Integrated Approach to Dependable Cyber-Physical Systems:

from Category Theory to Machine Learning

Fuyuki Ishikawa, NII

with ERATO-MMSD Project / QAML Project

f-ishikawa@nii.ac.jp

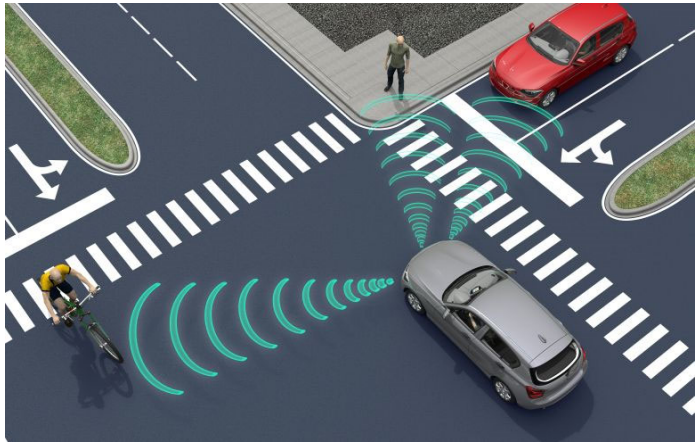
http://research.nii.ac.jp/~f-ishikawa/en/

大学共同利用機関法人 情報・システム研究機構
国立情報学研究所
National Institute of Informatics

Focus on Physics, Theory Side

CPS

Cyber-Physical Systems



- Continuous dynamics
- Differential/Difference equations
- Quantitative goals
  (distance, energy, probability, etc.)

[Lee, Cyber-Physical Systems - Are Computing Foundations Adequate?, 2006]

Scientific Challenge in CPS

- Scope beyond the classical software science
  - Hybrid models with not only discrete dynamics but also continuous dynamics for speed, energy, electricity, etc.

➡ **Heterogenization** or **quantification** of software science and formal engineering methods

  - Formal specification with probability [Morgan, ZB'05]
  - Model checking on energy consumption [Nakajima, FM'15]
  - Robustness evaluation of (un)satisfaction [Fainekos, TCS'09]
  - Theorem prover on "programs" including continuous state change by differential equations [Platzer, IJCAR'08]
  - … *How many do we have?*

Many, Many Mathematicians?

Techniques		*New Aspects*		*Heterogenized Techniques*
 T_1	+	e_1	→	$T_1(e_1)$
T_2	+	e_2	→	$T_2(e_2)$
 T_3	+	e_3	→	$T_3(e_3)$
		• • •		

Ask Meta-Mathematicians! (never me)

$T \quad + \quad e \quad \rightarrow \quad T(e)$

Enhancement

Generalization

New Instantiation
Enhancement

Techniques		*New Aspects*		*Heterogenized Techniques*
 T_1	+	e_1	→	$T_1(e_1)$
T_2	+	e_2	→	$T_2(e_2)$
 T_3	+	e_3	→	$T_3(e_3)$
		• • •		

Note: here "meta-" in a general sense,
not necessarily the classical area of "metamathematics"

Meta-Mathematics for Systems Design Project?

Group 0: Metamathematical Integration

$T \quad + \quad e \quad \rightarrow \quad T(e)$

Category Theory

led by Ichiro Hasuo (NII) (2016-2022)

Group 1: Heterogenous Formal Methods

$T_1 \quad + \quad e_1 \quad \rightarrow \quad T_1(e_1)$

$T_2 \quad + \quad e_2 \quad \rightarrow \quad T_2(e_2)$

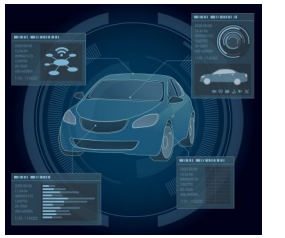
$\cdots$

Computer Science

Control Theory

Group 2: Formal Methods in Industry

Advanced setting in autonomous driving



Automotive Industry

Is this all?

