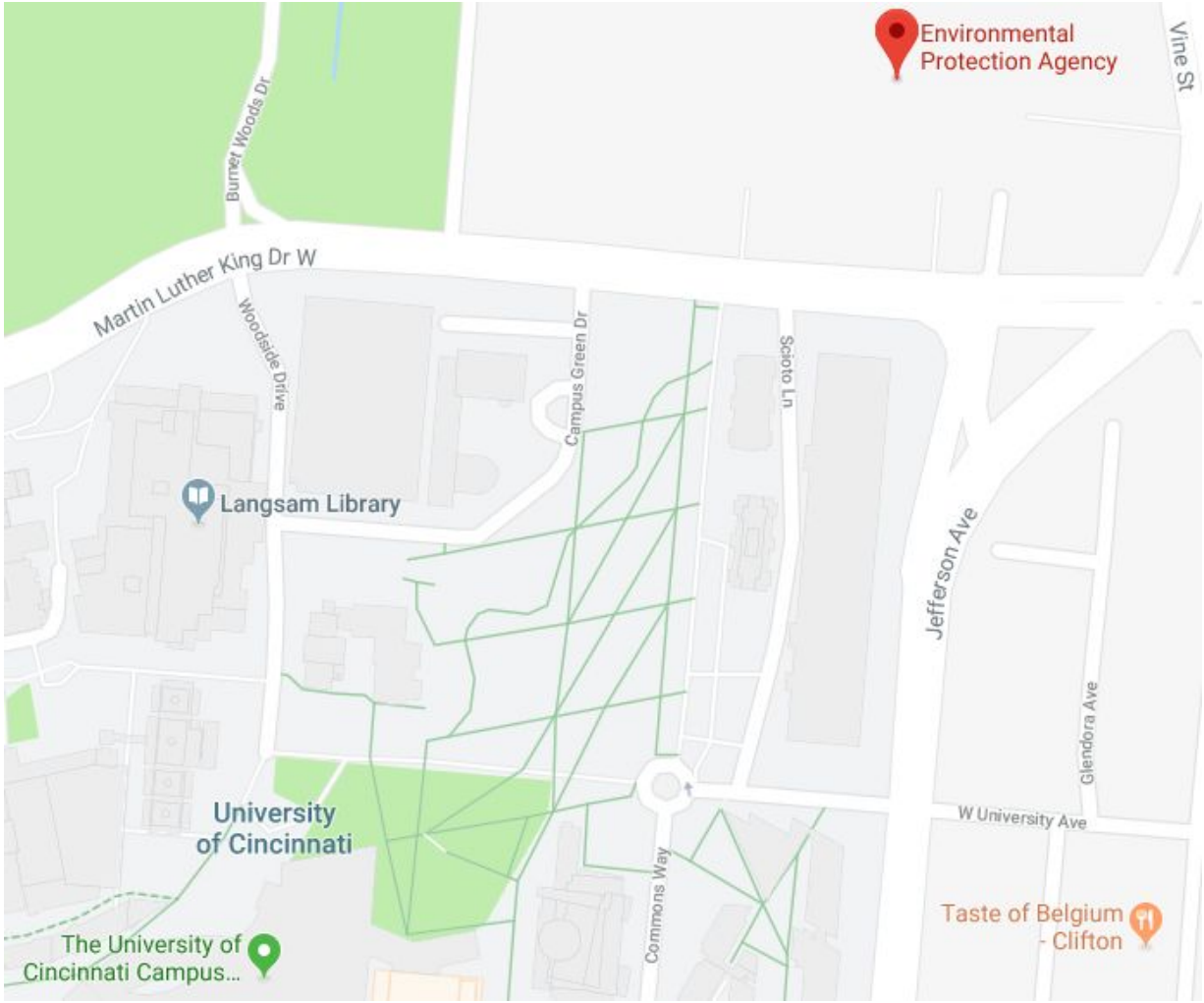
Hierarchical Metamorphic Relations for Testing Scientific Software

Xuanyi Lin[1,2], Michelle Simon[2], and Nan Niu[1] (nan.niu@uc.edu)
[1]Department of EECS, University of Cincinnati, USA
[2]EPA Office of R&D, Water Systems Division, Cincinnati

