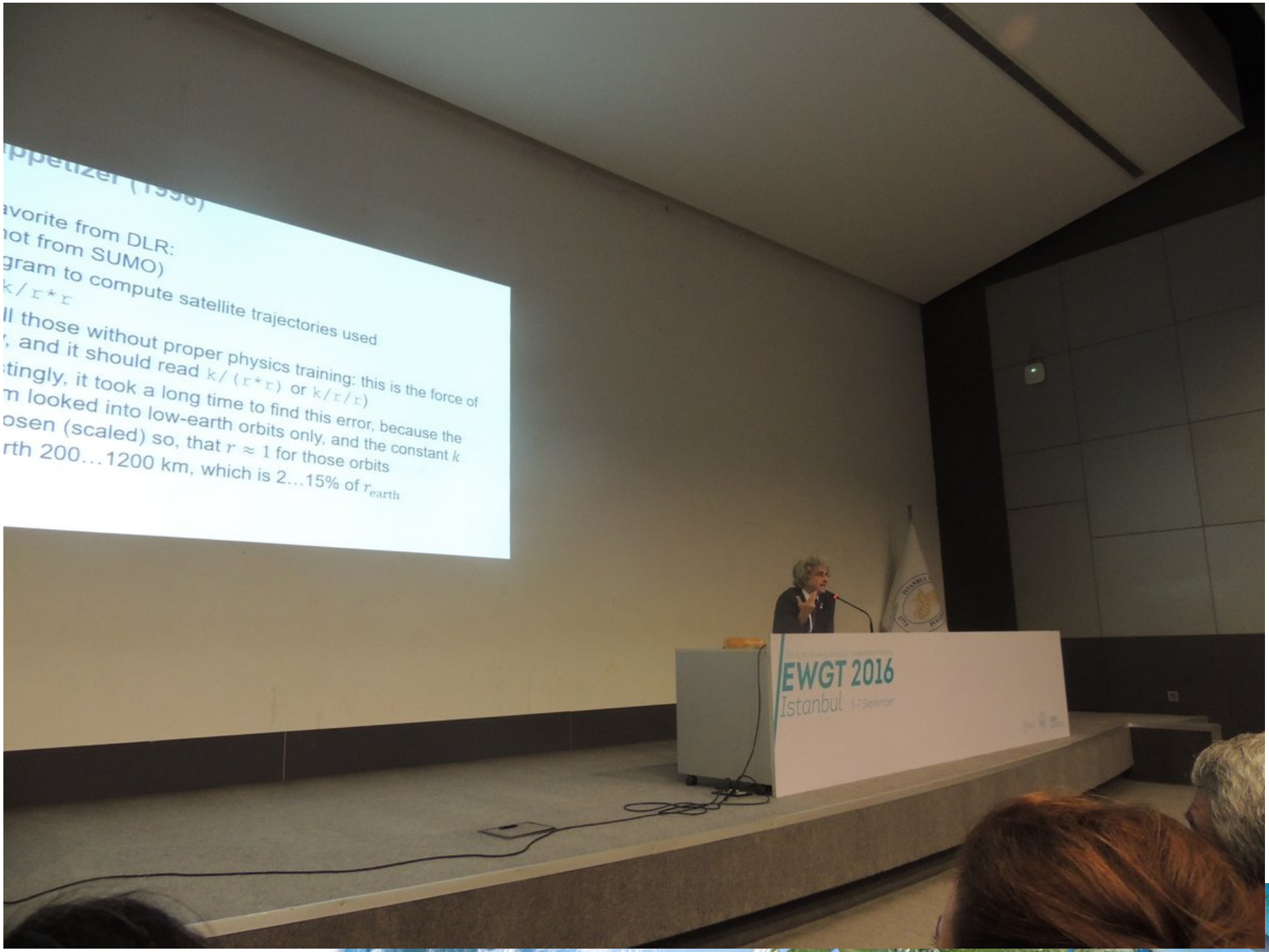
Towards Reliability in the Usage of Traffic Simulation Tools

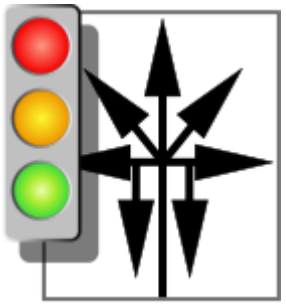
Peter Wagner, Institute of Transport Systems, DLR
and TU Berlin, Institute of Land- and Sea-Traffic

European Workgroup on Transportation (EWGT) 2016
Istanbul, Turkey
5 – 7 September 2016

DLR


avorite from DLR:
ot from SUMO)
gram to compute satellite trajectories used
k/r*r
ll those without proper physics training: this is the force of
, and it should read k/(r*r) or k/r/r)
tingly, it took a long time to find this error, because the
m looked into low-earth orbits only, and the constant k
osen (scaled) so, that r ≈ 1 for those orbits
rth 200...1200 km, which is 2...15% of r_earth
EWGT 2016
Istanbul

You all know…



http://sumo.dlr.de

- Traffic simulation is a great tool!
- Especially this one
- BUT: Traffic simulation, as any simulation, is the incarnation of a certain (traffic) model in software
- So, this approach has three possible sources of bugs:
  - The embedded models
  - The simulation software itself
  - And the ones using the simulation
- My specialty is in models, so this presentation will be biased.
- Lets start with three theses…

George E. P. Box: “essentially, all models are wrong, but some are useful”

http://www.allmodelsarewrong.com

ALL MODELS ARE WRONG

...but some are useful. A grown-up discussion about how to quantify uncertainties in modelling climate change and its impacts, past and future.

Box‘s statement is neat…

So, I may paraphrase:

- All Software is buggy, but some give correct results
- All users are clueless, but some of them do the right thing

Let us take a closer look at these three:
Software, Users, and Models

Software

Software IS buggy…

- With horrible lists with the greatest failures
- http://www.testlab4apps.com/major-quality-assurance-fails-and-solutions-of-the-past-25-years-infographic/
- And of course, all of you know the old joke:
  If the automobile industry had developed like the software industry we would all be driving $25 cars that get 1,000 miles to the gallon.
- Yeah, and if cars were like software, they would crash twice a day for no reason, and when you called for service, they'd tell you to reinstall the engine.
- However, there is a much longer list of things that were impossible to do without it

https://www.technologyreview.com/s/401594/why-software-is-so-bad/

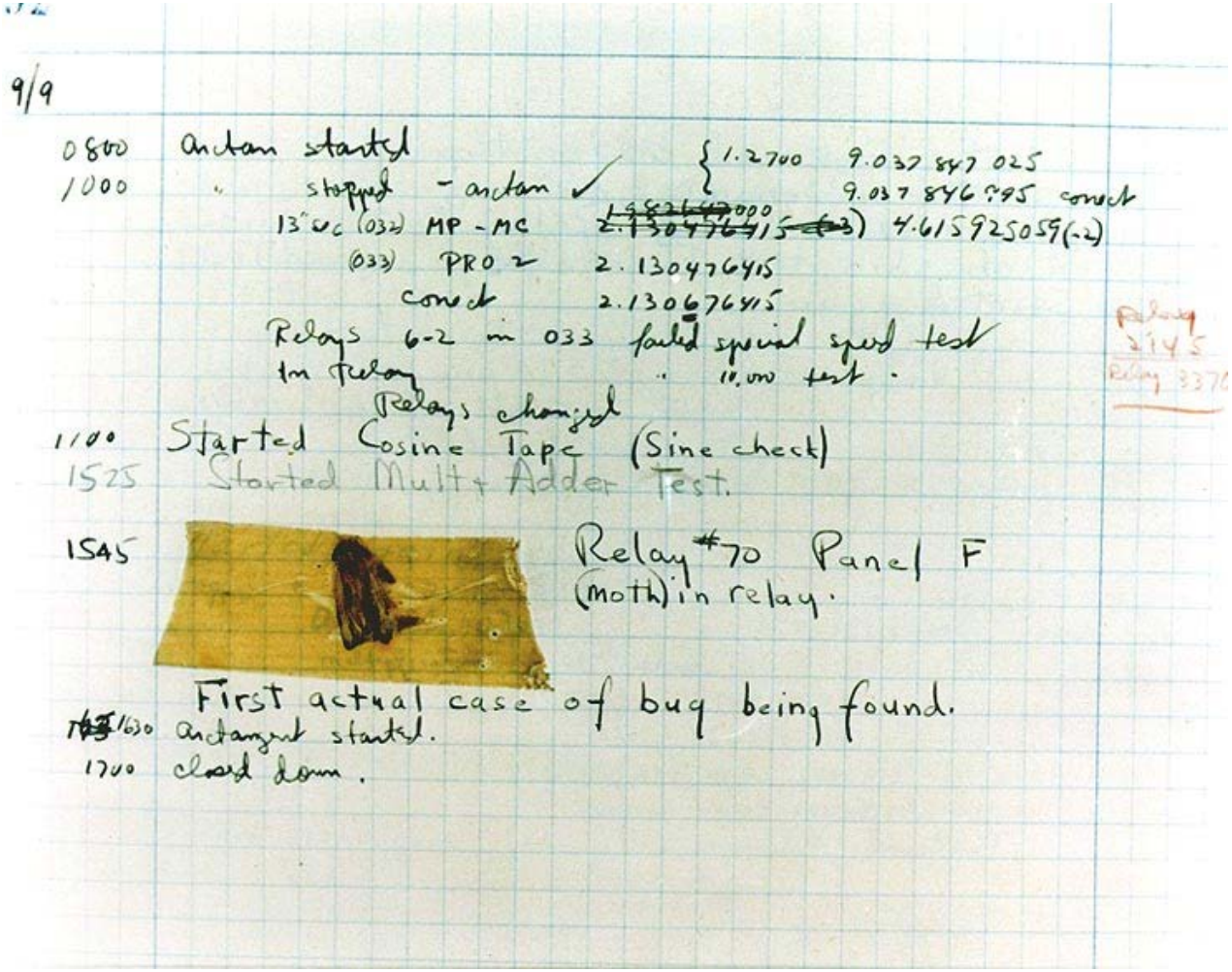
Who invented the bug? – “Amazing Grace“ Hopper

