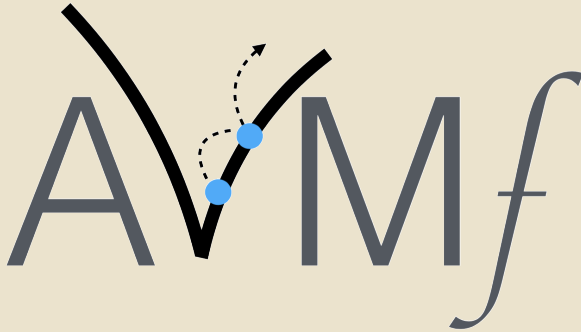
Original AVM plus enhance-
ments for data and search

Rescued by *AVMf*



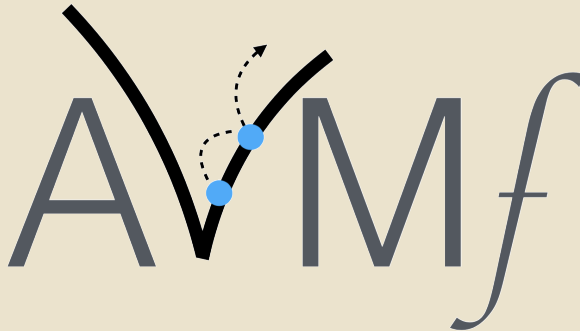
Clear implementa-
tion of core algorithms

Rescued by *AVMf*



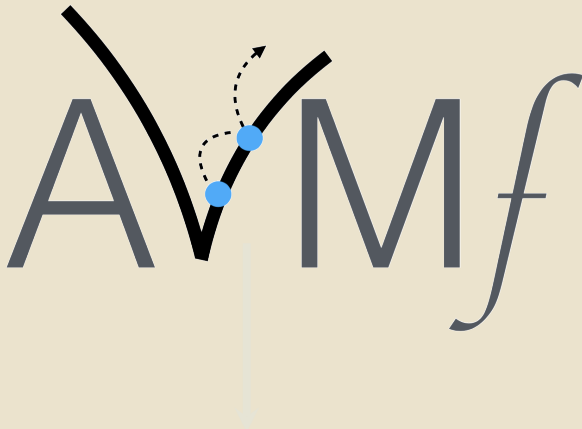
Adheres to the principles
of object-oriented design