SE4Science Workshop, June 2, 2018



4,205 mi

Distance from Cincinnati to Gothenburg

6767.292

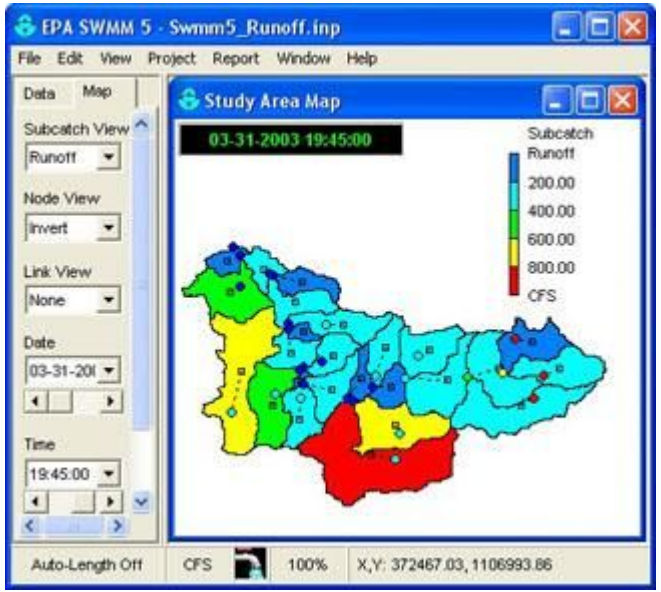
Kilometer

UNITED STATES ENVIRONMENTAL PROTECTION AGENCY

OAK RIDGE INSTITUTE FOR SCIENCE AND EDUCATION
Managed by ORAU for DOE

UNIVERSITY OF Cincinnati

Storm Water Management Model (SWMM)



scientific model developer

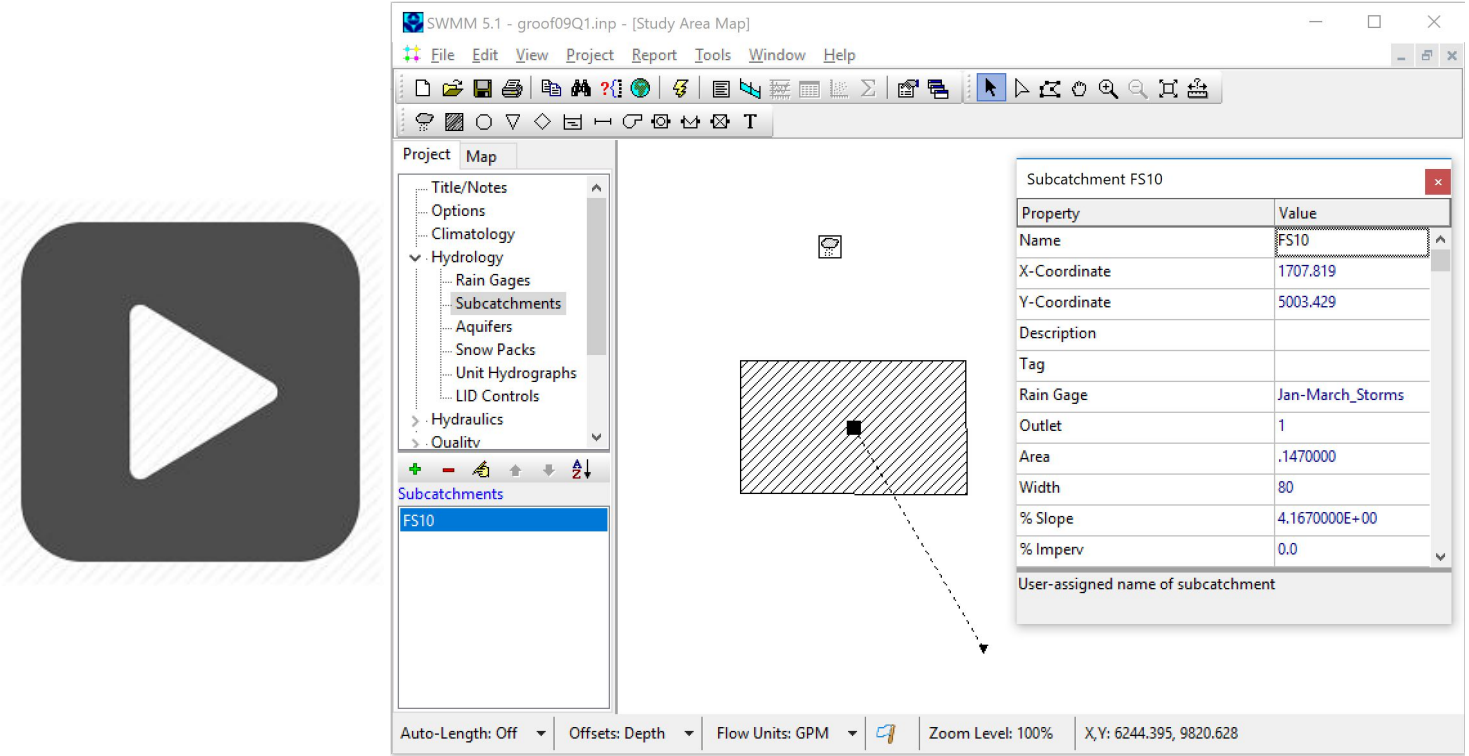
users (e.g.,National Stormwater Calculator)

United States Environmental Protection Agency

… and many other stakeholders

https://www.epa.gov/water-research/storm-water-management-model-swmm

SWMM tracks the quantity and quality of runoff made within each sub-catchment.

