Software Validation via Model Animation

Aaron Dutle, César Muñoz, Anthony Narkawicz, Ricky Butler

NASA Langley Research Center,
Hampton, Virginia, US

9th Conference on Tests and Proofs
July 23, 2015

Stratway

Prototype software for Air Traffic Management research:

- Generates trajectories from waypoints.
- Determines if aircraft are in conflict.
- Finds manuevers to resolve conflicts.
- ...

Used at NASA, by industry, academia, and others. Needs to be easy to read, use and extend (Java, C++), but with strong assurance of correctness (PVS).

Stratway

Prototype software for Air Traffic Management research:

- Generates trajectories from waypoints.
- Determines if aircraft are in conflict.
- Finds manuevers to resolve conflicts.
- ...

Used at NASA, by industry, academia, and others. Needs to be easy to read, use and extend (Java, C++), but with strong assurance of correctness (PVS).

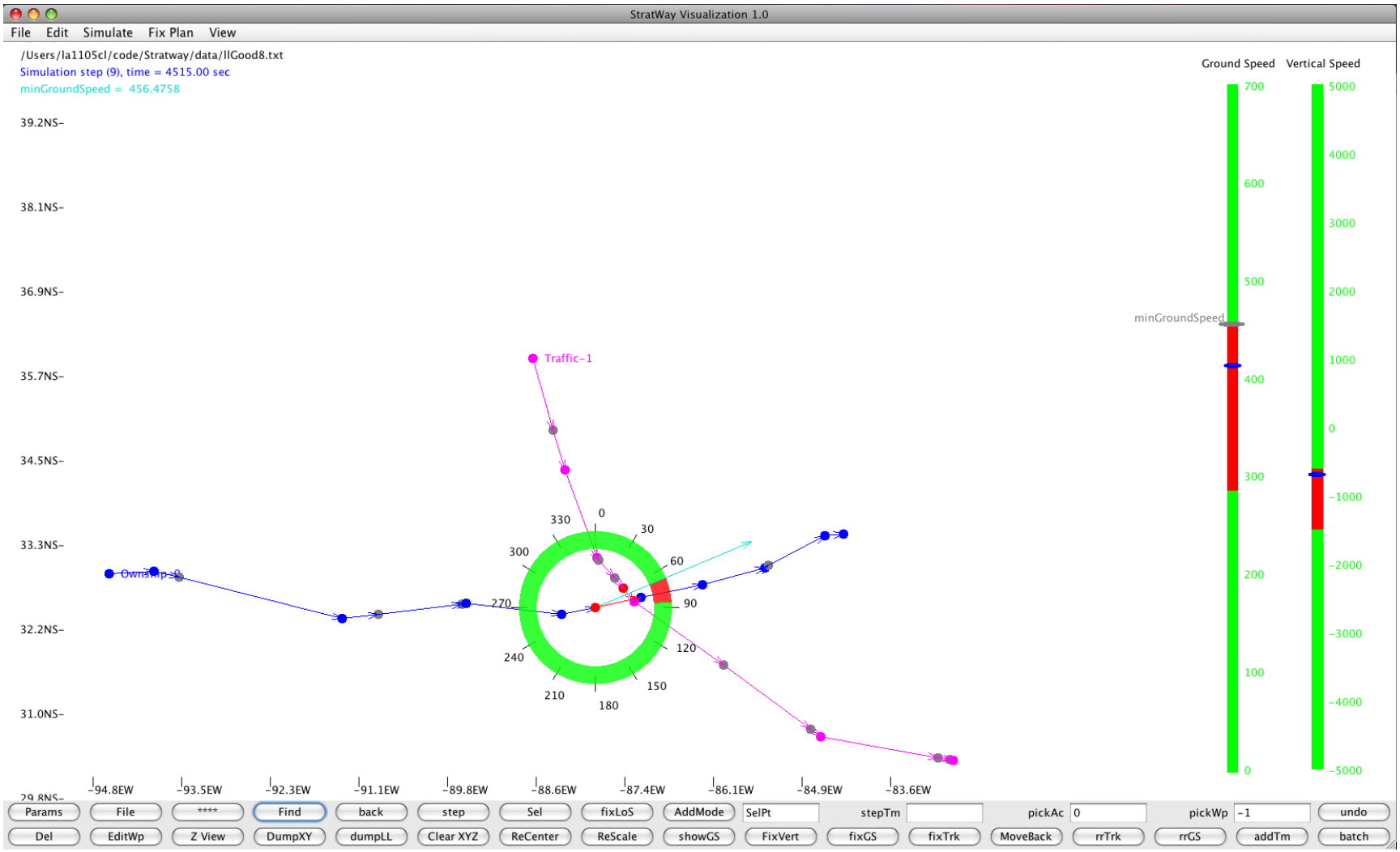
Stratway

Prototype software for Air Traffic Management research:

- Generates trajectories from waypoints.
- Determines if aircraft are in conflict.
- Finds manuevers to resolve conflicts.
- ...

Used at NASA, by industry, academia, and others. Needs to be easy to read, use and extend (Java, C++), but with strong assurance of correctness (PVS).