Rescued by *AVMf*

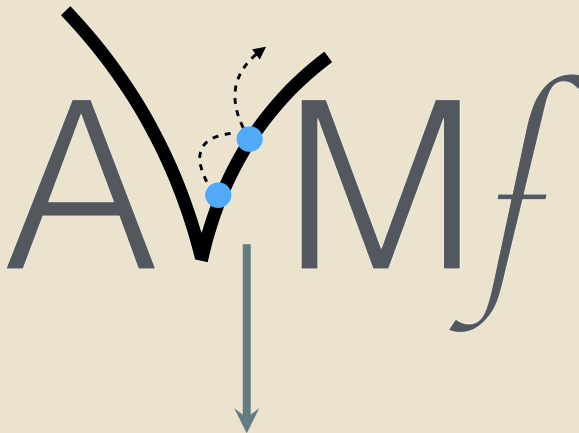


Free and open-source software from avmframework.org

Rescued by *AVMf*

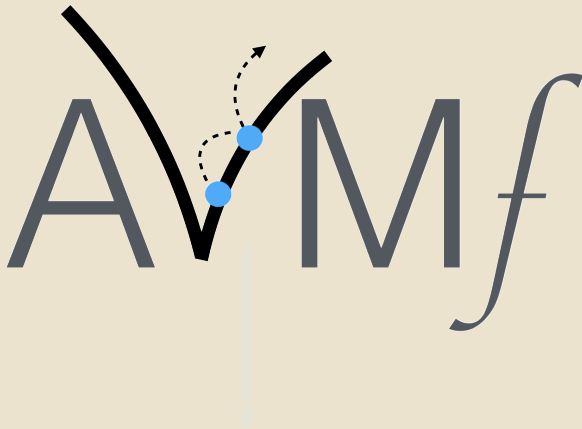


Rescued by *AVMf*

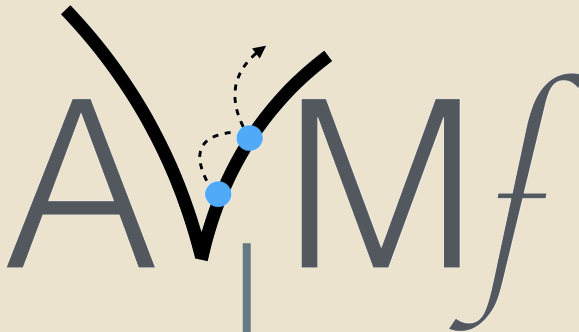


New Application Domain

Rescued by *AVMf*

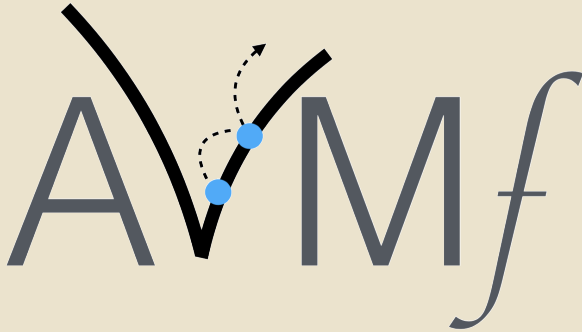


Rescued by *AVMf*

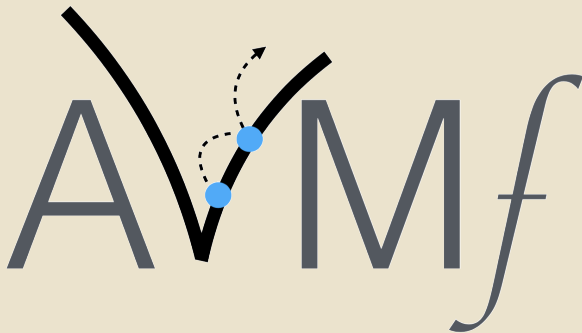


New Search Algorithm

Design of *AVMf*

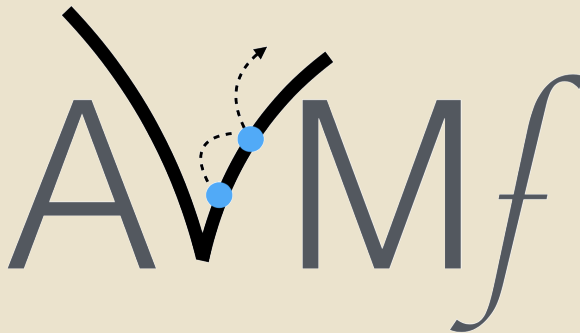


Design of AVM_f



Configure

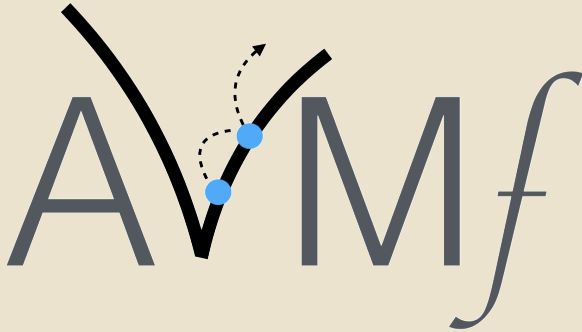
Design of AVM_f



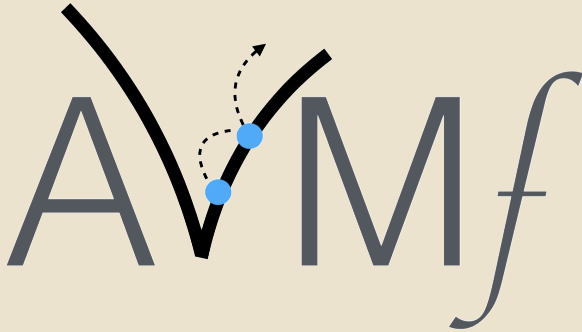
Configure

Represent

Design of *AVMf*

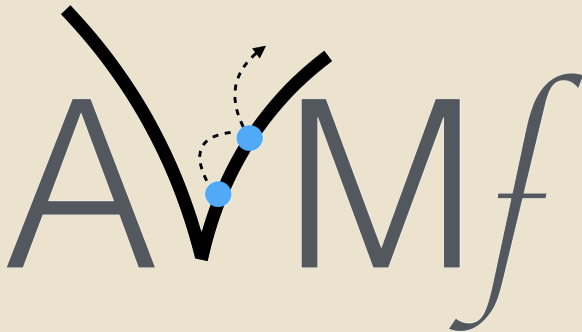


Design of AVM_f



Objective

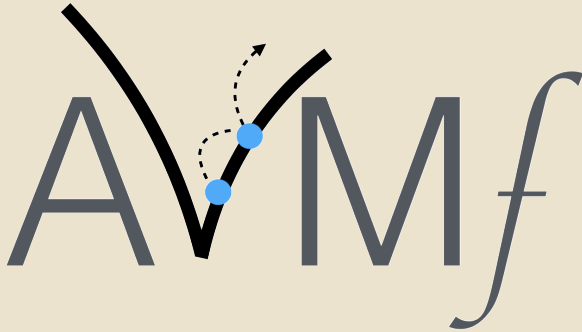