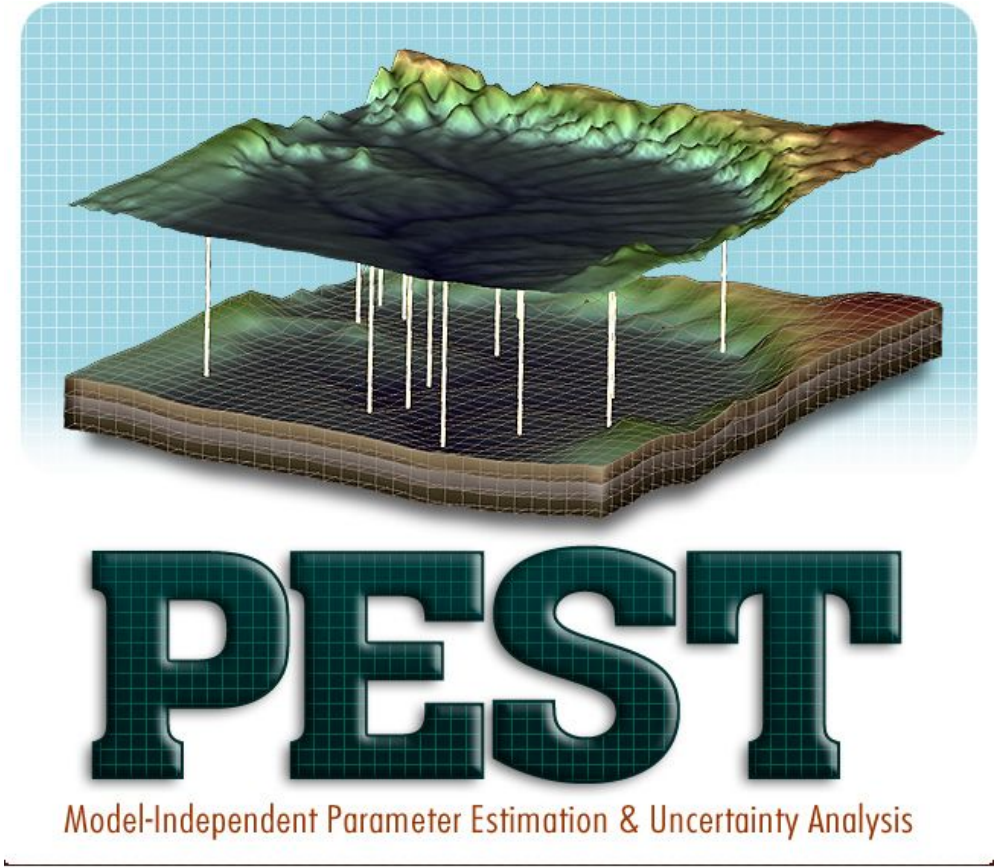


Characteristics of SWMM

- Development began in 1969, released in 1971
- Version 5.1.012 was released in 2017
- Computational engine is written in C & the UI in Delphi.XE2
- C code size is about 45,500 LoC

PEST++

- First version of PEST was released in 1994
- Version 13.3 was released in 2014
- C++ code size is about 180,000 LoC