Practical setting to improve present practices

Topic Example: Categological Transfer

Sound

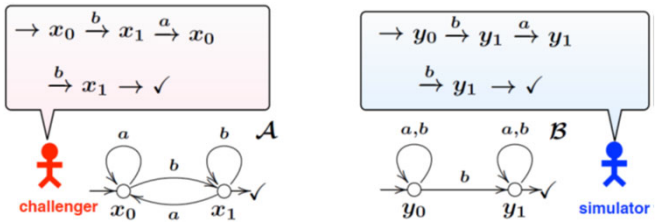
Existing Technique

$$L(\mathcal{A}) = \left\{ \begin{matrix} b, \\ bab, \\ babab, \ \ldots \end{matrix} \right\} \overset{?}{\subseteq} L(\mathcal{B}) = \left\{ \begin{matrix} b, \\ ab, \\ bb, \ \ldots \end{matrix} \right\}$$

Non-deterministic automata

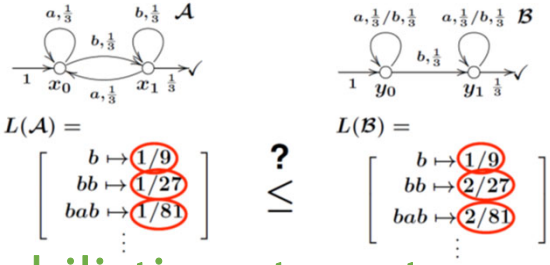
Solving the language inclusion problem by the simulation relationship

-> By a game-theoretic algorithm



Sound ??

Novel Technique



Probabilistic automata

Solving the language inclusion problem by a simulation relationship ????

-> By a game ????

Figures by N. Urabe. [Urabe, CONCUR'14 / LMCS'17]

Topic Example: Categological Transfer

Sound

Abstract Technique
T[_]

Sound

Existing Technique
$T_0 = T[e_0]$

Kleisli simulation
by Hasuo (2006)

categorically represented system

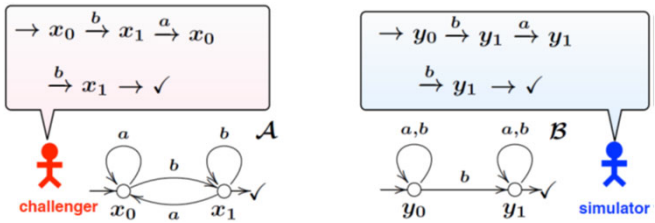
$1 \overset{s}{\nrightarrow} X \overset{c}{\nrightarrow} \overline{F}X$

What instantiation? (useful problem setting & practical solution)

$$L(\mathcal{A}) = \left\{ \begin{array}{l} b, \\ bab, \\ babab, \ \ldots \end{array} \right\} \overset{?}{\subseteq} L(\mathcal{B}) = \left\{ \begin{array}{l} b, \\ ab, \\ bb, \ \ldots \end{array} \right\}$$

Non-deterministic automata

Solving the language inclusion problem
by the simulation relationship
-> By a game-theoretic algorithm



Figures by N. Urabe. [Urabe, CONCUR'14 / LMCS'17]

Topic Example: Categological Transfer

Sound

Abstract Technique
T[_]

Sound

Existing Technique
$T_0 = T[e_0]$

Sound

Novel Technique
$T_1 = T[e_1]$

Kleisli simulation
by Hasuo (2006)

categorically represented system

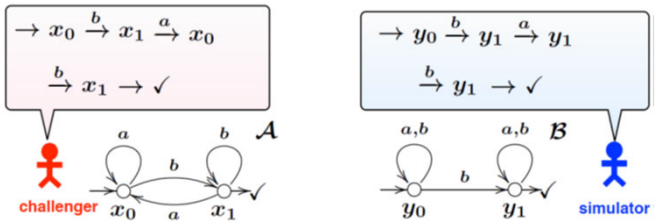
$1 \overset{s}{\nrightarrow} X \overset{c}{\nrightarrow} \overline{F}X$

$$L(\mathcal{A}) = \left\{ \begin{array}{l} b, \\ bab, \\ babab, \ \ldots \end{array} \right\} \overset{?}{\subseteq} L(\mathcal{B}) = \left\{ \begin{array}{l} b, \\ ab, \\ bb, \ \ldots \end{array} \right\}$$

$$L(\mathcal{A}) = \begin{bmatrix} b \mapsto 1/9 \\ bb \mapsto 1/27 \\ bab \mapsto 1/81 \\ \vdots \end{bmatrix} \overset{?}{\leq} L(\mathcal{B}) = \begin{bmatrix} b \mapsto 1/9 \\ bb \mapsto 2/27 \\ bab \mapsto 2/81 \\ \vdots \end{bmatrix}$$