Stratway

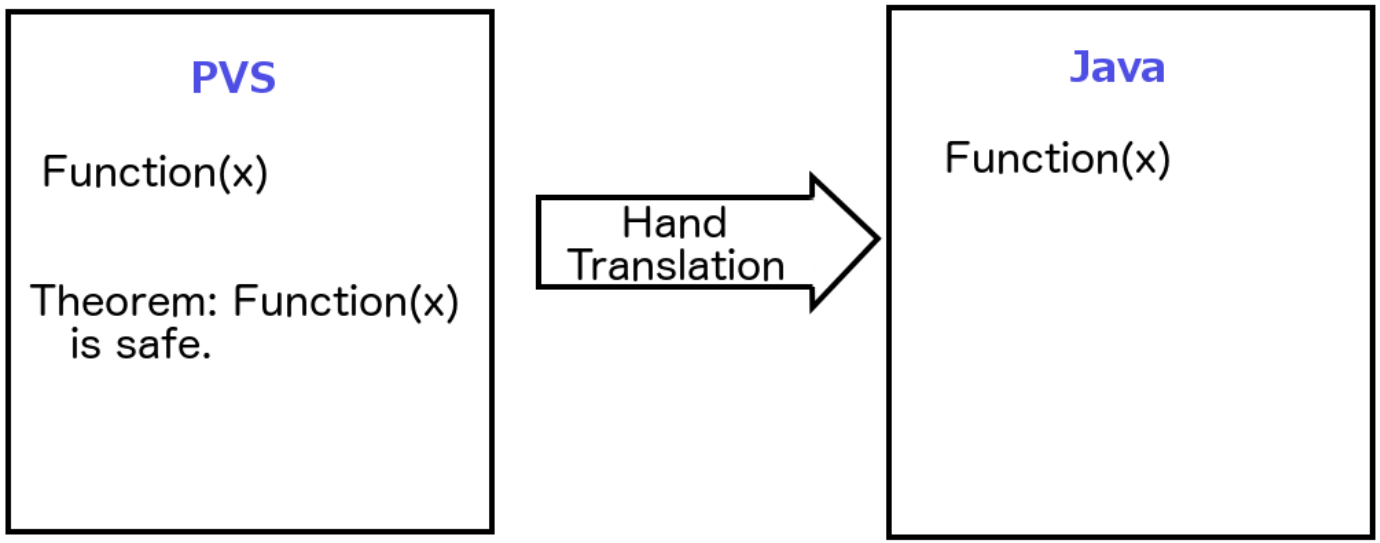

StratWay Visualization 1.0
File Edit Simulate Fix Plan View
/Users/la1105cl/code/Stratway/data/llGood8.txt
Simulation step (9), time = 4515.00 sec
minGroundSpeed = 456.4758
Ground Speed
Vertical Speed
minGroundSpeed
Traffic-1
39.2NS
38.1NS
36.9NS
35.7NS
34.5NS
33.3NS
32.2NS
31.0NS
94.8EW
93.5EW
92.3EW
91.1EW
89.8EW
88.6EW
87.4EW
86.1EW
84.9EW
83.6EW
Params File **** Find back step Sel fixLoS AddMode SelPt stepTm pickAc 0 pickWp -1 undo