Design of AVM_f



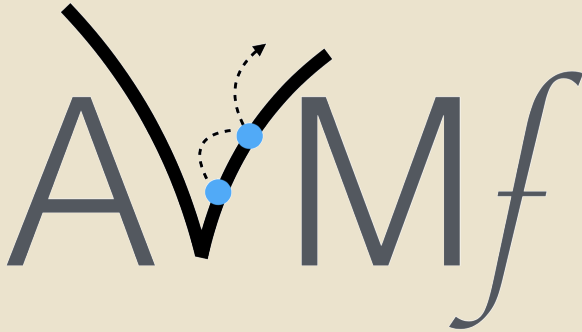
Objective

Report

Design of *AVMf*

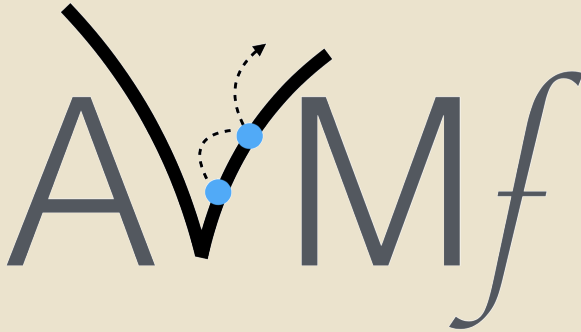


Design of AVM_f



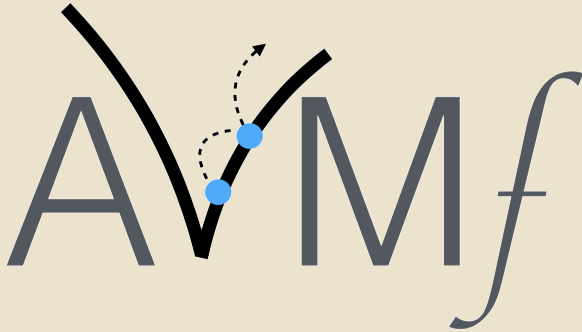
See the paper for more design
and implementation details

Design of *AVMf*

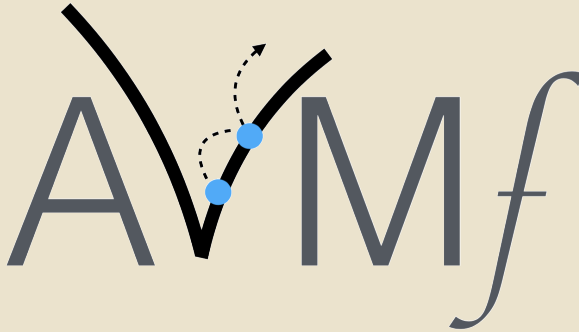


The tool's website contains
extensive documentation

*A*V*M**f* Demonstration

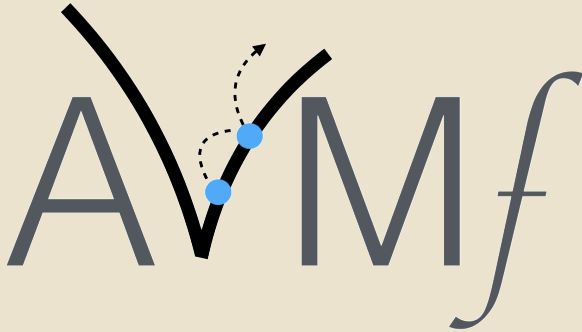


AVMf Demonstration



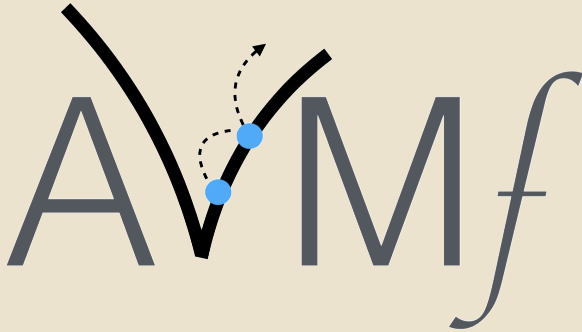
[java.org.avmframework.
examples.Quadratic](http://java.org.avmframework.examples.Quadratic)

AVMf Demonstration



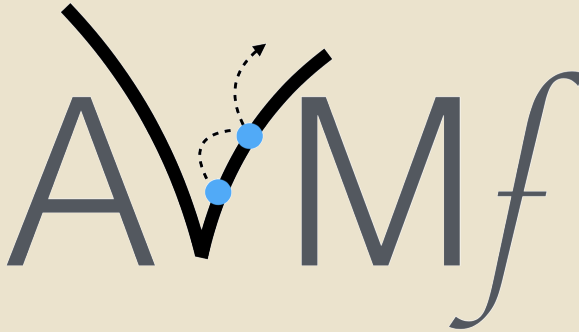
[java.org.avmframework.
examples.StringOptimization](http://java.org.avmframework.examples.StringOptimization)