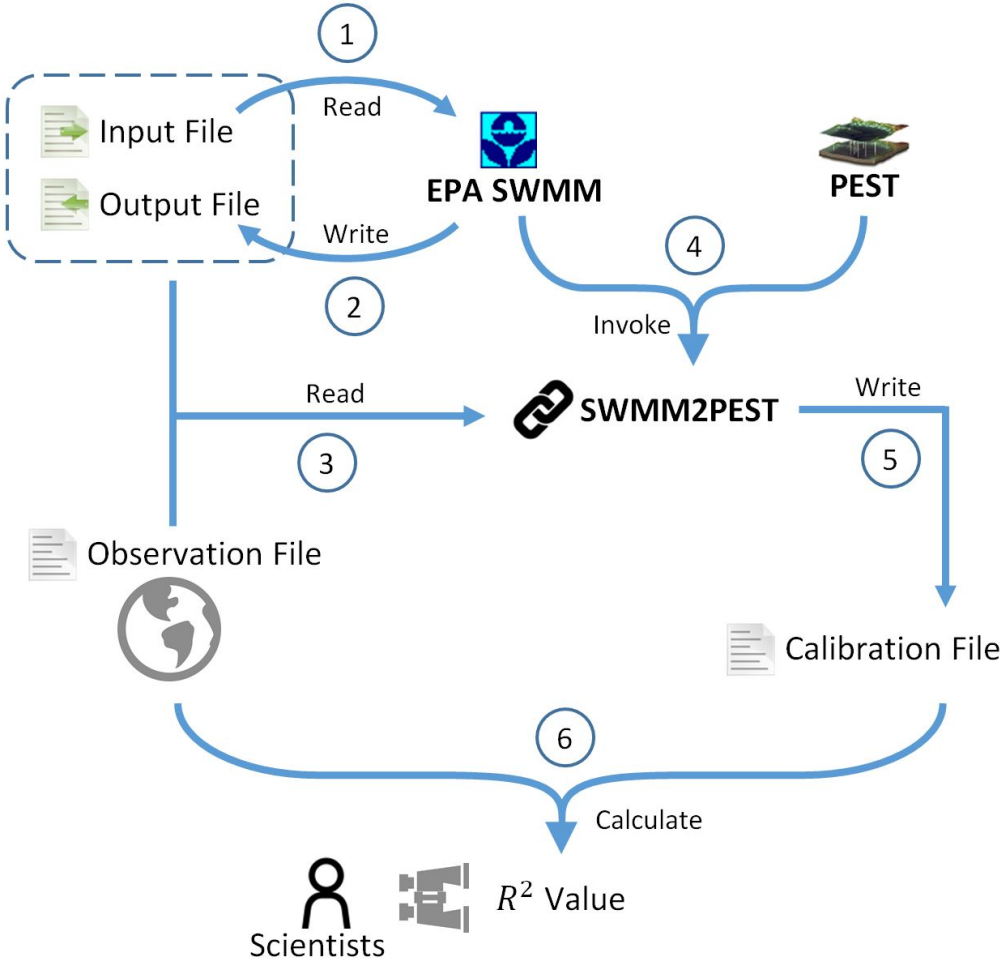


SWMM2PEST

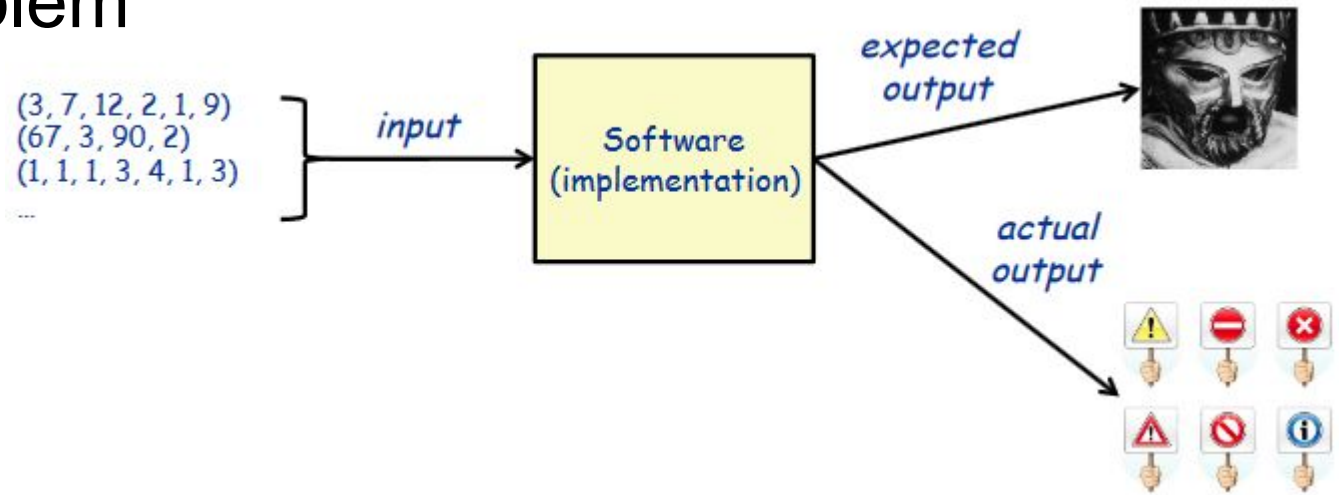
- Python
- 2,500 LoC

SE4Science17

“A Novel Coupling Pattern in Computational Science and Engineering Software”