Del EditWp Z View DumpXY dumpLL Clear XYZ ReCenter ReScale showGS FixVert fixGS fixTrk MoveBack rrTrk rrGS addTm batch

Stratway

Kinematics library: Provides basic functions for simulating motion.

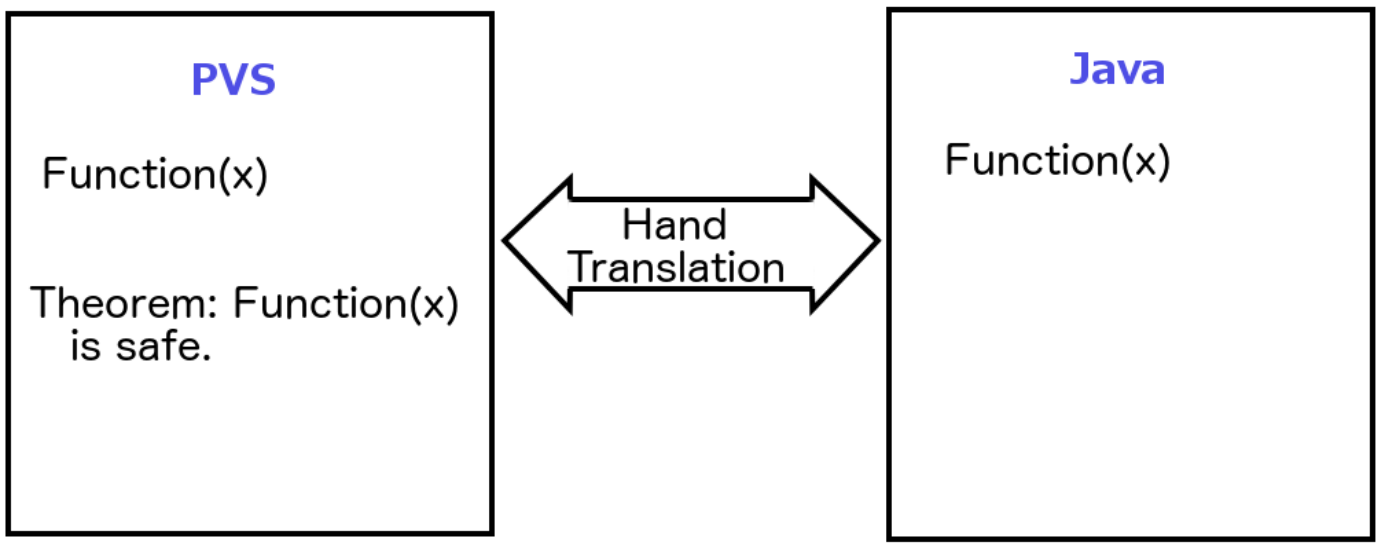
- Constant acceleration vertical maneuvers,
  `vsAccelUntil`
  `vsAccelUntilWithRampUp`
  `vsLevelOut`
- Constant acceleration ground speed maneuvers,
  `gsAccelUntil`
- Circular arc turn maneuvers,
  `turnOmega`
- ...