Admiral Grace Hopper liked to tell a story in which a technician solved a glitch in the Harvard Mark II mainframe by pulling an **actual insect** out from between the contacts of one of its relays

~ The Jargon File

http://www.slideshare.net/noahsussman/software-entomology-or-where-do-bugs-come-from

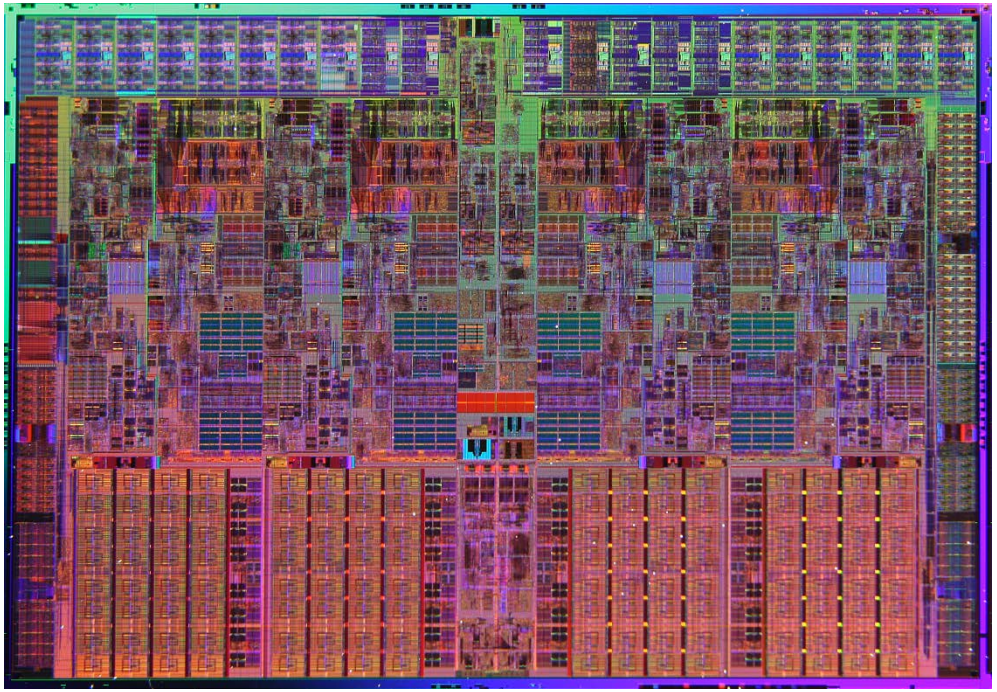
1st recorded bug in computer history



Courtesy of the Naval Surface Warfare Center, Dahlgren, VA., 1988. - U.S. Naval Historical Center Online Library Photograph NH 96566-KN, Gemeinfrei, https://commons.wikimedia.org/w/index.php?curid=165211

Good old times…

- Were you could repair a computer manually by picking a bug from its interns.
- Try this with a modern chip ☺



- Nevertheless, three glitches in software which I found particularly interesting.
- (Was hard to choose, I had many candidates)

An appetizer (1996)

- My favorite from DLR:
  (no, not from SUMO)
- A program to compute satellite trajectories used
- `F = k/r*r`
- (For all those without proper physics training: this is the force of gravity, and it should read `k/(r*r)` or `k/r/r`)
- Interestingly, it took a long time to find this error, because the program looked into low-earth orbits only, and the constant k was chosen (scaled) so, that $r \approx 1$ for those orbits
- Low-earth 200…1200 km, which is 2…15% of r_{earth}

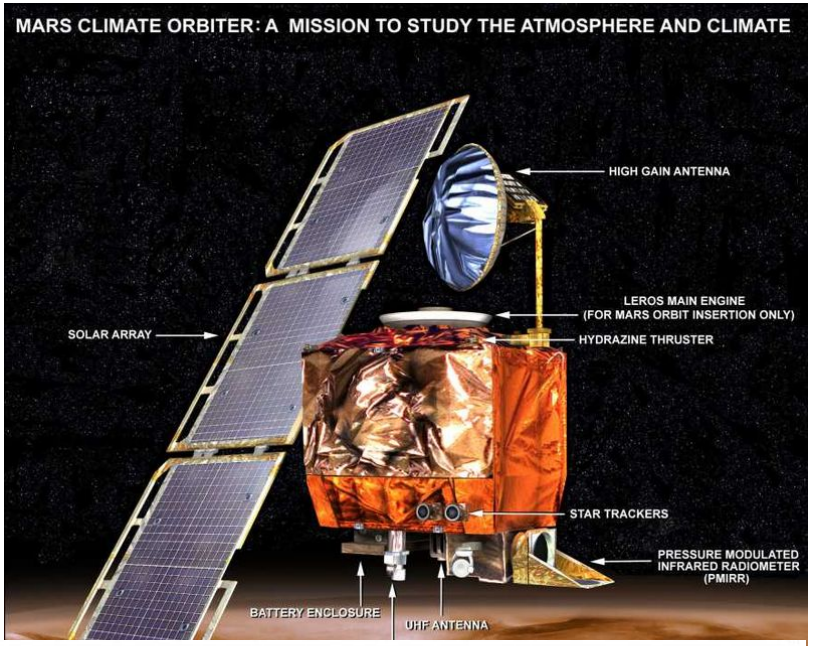
Jurassic park, 1990

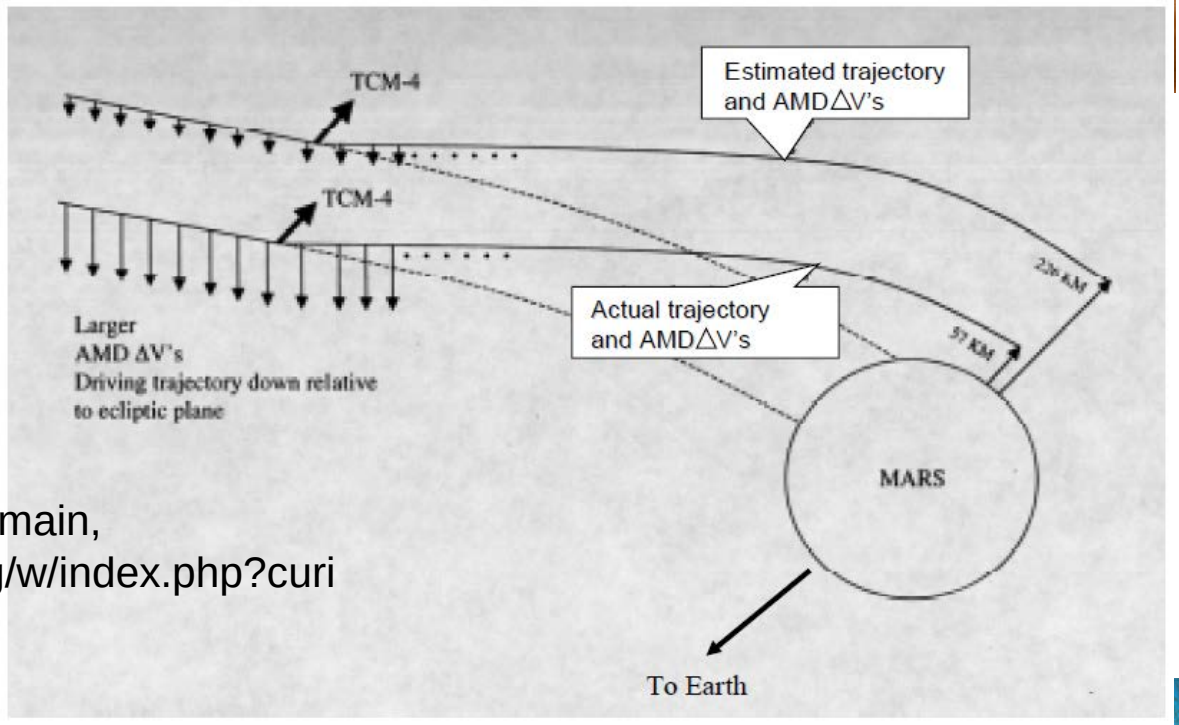
- Have you read the book?
- (It is also in the movie, but much harder to detect.)
- The stability of the park depended on a constant number of dinosaurs
- They had an software implemented to count the beasts
- It worked by recognizing them, computationally very costly
- They were absolutely sure they could not breed, so they looked for shrinking numbers and missed their number was increasing, putting the park in peril.
- Why?
- Software to do so had hard coded:
- **`Const Integer MAX_NR_DINOS = 222`**

By Dave Pape - Own work, Public Domain, https://commons.wikimedia.org/w/index.php?curid=1328267

Mars Climate Orbiter (MCO) (1999)

- 170 km too close to Mars
- Famous for the mixture of metric and English units
- BUT: the story is a little bit more complex:
- MCO had only one solar-cell panel
- …





From NASA - NASA, Public domain, https://commons.wikimedia.org/w/index.php?curid=27798439

Little irony

- A computer magazine (I don‘t tell you the name) explained the year 2038 “bug“ as a counter that jumps from
- “111 1111 1111 1111 1111 1111 1111 1111 11112” to “000 0000 0000 0000 0000 0000 0000 0000 00002”
- A reader notified them about the 2 in a binary number (article is from May 2016), but they did not correct it (Sep 2016)

- Have you spotted the second error (I used <ctrl><c> and <ctrl><v> to copy it here, avoiding another error by me)?