Oracle Problem



R^2

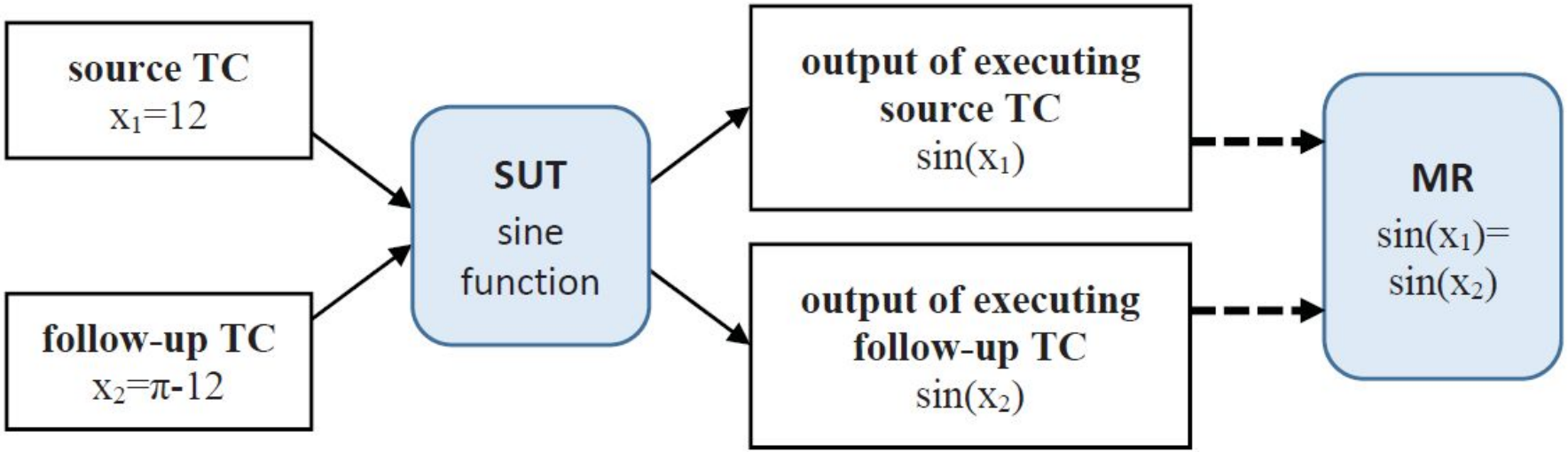
search engines

sine(x), *cosine(x)*, *log(x)*, *sqrt(x)*, ...

develop software to find answers which are previously unknown

...

Metamorphic Testing



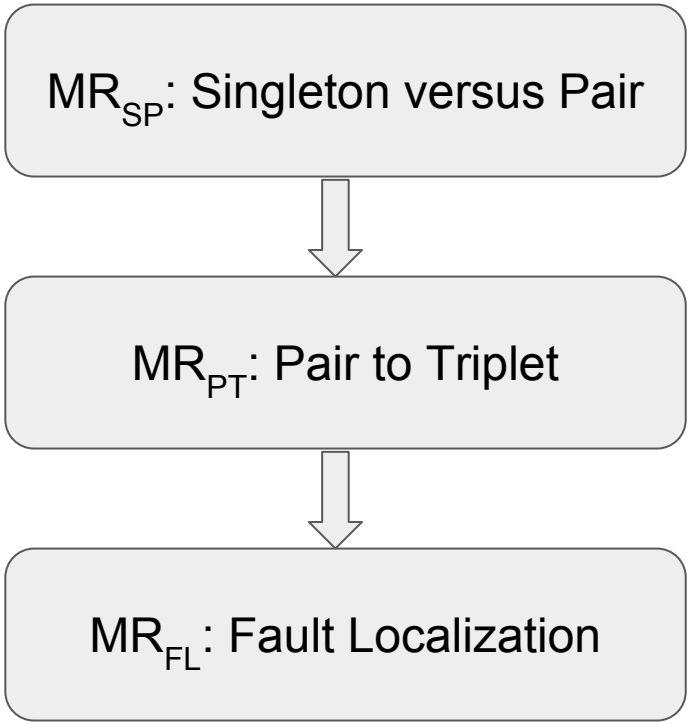
Metamorphic Relations

In each paper, 5 MRs are defined at one-shot, organized at the same level, and executed & analyzed separately:

- J. Ding *et al.* “An application of metamorphic testing for testing scientific software”, In *International Workshop on Metamorphic Testing*, pages 37-43, May 2016.
- Z. Zhou *et al.* “Metamorphic testing for software quality assessment: a study of search engines”, *IEEE Transactions on Software Engineering*, 42(3): 260-280, March 2016.

Hierarchical Metamorphic Relations

We define and organize MRs in a hierarchical manner where one MR's execution & analysis will influence the follow-up MRs



MR_{SP}: Singleton versus Pair

Source TC: calibrate *Pi* & calibrate *Pj*

Follow-up TC: calibrate *Pi* ^ *Pj*

ID	R_i^2 ? $R_{i \wedge j}^2$	R_j^2 ? $R_{i \wedge j}^2$
SP_1	<	<
SP_2	<	=
SP_3	<	>
SP_4	=	<
SP_5	=	=
SP_6	=	>
SP_7	>	<
SP_8	>	=
SP_9	>	>