Current Workflow



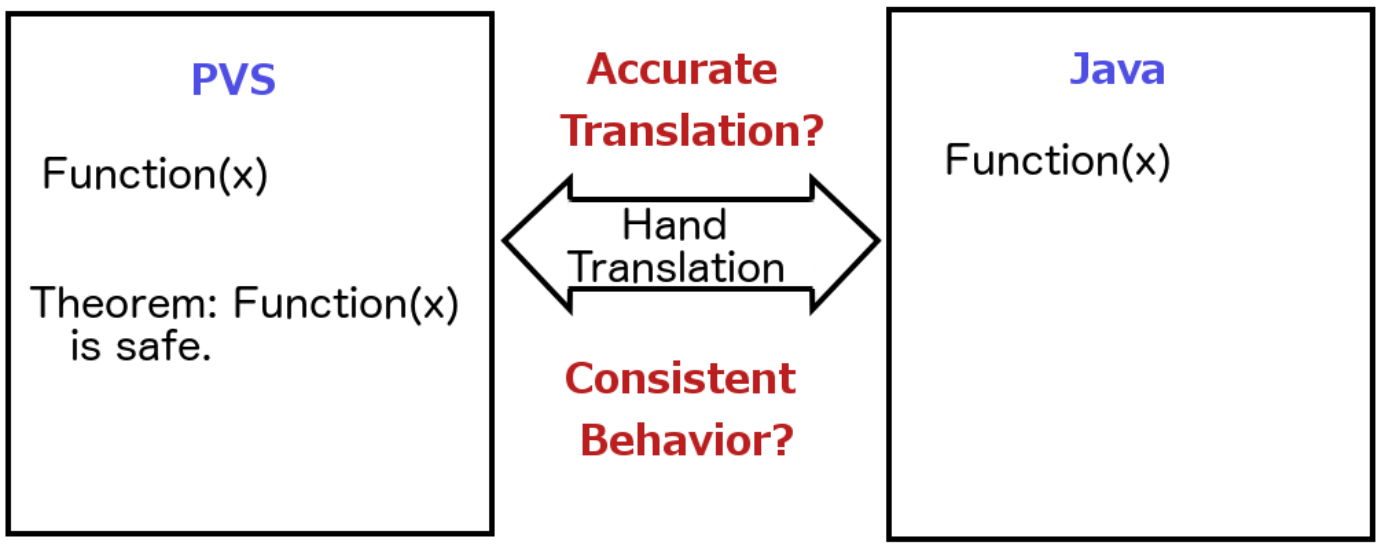
Solution: **Software Validation via Model Animation**. Compare the outputs of the two implementations on an appropriate collection of test points.

Current Workflow



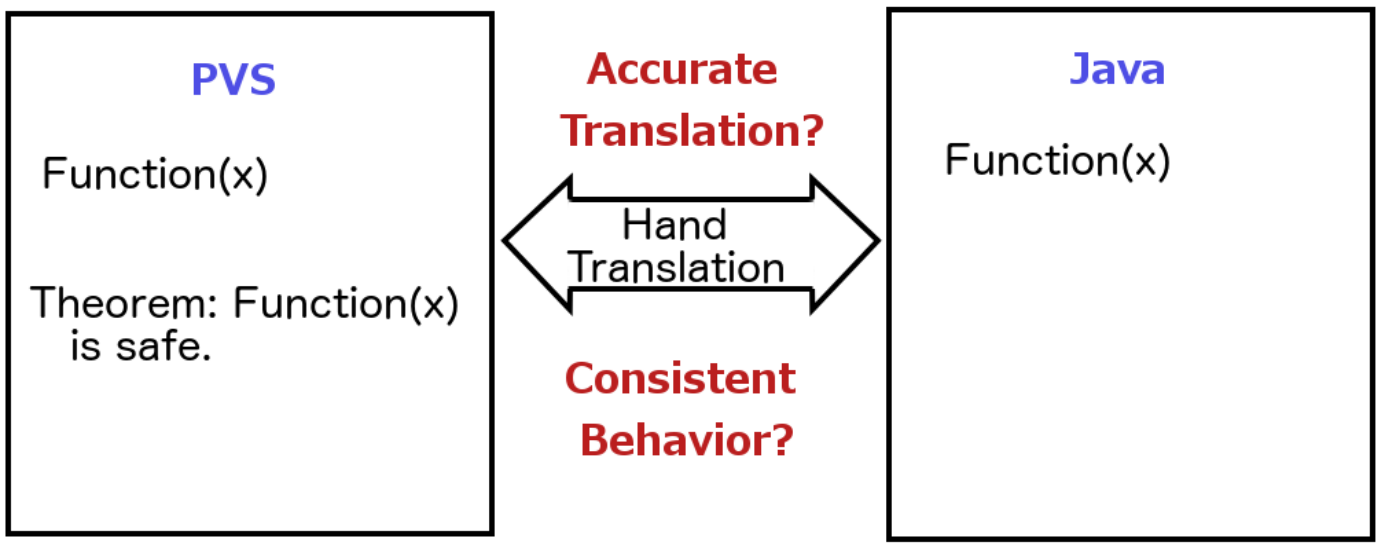
Solution: **Software Validation via Model Animation**. Compare the outputs of the two implementations on an appropriate collection of test points.