Little irony

- A computer magazine (I don‘t tell you the name) explained the year 2038 “bug“ as a counter that jumps from
- "111 1111 1111 1111 1111 1111 1111 1111 11112" to "000 0000 0000 0000 0000 0000 0000 00002"
- A reader notified them about the 2 in a binary number (article is from May 2016), but they did not correct it (Sep 2016)

- Have you spotted the second error (I used <ctrl><c> and <ctrl><v> to copy it here, avoiding another error by me)?
- Which is the perfect transition to the next point on my list

Users

I disappoint you – just one slide

- If users of the intended audience cannot use the software effectively, where does the blame lie?
- http://infodesign.com.au/usabilityresources/articles/themythofthestupiduser/

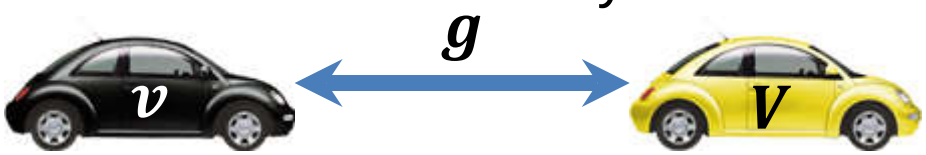
- But, nevertheless, they are sometimes interested in features, that are simply not possible, or morally questionable, or stupid, or already there…
- One of the SUMO developers put this once: they want a “deliver-my-thesis-now“ button in the software
- Or, a bit more high-level: Myhrvold (former CTO @ Microsoft): “Software sucks because users demand it to.”

And finally, models.

Examples, only. No general theory.

Awful: an example from the heart of μ-Sim

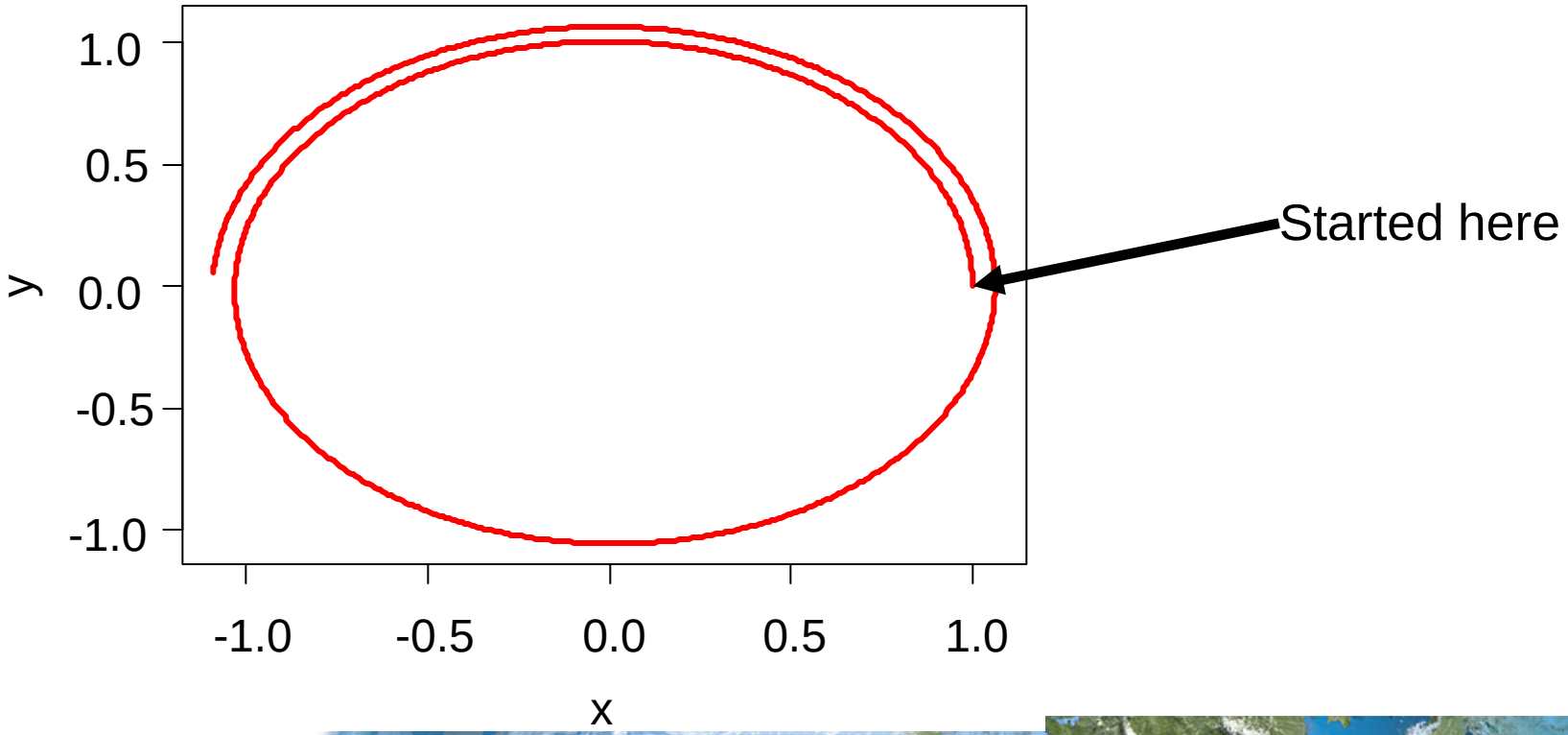
- A physicist by training, I have learned how to correctly solve differential equations (ODE)
- In traffic, people do this:



- They have an ODE, which is assumed to model a human driver:
- $a = \frac{d}{dt}v = \dot{v} = f(g, v, V)$ a acceleration; g distance; v, V speeds
- And then they do the most elementary discretization, the so called Euler discretization, to “solve“ it:
- $v(t + \Delta t) = v(t) + \Delta t f(g, v, V)$
- Nobody cares about problems related to this approach…
- Ignoring completely, that there is a very detailed theory how to do this correctly

Euler is bad

- When you use Euler to compute the trajectory of a planet orbiting the sun, it turns out completely wrong
- (Try it yourself! It is fairly easy, just a few lines of code in R)



Euler is bad

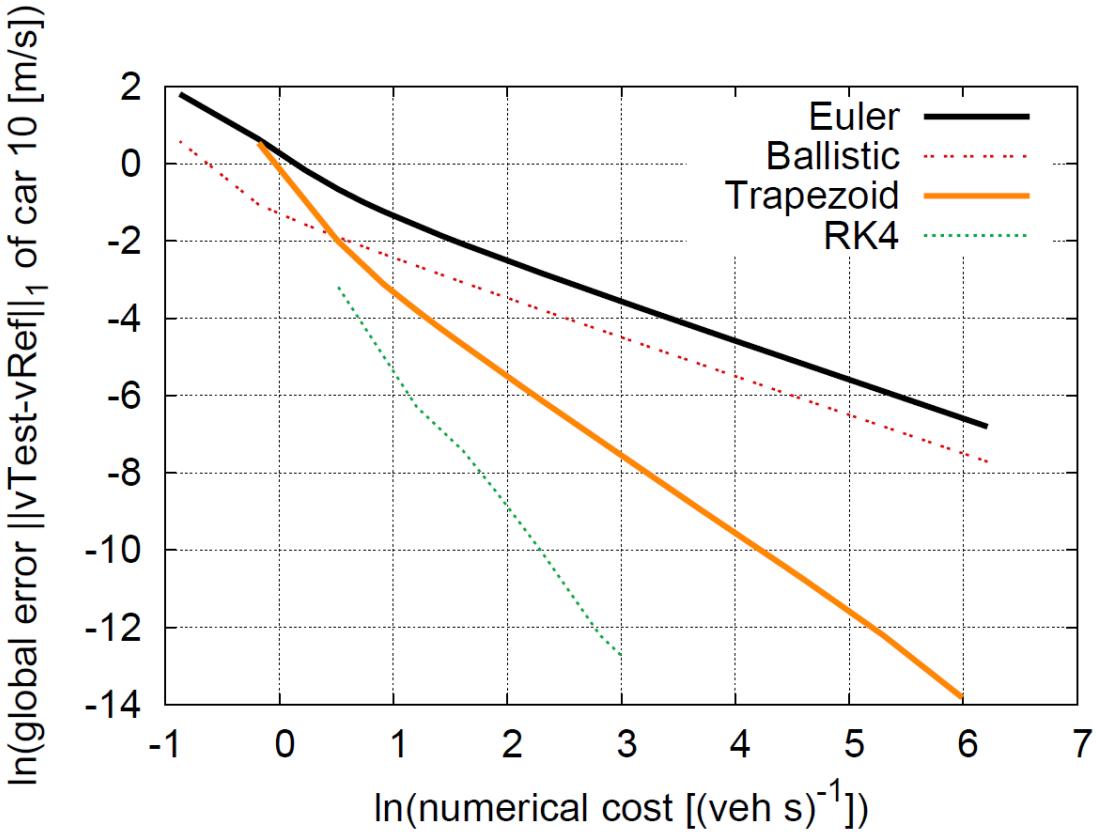
- When you use Euler to compute the trajectory of a planet orbiting the sun, it turns out completely wrong
- (Try it yourself! It is fairly easy, just a few lines of code in R)

- Fortunately, traffic equations have two features that make them robust against the wrong-doing of a bad integration routine:
  - They describe a dissipative and driven system, which is different from the planetary motion example
  - We do not know the correct model, so any misdoing is swamped into the difference to reality anyway
- There is a third feature that helps, but not anybody agrees with me on this: those μ-Sim models are stochastic

Euler is o.k.