MR_{SP}: Singleton versus Pair

Source TC: calibrate *Pi* & calibrate *Pj*

Follow-up TC: calibrate *Pi* ^ *Pj*

Desired = {SP_1}

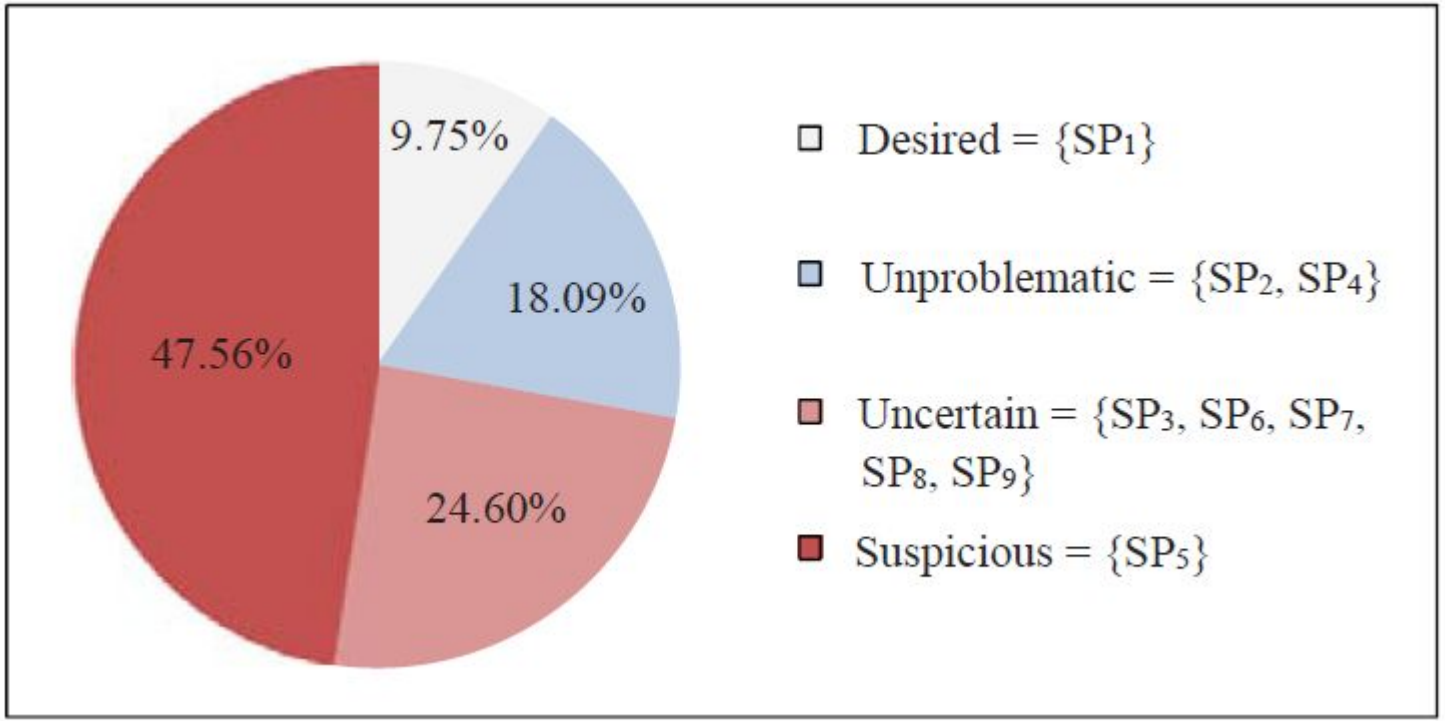
Unproblematic = {SP_2, SP_4}

Uncertain = {SP_3, SP_6, SP_7, SP_8, SP_9}

Suspicious = {SP_5}

ID	R_i^2 ? $R_{i\wedge j}^2$	R_j^2 ? $R_{i\wedge j}^2$
SP_1	<	<
SP_2	<	=
SP_3	<	>
SP_4	=	<
SP_5	=	=
SP_6	=	>
SP_7	>	<
SP_8	>	=
SP_9	>	>

MR_{SP}: Singleton versus Pair (*N*=431 pairs)