Current Workflow



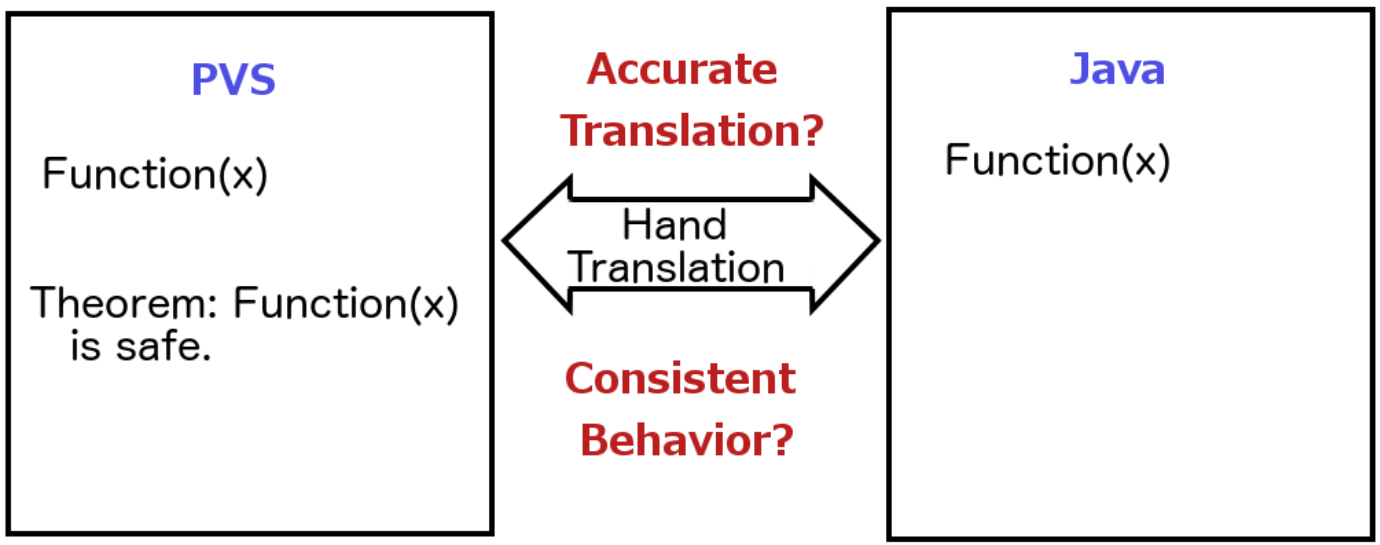
Solution: **Software Validation via Model Animation**. Compare the outputs of the two implementations on an appropriate collection of test points.

Current Workflow



Solution: **Software Validation via Model Animation**. Compare the outputs of the two implementations on an appropriate collection of test points.

Current Workflow



Solution: **Software Validation via Model Animation**. Compare the outputs of the two implementations on an appropriate collection of test points.