Treiber & Kanagaraj (2015) *Comparing Numerical Integration Schemes for Time-Continuous Car-Following Models*, Physica A **419C**, 183-195

- Nobody cares? No! Martin Treiber & Venkatesan Kanagaraj do:
- Truly nice work to compare different integration schemes
- Good news:
- Differences exist, but they are not dramatic and do not change much the total outcome
- At least where they have looked!



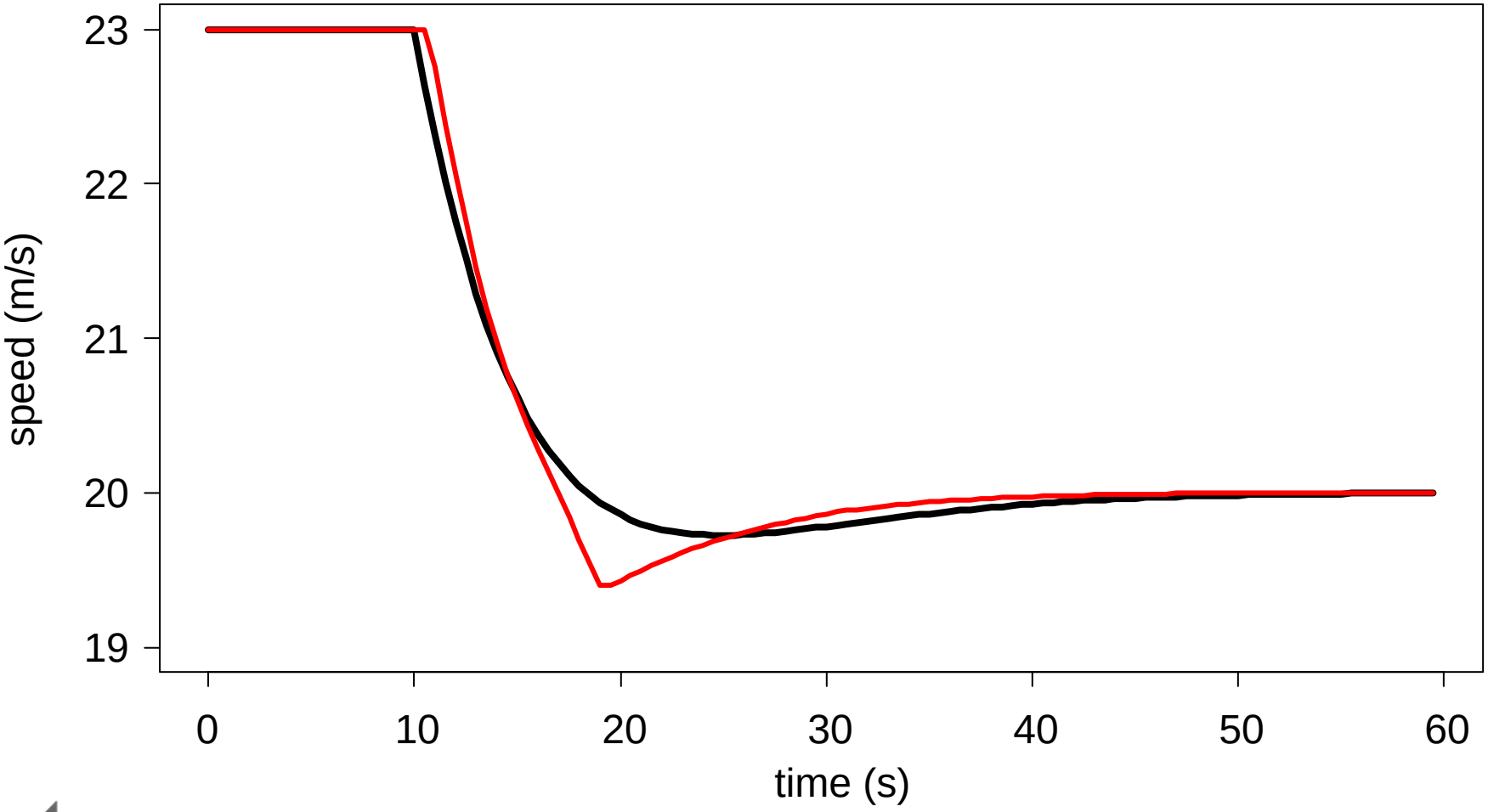
Still…

- They have not used the state-of-the-art, which is a higher order scheme PLUS step-size control
- All this is implemented now in BOOST, and so, if you manage BOOST, life becomes very simple to do state-of-the-art integration of ODE‘s
- (At least, if you are a C++ programmer)
- However, we do not have any plans to change our favorite μ-Sim tool from its ballistic update:
- When considering vehicles in different parts of the network anything more complicated than ballistic becomes a night-mare
- However, there might examples where it is needed: watch out!

Euler is o.k., especially after I told you this…

- One additional feature, again not anybody is happy with it: humans do not work like a controller that can be described by an ODE:
- $a = \frac{d}{dt}v = \dot{v} = f(g, v, V)$
- they work on a discrete-event based description called action-points (AP's)
- And if you agree with me on this, then the ballistic update is (almost) exact.
- However, you have to know which is the algorithm causing humans to choose an AP
- (Almost: well, even if gas-pedal is constant, acceleration of the car is subject to e.g. air-drag, and this ***is*** continuous)

Is it important? It depends…

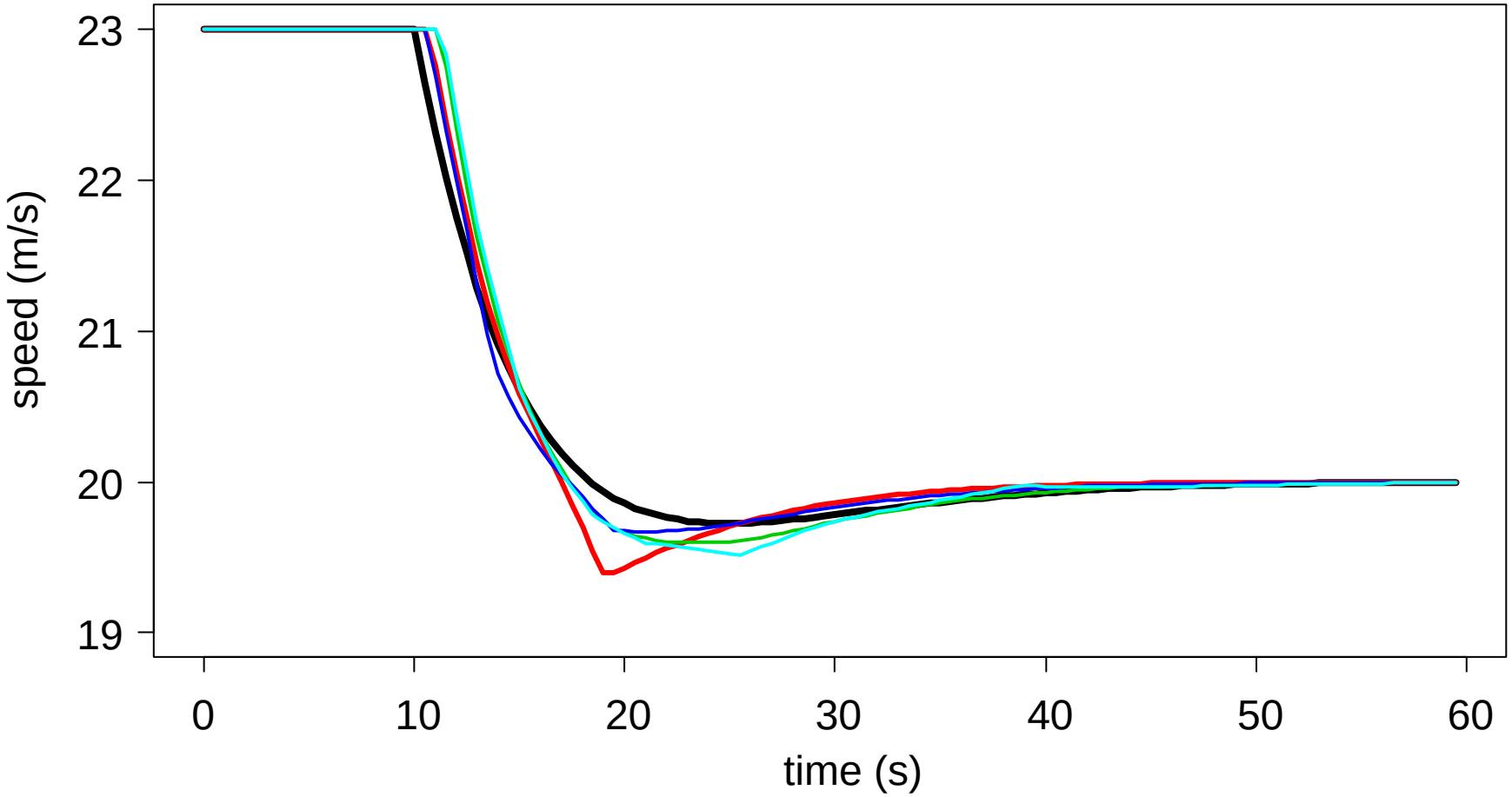


Depends: are AP's deterministic or random?