MR_{PT}: Pair to Triplet

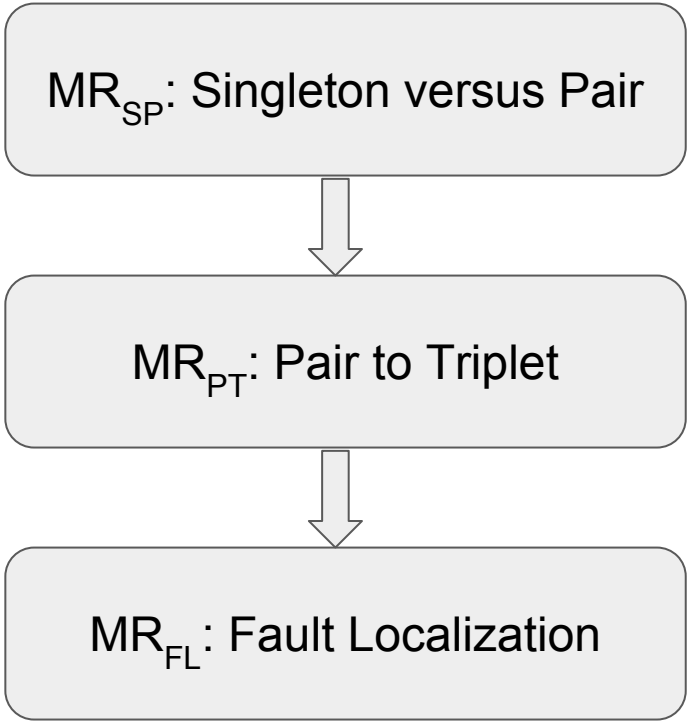
Source TC: a “suspicious” pair

Pi ^ *Pj* according to MR_{SP}

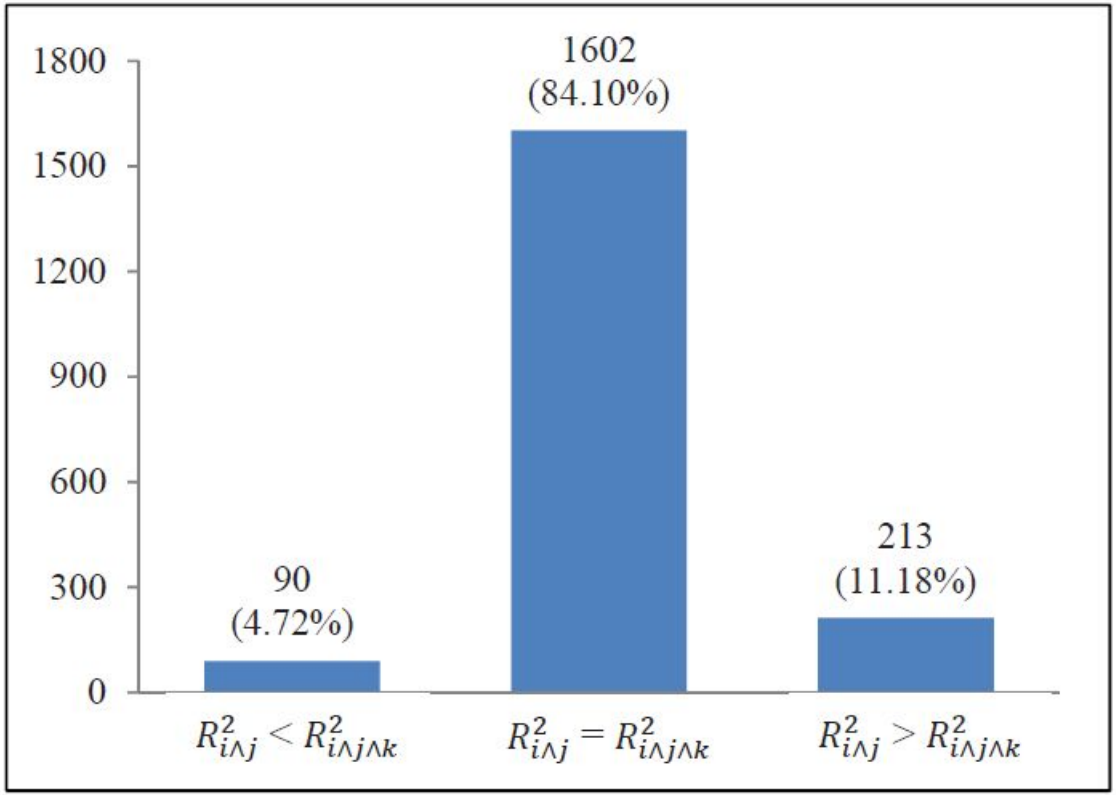
Follow-up TC: *Pi* ^ *Pj* ^ *Pk* such

that *Pk* is “desired” w.r.t MR_{SP}

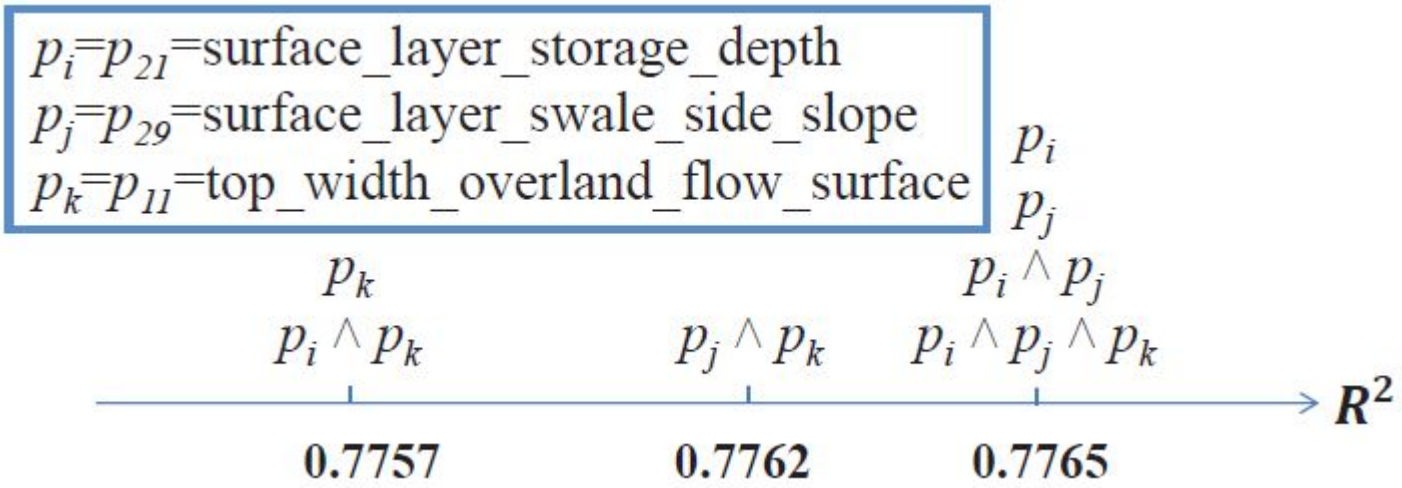
Note that MR_{SP} influences MR_{PT}



MR_{PT}: Pair to Triplet (*N*=1,905 triplets)