Model Animation

Bring specifications to life by evaluating them on concrete inputs.

PVSio = PVS ground evaluator + Semantic attachments.
New semantic attachments for square root, sine, cosine, and arctangent that have

$$|f(x) - \mathtt{f}_{sa}(x)| \leq \epsilon$$

for any user specified ϵ.
Future Goal: Provide guaranteed *output* precision, or upper and lower bounds.

Model Animation

Bring specifications to life by evaluating them on concrete inputs.

PVSio = PVS ground evaluator + Semantic attachments.
New semantic attachments for square root, sine, cosine, and arctangent that have

$$|f(x) - \mathtt{f}_{sa}(x)| \leq \epsilon$$

for any user specified ϵ.
Future Goal: Provide guaranteed *output* precision, or upper and lower bounds.

Model Animation

Bring specifications to life by evaluating them on concrete inputs.

PVSio = PVS ground evaluator + Semantic attachments.
New semantic attachments for square root, sine, cosine, and arctangent that have

$$|f(x) - \mathtt{f}_{sa}(x)| \leq \epsilon$$

for any user specified ϵ.
Future Goal: Provide guaranteed *output* precision, or upper and lower bounds.

Model Animation

Bring specifications to life by evaluating them on concrete inputs.

PVSio = PVS ground evaluator + Semantic attachments.
New semantic attachments for square root, sine, cosine, and arctangent that have

$$|f(x) - \mathtt{f}_{sa}(x)| \leq \epsilon$$

for any user specified ϵ.
Future Goal: Provide guaranteed *output* precision, or upper and lower bounds.

Model Animation

PVSioChecker Automates common tasks for model animation

- Reading and writing of files,
- Converting inputs to exact rationals,
- Comparison of values to user-specified precision,
- Aggregating and printing error, timing, and other information.

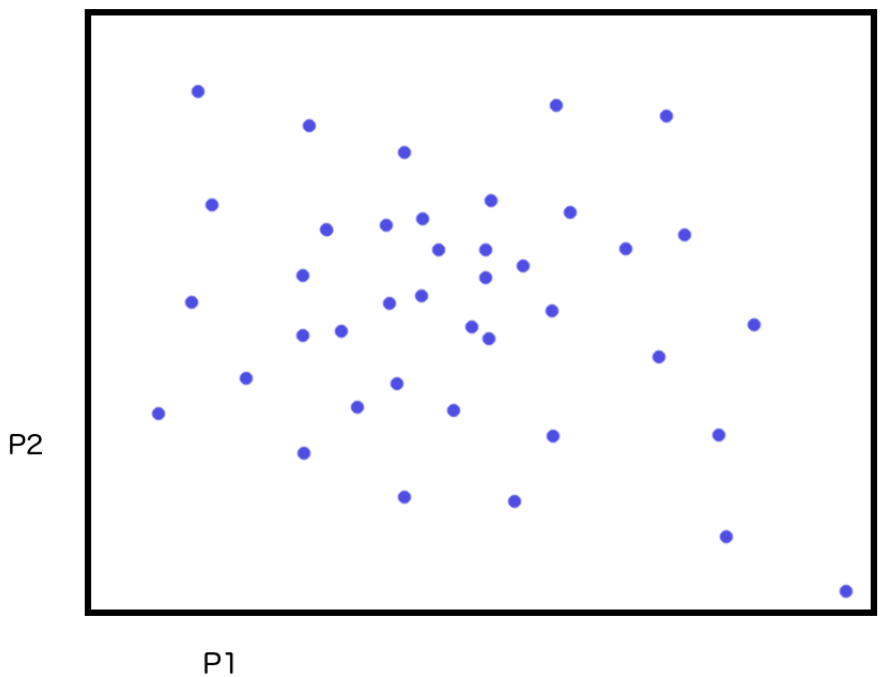
Test Generation

- Test cases can be any set based on the situation, generated by any means.
- Three methods used, each implemented for each function with a Java program.