- VISSIM (?): each driver has own set of thresholds that trigger the AP's
- While the thresholds of all drivers follow some distribution, the thresholds of one driver is fixed: this is deterministic!

- I think: the driver decides randomly, in each moment she (unconsciously) asks: make a change, or not
- Of course, coming too close dramatically increases the likelihood for change, but it remains a likelihood
- My opinion.

Stochastic AP‘s



Does it make a difference?

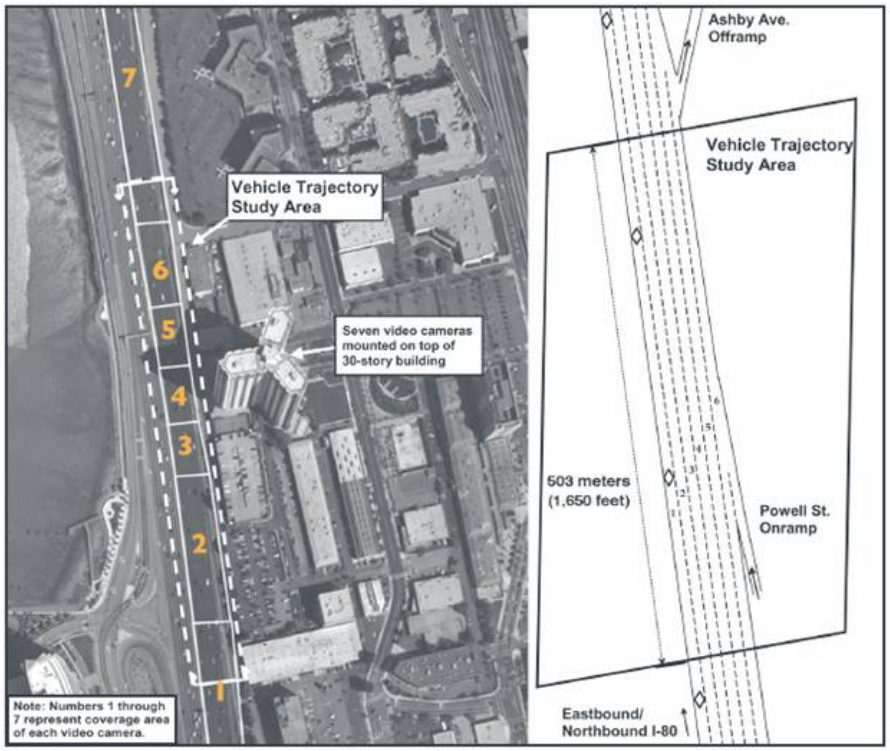
- I had done some work on this [1], but in the light of my recent involvement with the modelling of autonomous vehicles, this has to be over-hauled.

[1] PW, European Physical Journal B, 84, 713-718 (2011)

A different story: boundary conditions

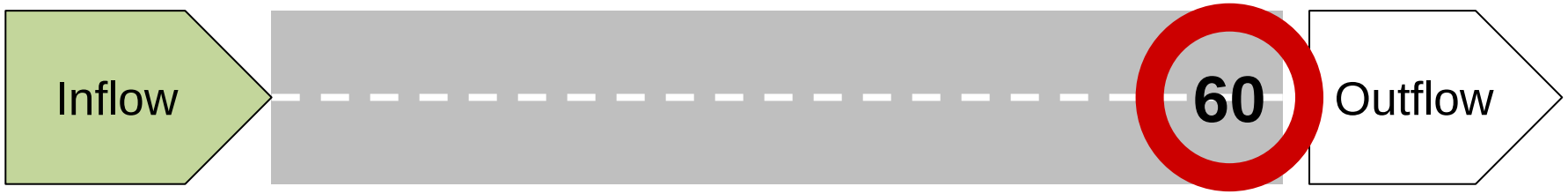
NGSIM = Next Generation Simulation Program.

- I remember a researcher dealing with these data, and a μ-Sim tool named AIMSUN
- Whatever he tried, he could not reproduce the congestion pattern
- Why?
- Because the congestion was not produced within the study area, it was imported from downstream (jams run backwards, do they?)



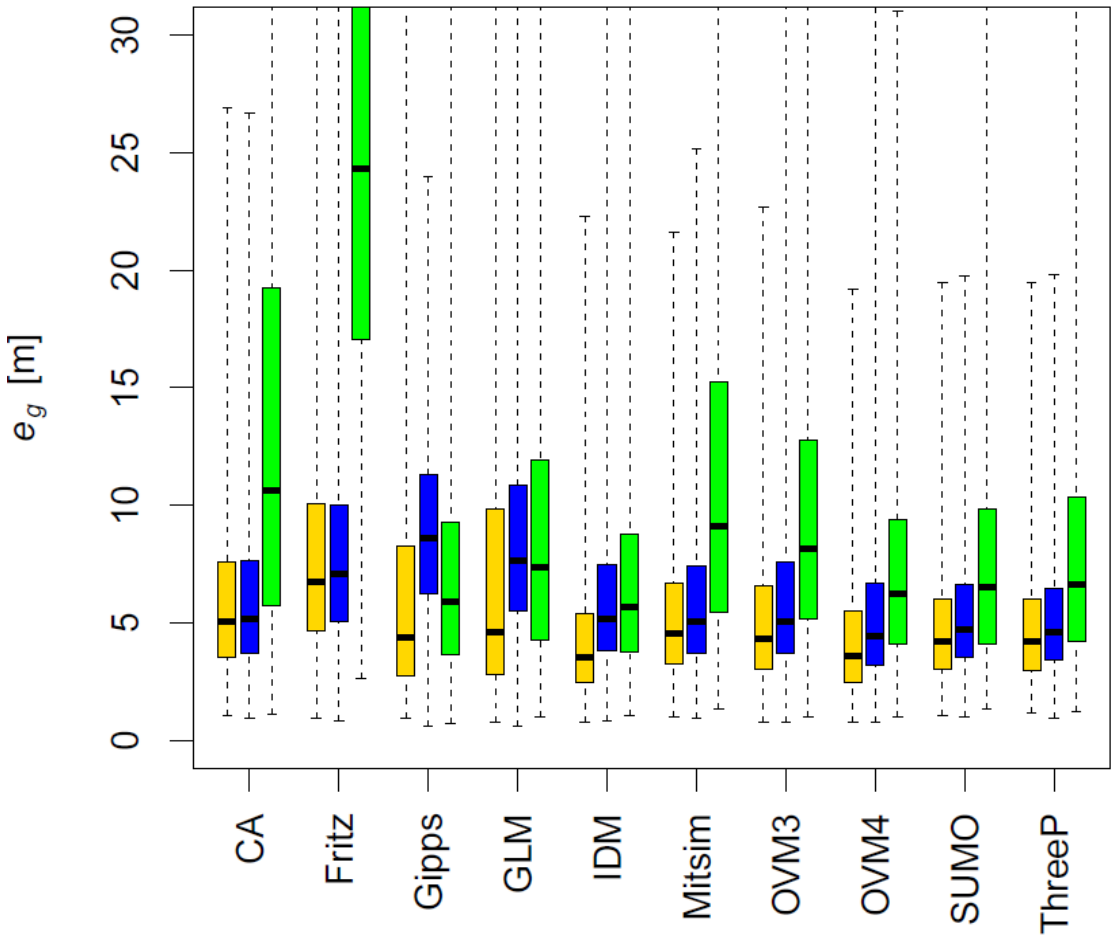
Making it correct

- Clearly, AIMSUN was driven with the upstream demand
- (which is not that easy if you look at the zoo of insertion algorithms available in SUMO)
- However, it has to be driven with the downstream condition as well!
- This can be either the measured flow itself (in a jam, it is reduced), or, much better, the measured speeds



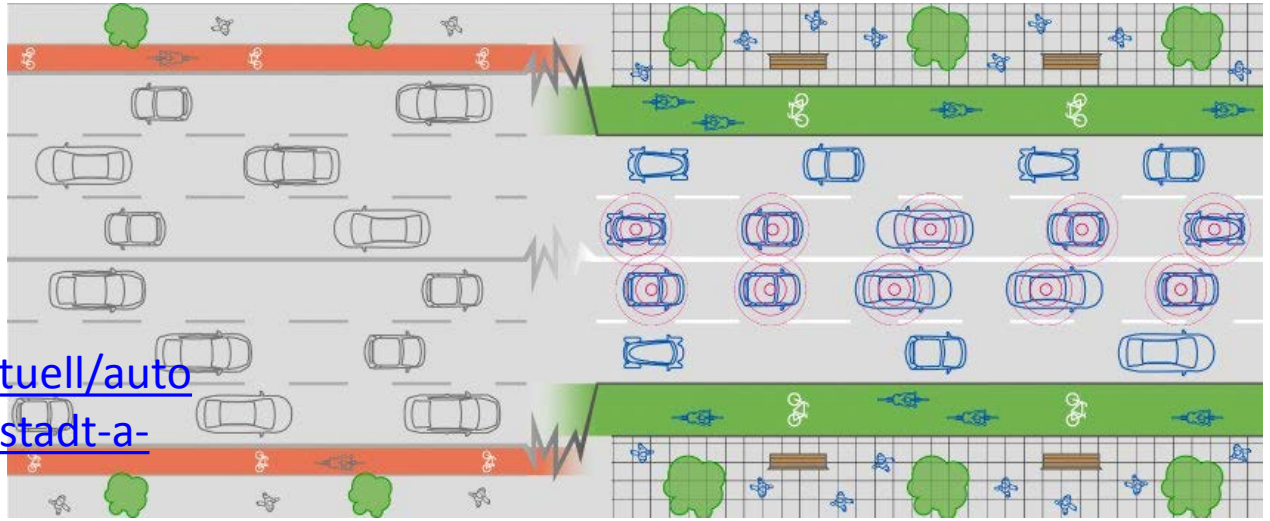
After anything else…

- It comes down to the models.
- Do we have the right models?
- My feeling: no, there is room for improvement
- And we need this urgently, because…