AVMf Demonstration



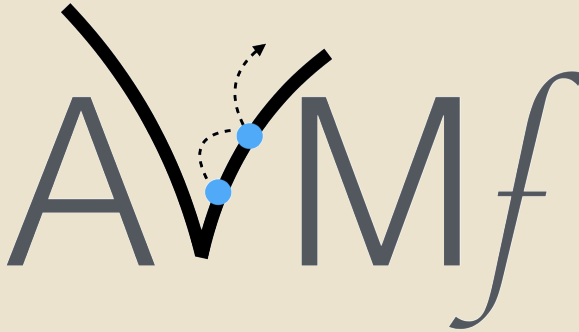
[java.org.avmframework.
examples.GenerateInputData](http://java.org.avmframework.examples.GenerateInputData)

*A*V*M**f* Demonstration



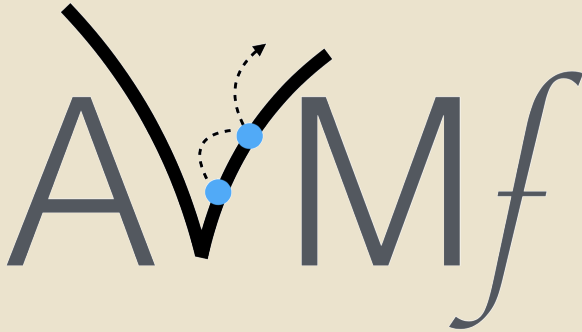
Input → Output
Stochastic Behavior

*A*V*M**f* Demonstration

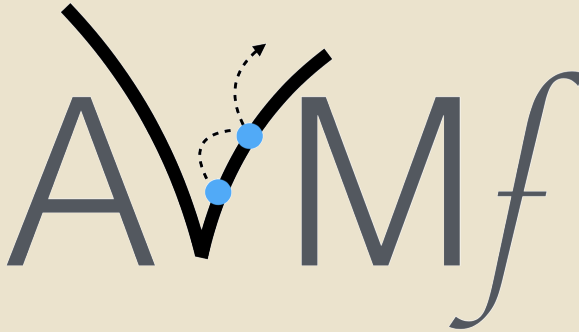


Already run:
git clone & mvn package

AVMf's Contributions

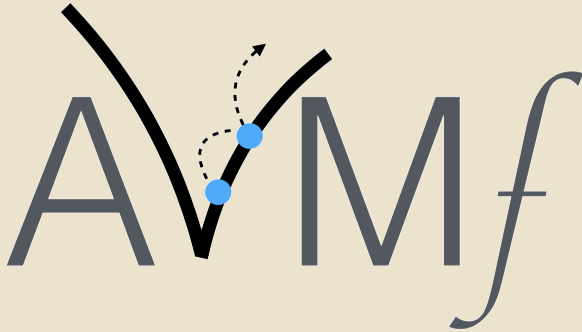


AVMf's Contributions



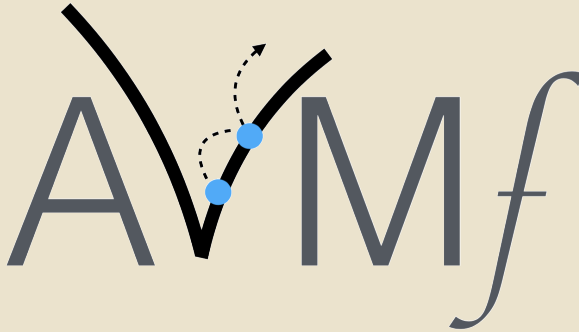
Overcomes the
challenges of using AVM

AVMf's Contributions



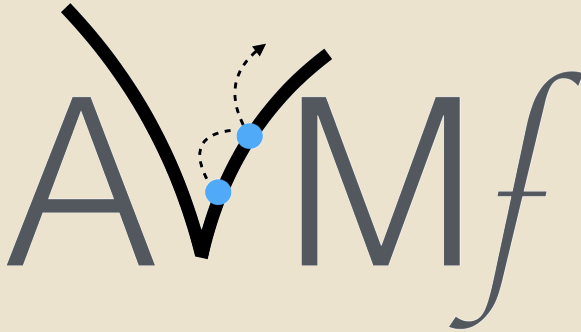
Provably faster searches
and new data types

AVMf's Contributions



Accessible object-oriented
and algorithmic design

AVMf's Contributions



Open-source download
from avmframework.org