Test Generation

- Test cases can be any set based on the situation, generated by any means.
- Three methods used, each implemented for each function with a Java program.

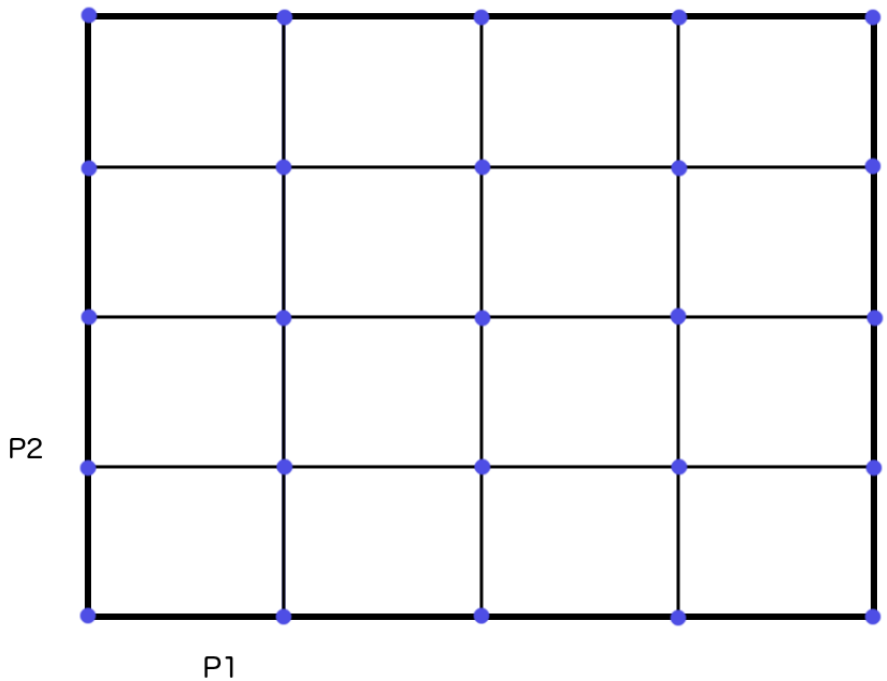
Random



Test Generation

- Test cases can be any set based on the situation, generated by any means.
- Three methods used, each implemented for each function with a Java program.

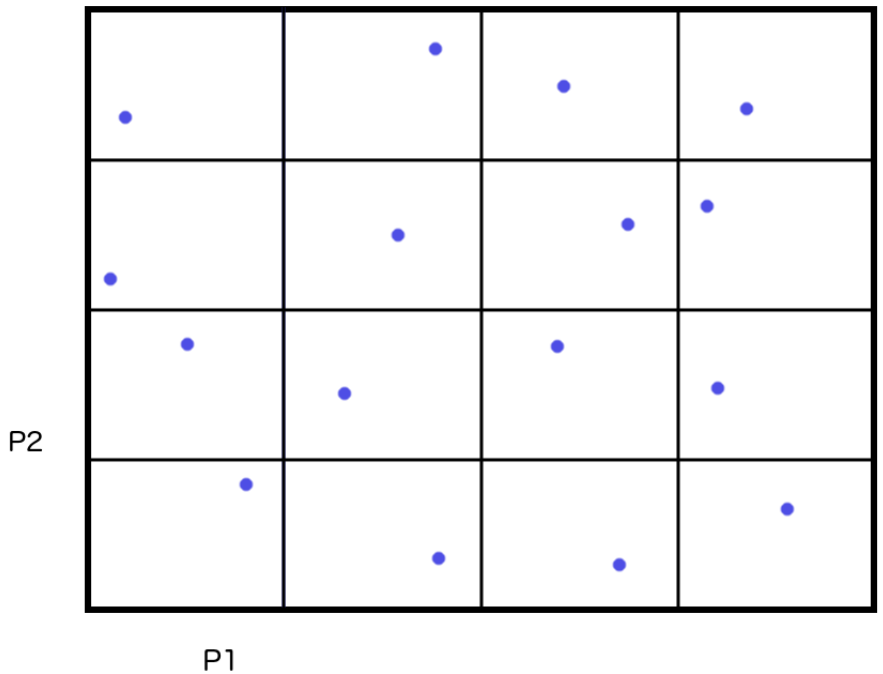
Grid



Test Generation

- Test cases can be any set based on the situation, generated by any means.
- Three methods used, each implemented for each function with a Java program.