Getting better is urgently needed…

- As we enter the new world of autonomous driving!
- And especially, if we have a mix of AV (autonomous vehicles) and HV (human-driven vehicles)
- At least in the beginning, they max exchange control!
- Consider vehicles driving AV with short headways, and then giving control back to HV.
- What is this?
- A bottleneck!



http://www.spiegel.de/auto/aktuell/autonomes-fahren-chance-fuer-die-stadt-a-997393.html

High-level approaches

Once there was MULTITUDE…

- A European COST action, where researchers involved reviewed and summarized the use of guidelines to make reliable traffic simulations
- It collects these efforts into a final report named "***Traffic Simulation: Case for guidelines***"
- http://publications.jrc.ec.europa.eu/repository/handle/111111111/30680
- It is from 2013, and it has nothing lost yet of its importance
- Of course, those guidelines are everywhere (see Case for Guidelines), but are they used?
- And: in the best of all worlds, they should be constantly updated

Other things I have stumbled about

- The tale of a German city: it is rumored, that some consultant used the seed of the random number generator to fit the model (I owe this story to Markus Friedrich and Peter Vortisch)
- I forget: it happened after the end of MULTITUDE. Last year.

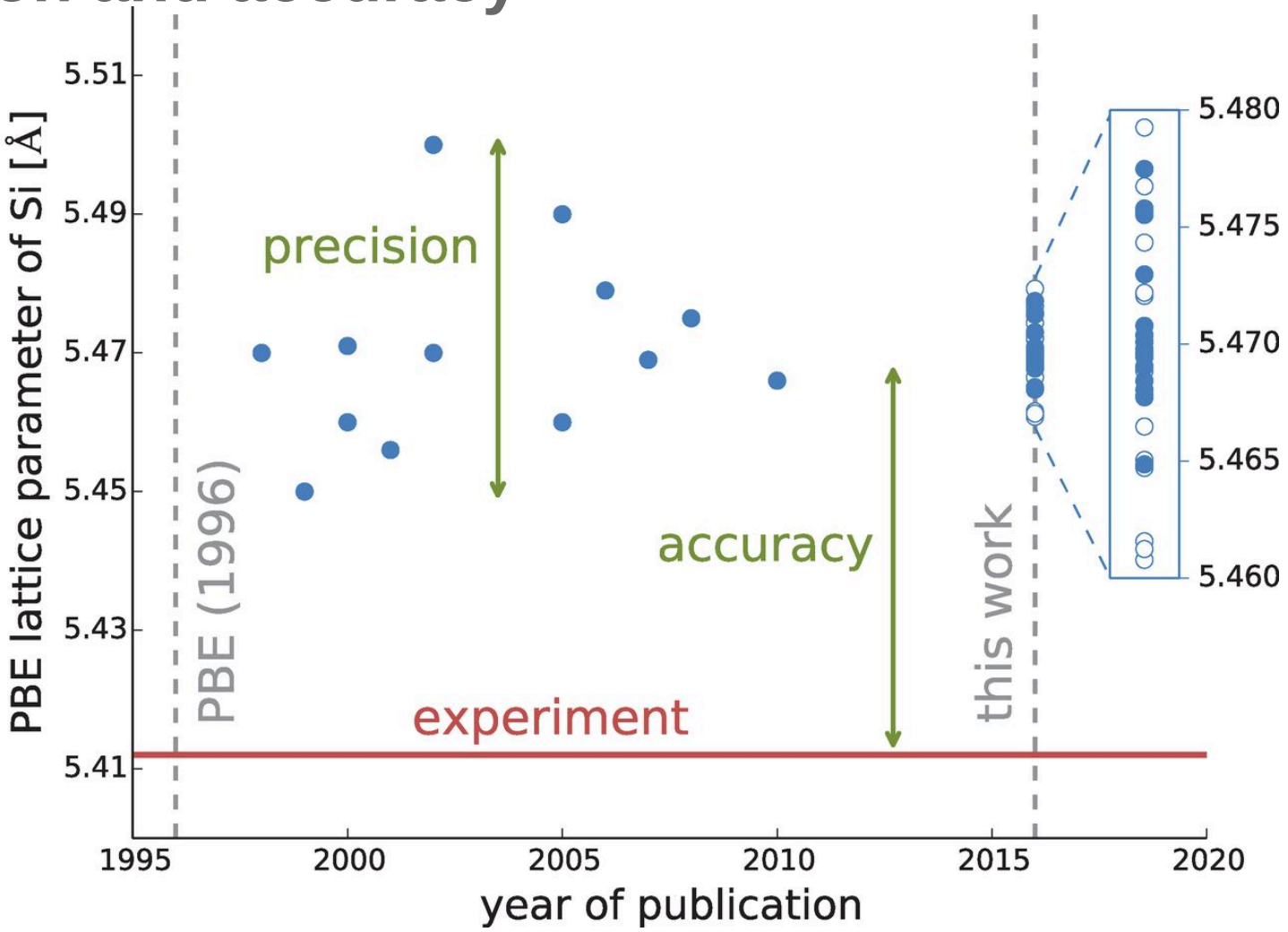
- And: I have seen funny things by my own: a simulation of an intersection, that did not include the traffic signals upstream
- I corrected the people, of course. But how often do you have something that could not recognized that easily?
- And: how blind am I myself? What is it that I overlook?

More general I: Testing

From another planet (or world?)

- Solid state physics
- Compute the features of a certain substance before you actually brew it
- Physicists have worked on this since decades, the method to do so is called DFT (density functional theory)
- Many different codes exist, like the μ-Sim models in traffic
- They are not bad, but: Science **351**, p 1394 (2016) and on pp. 1415ff, http://dx.doi.org/10.1126/science.aad3000
- Benchmark for materials simulation is needed, and they have it
- They use three KPI (key performance indicators): Consistency, precision, and accuracy

Precision and accuracy



Kurt Lejaeghere et al. Science 2016;351:aad3000

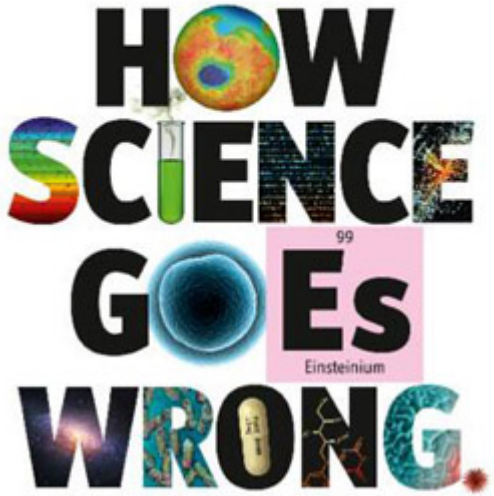
Published by AAAS

So, for Silicone…

- Achieved 5.47 Å; reality is at 5.41 Å, an error of 1.1%
- Of course: some very specialized branches of physics do much better than this (check out the g-factor of an electron)

- Remember: for traffic models, we typically reach:
  - 5% for the speeds (they are particularly simple to predict)
  - 10% for the distances
  - 85 / 15 rule for networks
- We have little idea of consistency (e.g., traditional planning vs mesoscopic vs μ-Sim models)
- The accuracy is tackled, but much harder than in the DFT tale above: we do not have one silicon, but one billion drivers

More general II: Recent Science issues



DLR

Reproducibility

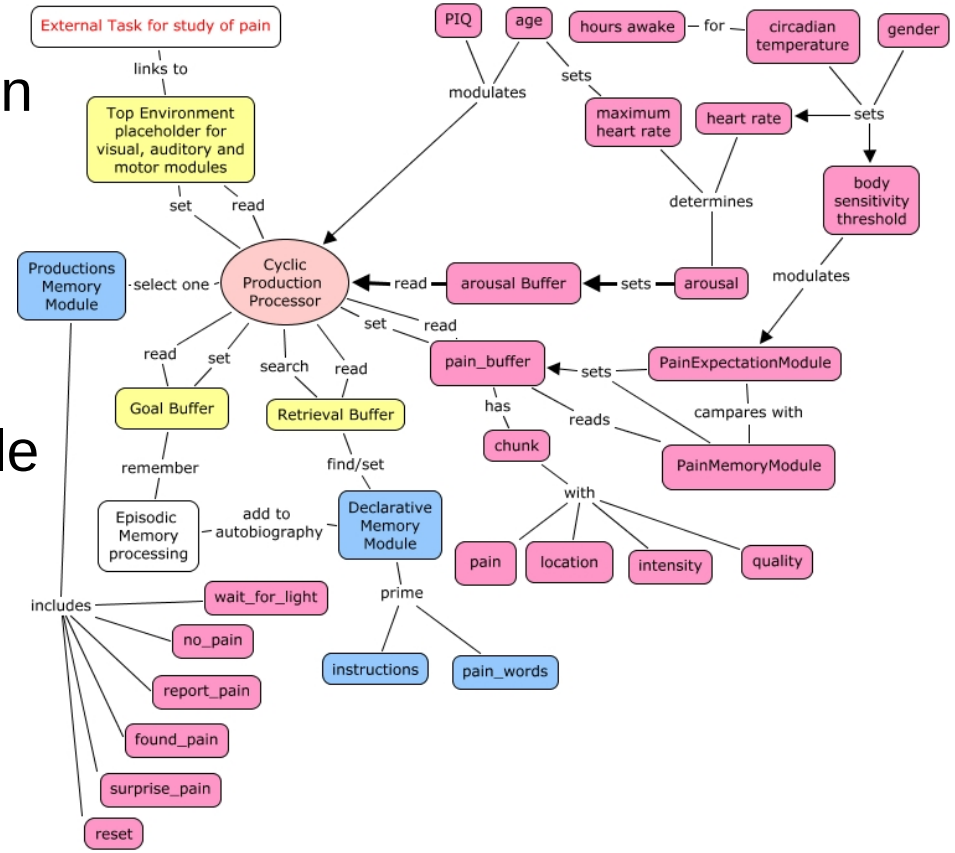
- Recently, people have raised concerns about reproducibility of scientific results
- Clearly, reproducibility is at the heart of this endeavor
- It has a certain focus right now on the social sciences, but in the light of recent misdoings, even engineering may be fallible to it.
- Consider results with your favorite μ-Sim tool
- A few versions later, do you still get the same results for the very same scenario?
- If you believe “yes“, have you actually tested it?
- There are some developments to conserve the old stuff, but this is a little helpless, isn‘t it? (Conserving old code etc.)