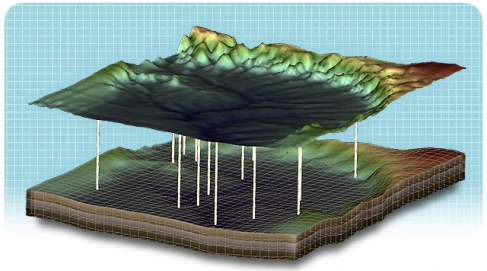


MR_{PT}: An Example



MR_{FL}: Pair to Triplet

SWMM2PEST



PEST

Model-Independent Parameter Estimation & Uncertainty Analysis

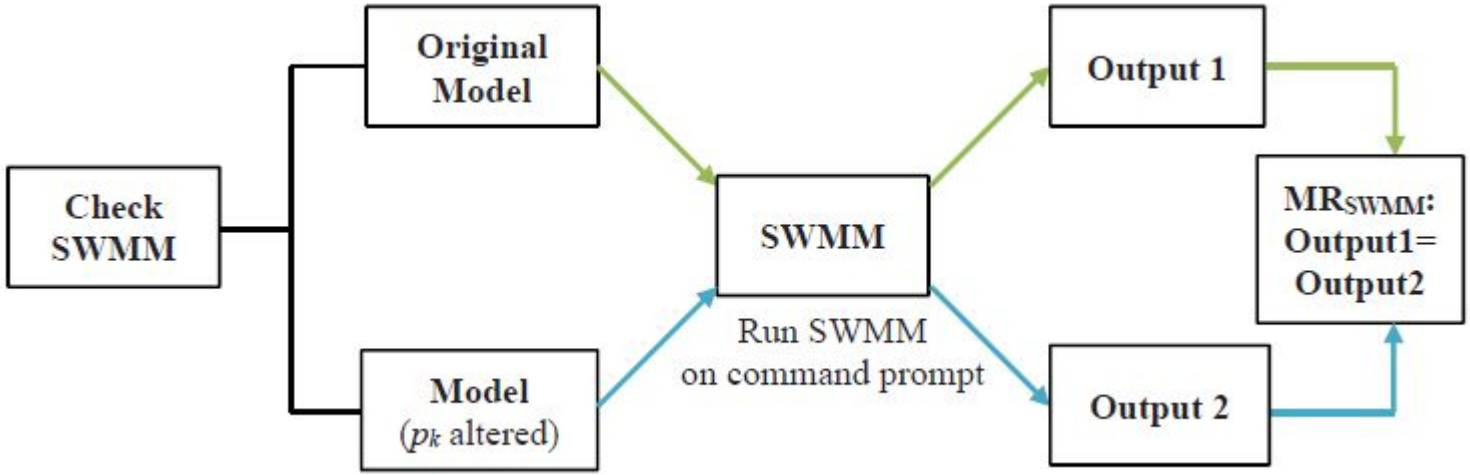
EPA SWMM

MR_{SP}: Singleton versus Pair

MR_{PT}: Pair to Triplet

MR_{FL}: Fault Localization

MR_{FL}: SWMM Fault Localization



Lessons Learned

- Metamorphic testing is effective in overcoming the oracle problem
  - Software integration
  - Speed, e.g., 50 hours in executing 1,905 triplets
- Hierarchical MRs are intended for (scientific model) developers to investigate more sensible test cases, given some (initial) testing results
  - Relations versus absolute values

Thank You

Hierarchical Metamorphic Relations for Testing Scientific Software

Backup slides

I suggest the following changes:

1. First sentence of abstract: "Scientist model developers have not yet routinely adopted systematic techniques to assure software quality "
2. Section 2.1, Second Paragraph: "Despite the differences, there appear to be some common aspects. First, the size of scientific software ranges from 1,000 to 100,000's of code [30]. Second, scientists often develop the software themselves rather than use professional code developers. (the remaining portion of the paragraph is ok.)
3. Section 2.1, Fourth Paragraph, second sentence: "testing to assure the quality of the software, as opposed to the fidelity of the code to model physical phenomena, has been unevenly practiced by scientific code developers. (the remaining portion of the paragraph is ok.)
4. Section 3. Second paragraph, first sentence. "The most current implementation of the model is version 5.1.013 which was released in 2017.

BTW - SWMM 5.1 was released in 2010. PEST++ has 180,000 lines of code.

I think that these minor changes can keep us out of hot water with scientific modeling community.