Grid-Random



Results

`vsAccelUntil`

	Records	Fails	CPU time
Rand	1,000,000	0	11.32 hr
Grid	622 ,080	0	4.11 hr
G-R	332,659	0	2.88 hr
totals	1,954,739	0	18.31 hr

`vsAccelUntilWithRampUp`

	Records	Fails	CPU time
Rand	960,000	0	11.7 hr
Grid	340,416	0	2.45 hr
G-R	665,429	0	6.48 hr
totals	1,965,845	0	20.63 hr

`vsLevelOut`

	Records	Fails	CPU time
Rand	810,000	0	11.53 hr
Grid	518,400	0	4.88 hr
G-R	915,000	8	11.42 hr
totals	2,243,400	8	27.83 hr

`gsAccelUntil`

	Records	Fails	CPU time
Rand	330,000	0	12.29 hr
Grid	315,000	0	11.8 hr
G-R	340,000	0	11.7 hr
totals	985,000	0	35.79 hr

`turnOmega`

	Records	Fails	CPU time
Rand	615,000	225	13.06 hr
Grid	504,000	300	7.89 hr
G-R	436,066	309	8.4 hr
totals	1,555,066	834	29.35 hr

Global Totals

	Records	Fails	CPU time
Rand	3,715,000	225	59.9 hr
Grid	2,299,896	300	31.13 hr
G-R	2,689,154	317	40.88 hr
totals	8,704,050	842	131.91 hr

Failure tolerance $= 10^{-8}$, attachment precision $= 10^{-15}$. Less than 0.01% failure rate, concentrated in `turnOmega`.

Results

`vsAccelUntil`

	Records	Fails	CPU time
Rand	1,000,000	0	11.32 hr
Grid	622 ,080	0	4.11 hr
G-R	332,659	0	2.88 hr
totals	1,954,739	0	18.31 hr

`vsAccelUntilWithRampUp`

	Records	Fails	CPU time
Rand	960,000	0	11.7 hr
Grid	340,416	0	2.45 hr
G-R	665,429	0	6.48 hr
totals	1,965,845	0	20.63 hr

`vsLevelOut`

	Records	Fails	CPU time
Rand	810,000	0	11.53 hr
Grid	518,400	0	4.88 hr
G-R	915,000	8	11.42 hr
totals	2,243,400	8	27.83 hr

`gsAccelUntil`

	Records	Fails	CPU time
Rand	330,000	0	12.29 hr
Grid	315,000	0	11.8 hr
G-R	340,000	0	11.7 hr
totals	985,000	0	35.79 hr

`turnOmega`

	Records	Fails	CPU time
Rand	615,000	225	13.06 hr
Grid	504,000	300	7.89 hr
G-R	436,066	309	8.4 hr
totals	1,555,066	834	29.35 hr

Global Totals

	Records	Fails	CPU time
Rand	3,715,000	225	59.9 hr
Grid	2,299,896	300	31.13 hr
G-R	2,689,154	317	40.88 hr
totals	8,704,050	842	131.91 hr

Failure tolerance $= 10^{-8}$, attachment precision $= 10^{-15}$. Less than 0.01% failure rate, concentrated in `turnOmega`.

Conclusion

Model animation

- A practical way to show that an implementation agrees with its specification.
- Bridges the semantic gap between theorem provers and common programming languages.
- Mitigates concerns over numerical (floating point) errors in software implementations.

Thanks for your attention!