Reproducibility II

- SUMO‘s 11 models have been changed countless times in how they have been coded
- Consider also something truly complicated like the models of
  - Wiedemann
  - Kerner
  - MITSIMLab model (not in SUMO)
- Are they really implemented correctly? What does correct mean?
- PTV once had to code a complicated lane-changing model into VISSIM, within the NGSIM project.
- Fortunately, they had the description and the code.
- As Peter Vortisch puts it in a memorable speech: “If there were differences between the text and the code, we believed the code.“

Truly complex models: ACT-R

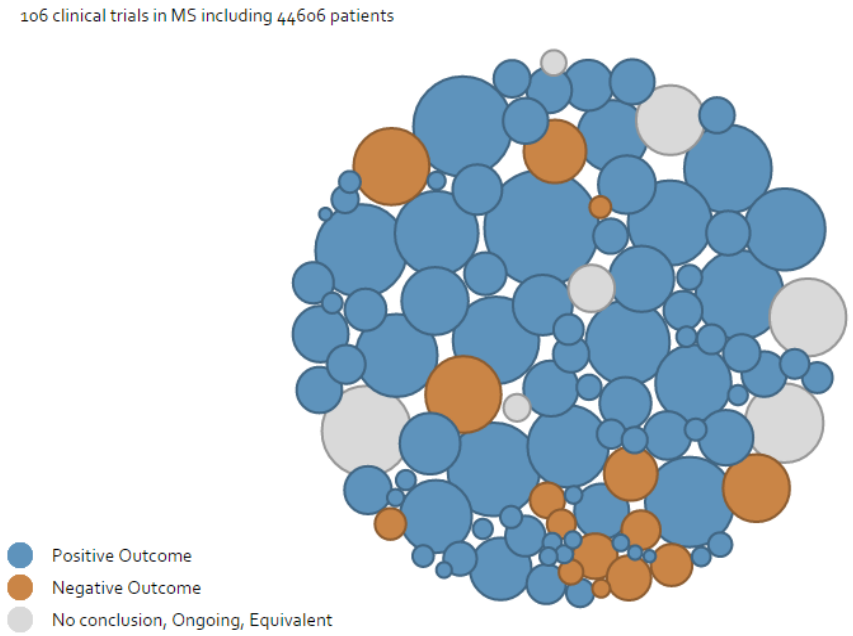
- The psychologists in my department are fans of ACT-R
- Against this complexity, which tries to simulate human decision processes, the most complicated simulation model pale
- The same holds true for a real model of an autonomous vehicle
- But we have to deal with that, albeit the physicist in me begs for simplifications…



Sterling, 1959

The file-drawer effect (publication bias)

- You came up with a cool idea how to better control a traffic light.
- Then you put your favorite μ-Sim software to work…
- But your method fails, completely
- Will you publish it?
- Certainly not, and this is bad: we all learn from failures.
- Estimates tell that >75% of all papers report positive results
- How likely is this?



http://www.alltrials.net/news/visualising-publication-bias-in-ms-trials/

Significance (Statistical Crisis in Science)

- In a different scenario, it seems that your approach is slightly better than what you had compared it against
- A statistical test (very rare for traffic engineers, but try, it is cool), states: difference is not significant
- (Which, but standard interpretation, means that it fails at the 5% level of significance.)
- Hope: falls short since it over-estimates quality of the test
- Repeat this a couple of times, if it fails in any case ➔ it is bad
- American Statistical Association recently strongly called for more care

American Scientist,
Illustration by Tom Dunne.

Coming back: The final story to be told…

- Happened to us recently: Two new traffic control methods, both did simulation contest very well, outperforming existing solution
- (SUMO & VISSIM agreed on that!)
- Then we went to the field (an epic endeavor, believe me)
- And: the results were to good to be true (>20% improvement),
- Very different from our μ-Sim results
- Not final: existing method was flawed due to a wrongly configured detector
- We have corrected this, of course; scheduled a repetition of the field test.

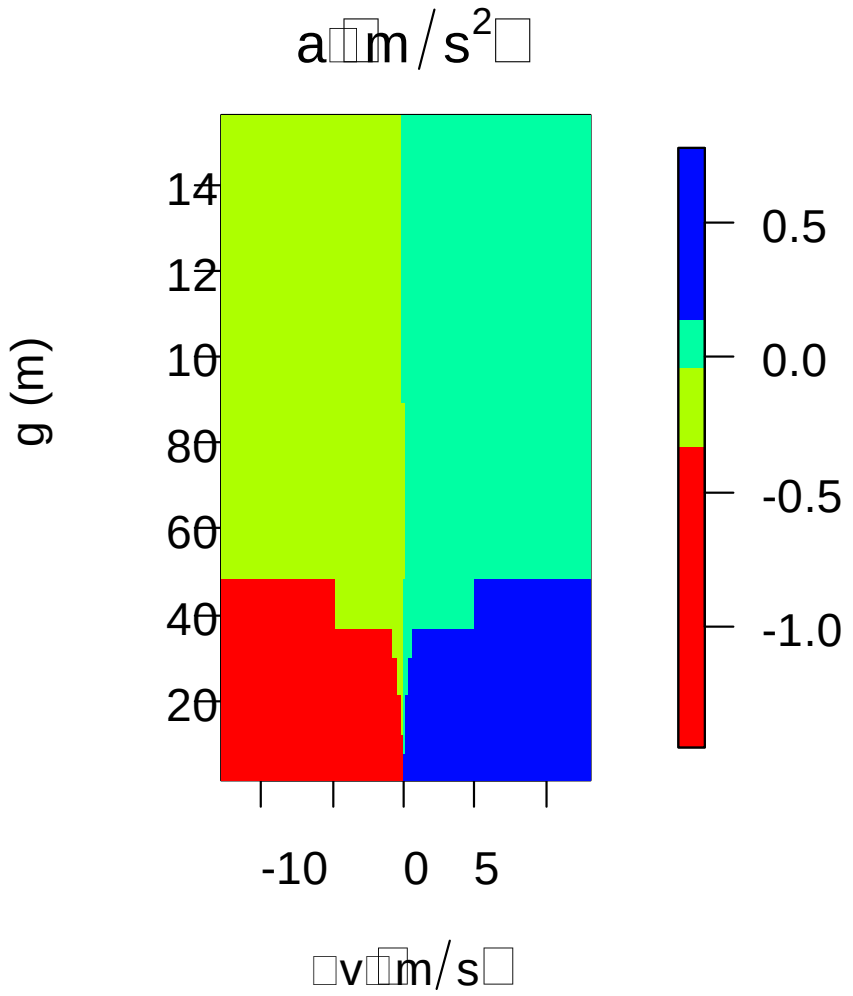
Openstreetmap.org

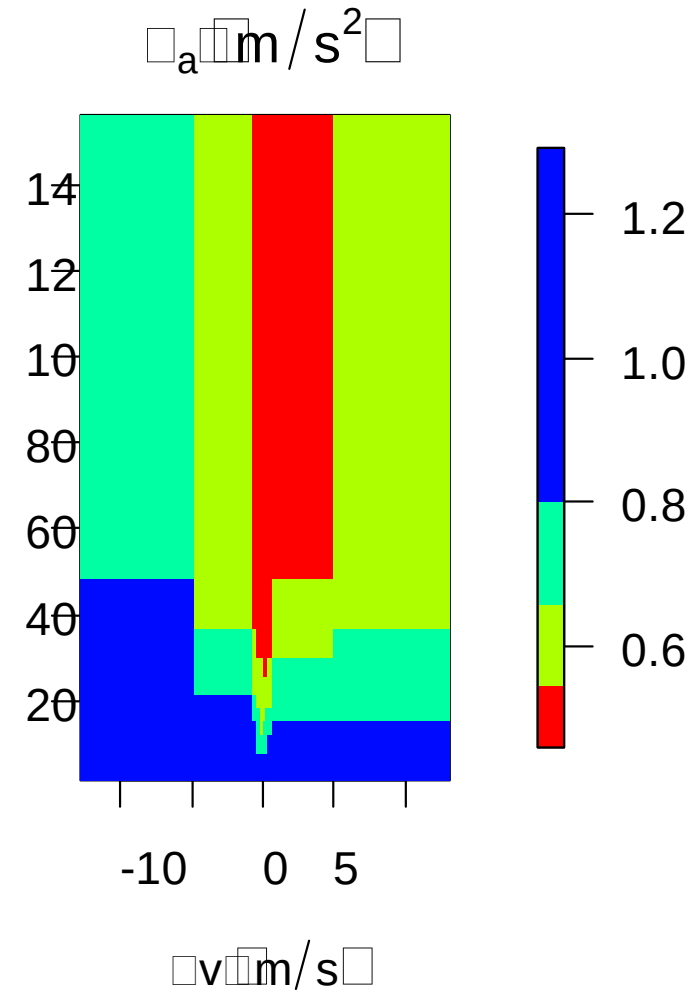
Conclusions

Conclusions: Hope you had fun!

- Was a grand tour, with a fairly large distance to traffic at times.
- The keynote format allowed for this, I trust
- Anyway, four things I have covered are at the core of next years research and development, especially in traffic:
  - Software,
  - Users,
  - Models,
  - Reproducibility.
- And we need to bring this to the young next folk in this area, as well as in other areas.

Thanks for your attention & patience!

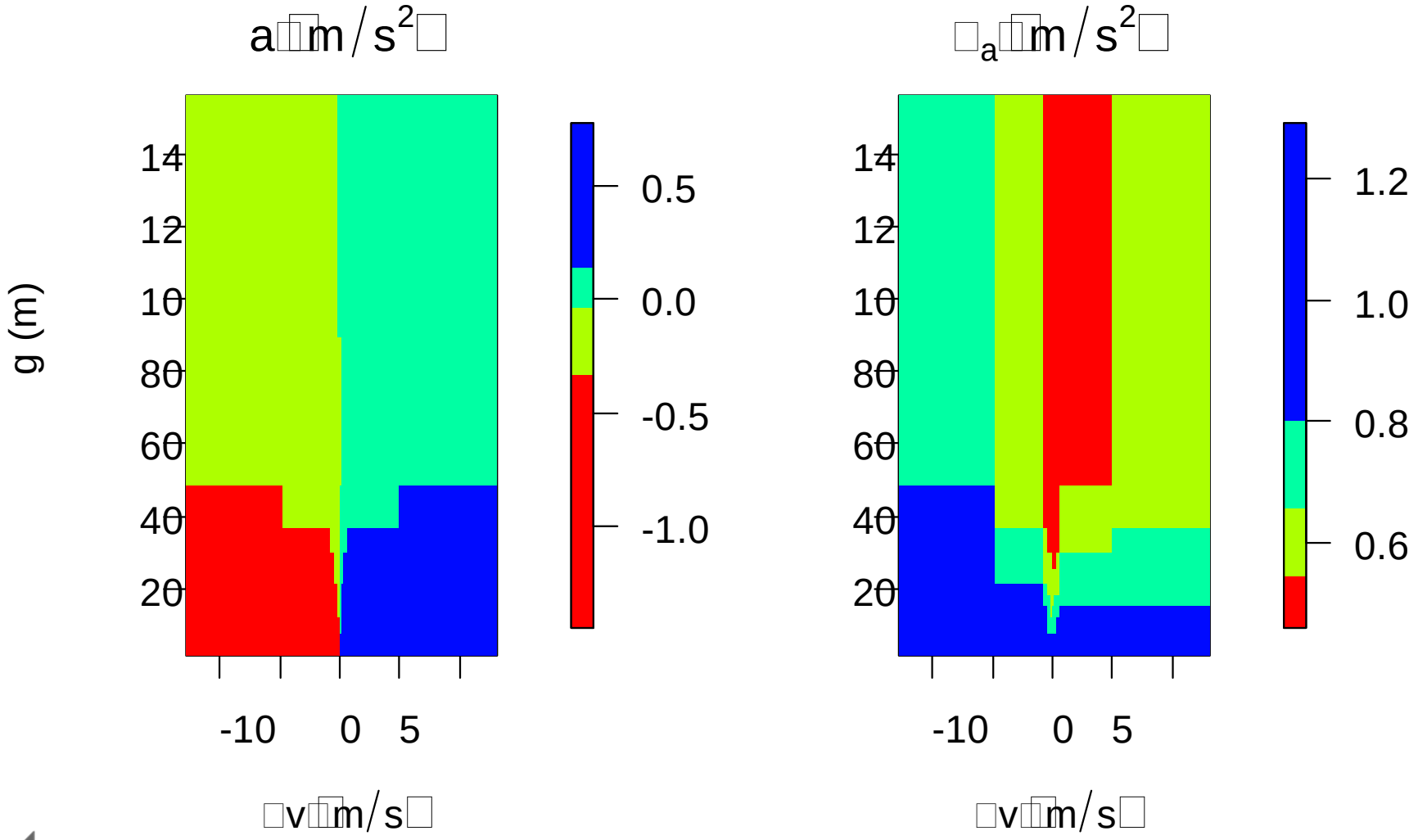




Thanks for listening.

Additional stuff

Stochastic or not, that is the question



All Software is buggy, but some works

- An old joke: If the automobile industry had developed like the software industry we would all be driving $25 cars that get 1,000 miles to the gallon.
  Yeah, and if cars were like software, they would crash twice a day for no reason, and when you called for service, they'd tell you to reinstall the engine.
- Watts S. Humphrey: "Software's simply terrible today. And it's getting worse all the time."
- Humphrey's definition of good software is:
- usable, reliable, defect free, cost effective and maintainable
- McCarthy "Most Software Sucks."
- But consider Myhrvold (CTO @ Microsoft): "Software sucks because users demand it to."
- Henry Petroski *The Evolution of Useful Things*, 1992, "form follows failure."

https://www.technologyreview.com/s/401594/why-software-is-so-bad/

A short list

- software defects have
  - wrecked a European satellite launch,
  - delayed the opening of the hugely expensive Denver airport for a year,
  - destroyed a NASA Mars mission,
  - killed four marines in a helicopter crash,
  - induced a U.S. Navy ship to destroy a civilian airliner,
  - and shut down ambulance systems in London, leading to as many as 30 deaths.

Towards Reliability in the Usage of Traffic Simulation Tools

When using simulation programs to support decisions, it is of the uttermost importance to make sure that the results are as reliable and realistic as possible. This turns out to be one of the bigger challenges with the development and the use of those programs, be it open source or not. With respect to traffic simulations, this means to tackle a wide range of modelling questions from driving a vehicle to route and mode choice decisions or even decisions regarding to move at all (demand modelling). While this being a more or less scientific and engineering job to do, in addition the model and simulation developers have to help the applicants in the daily use of these simulation programs to avoid wrong usage resulting in bad decisions. This keynote will give some examples which show, that there are still a lot of hurdles to surmount until we may reach a state where these programs can be considered ripe.

Random numbers

- Telling this tale? Not sure.

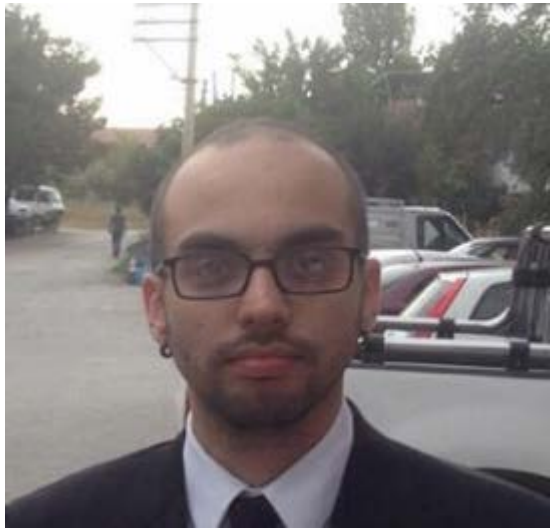
ODE‘s, summing up

- A quote: “to solve ODE‘s, you ALWAYS have to resort to discrete time dynamics” ➔ a computer can never actually solve an ODE

- Is it true or not?

- Well, a little bit. Second part true, but not due to discrete time
- Discrete time is just one trick to solve ODE‘s
- You can do it instead by using a polynomial and it’s coefficients to evolve your state from t to $t + \Delta t$
- If you are ever in need of this, then you may remember this keynote and say: “let us switch to a dense integration scheme“
- (Of course, implemented in boost, too.)

EWGT Stuff



Electronic Ticket Receipt	Booking ref:	68UTJ3		Check My Trip	
	Issue date:	27 JULY 16		Baggage	
	Airline booking ref:	TK/SUYYMY			
	Issuing Airline:	TURKISH AIRLINES			
	Ticket:	235-2282841207			
	Issued In Exchanged Ticket Number:	235-1796726494			
	Original Ticket Number:	235-1796726494			

Itinerary														
From	To		Flight	Class	Date	Departure	Arrival	Resa (1)	NVB(2)	NVA(3)	Last check-in	Baggage (4)	Seat	
BERLIN TEGEL	ISTANBUL		TK1726	T	04Sep	14:45	18:40	Ok	04Sep	04Sep		30K		
	Terminal I						Fare Basis				THY2XP			
Operated by			TURKISH AIRLINES				Marketed by				TURKISH AIRLINES			
											Duration	02:55 (Non Stop)		
-														
ISTANBUL	GRAZ		TK1459	T	07Sep	09:20	10:35	Ok	07Sep	07Sep		30K		
Terminal I							Fare Basis				THT2PC			
Operated by			TURKISH AIRLINES				Marketed by				TURKISH AIRLINES			
											Duration	02:15 (Non Stop)