Non-deterministic automata

Solving the language inclusion problem by the simulation relationship
-> By a game-theoretic algorithm



Probabilistic automata

Solving the language inclusion problem by a new "matrix simulation" relationship
-> Via linear programming

Figures by N. Urabe. [Urabe, CONCUR'14 / LMCS'17]

Topic Example: Categological Transfer

Sound

Abstract Technique
T[_]

Sound

Existing Technique
$T_0 = T[e_0]$

Sound

Novel Technique
$T_1 = T[e_1]$

Kleisli simulation
by Hasuo (2006)

categorically represented system

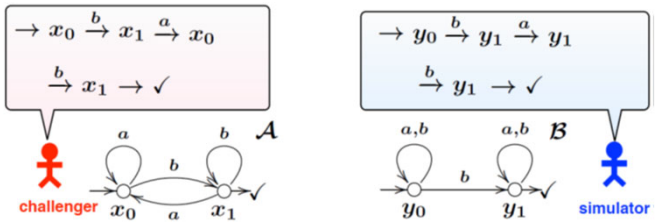
$1 \overset{s}{\nrightarrow} X \overset{c}{\nrightarrow} \overline{F}X$

$$L(\mathcal{A}) = \left\{ \begin{matrix} b, \\ bab, \\ babab, \ \ldots \end{matrix} \right\} \overset{?}{\subseteq} L(\mathcal{B}) = \left\{ \begin{matrix} b, \\ ab, \\ bb, \ \ldots \end{matrix} \right\}$$

$$L(\mathcal{A}) = \begin{bmatrix} b \mapsto 1/9 \\ bb \mapsto 1/27 \\ bab \mapsto 1/81 \\ \vdots \end{bmatrix} \overset{?}{\leq} L(\mathcal{B}) = \begin{bmatrix} b \mapsto 1/9 \\ bb \mapsto 2/27 \\ bab \mapsto 2/81 \\ \vdots \end{bmatrix}$$

Non-deterministic automata

Solving the language inclusion problem by the simulation relationship
-> By a game-theoretic algorithm



~~Probabilistic automata~~

Weighted automata in general

Solving the language inclusion problem by a new "matrix simulation" relationship
-> Via linear programming

Figures by N. Urabe. [Urabe, CONCUR'14 / LMCS'17]

It's ok if it is shown to work!

(I cannot theoretically ensure …)

Pragmatic Side

Pragmatic Approach to be Combined

- Heterogenized verification is often infeasible…
  - In a sense ensuring given properties are always met
    – undecidable or too time-consuming in many cases
  - Mathematically rigorous models often unavailable
    (e.g., Simulink models without clear semantics)

Pragmatic Approach to be Combined

- Heterogenized verification is often infeasible…
  - In a sense ensuring given properties are always met
    – undecidable or too time-consuming in many cases
  - Mathematically rigorous models often unavailable
    (e.g., Simulink models without clear semantics)

➡ Focus on testing / falsification !

" try to find faulty scenarios (via executions)"

Pragmatic Approach to be Combined

- Heterogenized verification is often infeasible…
  - In a sense ensuring given properties are always met
    – undecidable or too time-consuming in many cases
  - Mathematically rigorous models often unavailable
    (e.g., Simulink models without clear semantics)

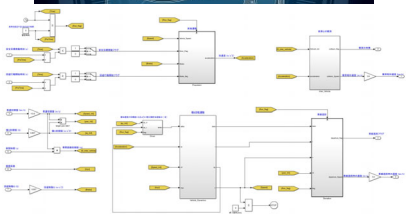
➡ Focus on testing / falsification !

" try to find faulty scenarios (via executions)"

but we need "intelligent" solutions …
(rather than purely random or exhaustive executions)

Typical Setting

Simulink model of car behavior with automated braking



Inputs

User Behavior
- Throttle signals
- Brake signals

Environment Condition
- Initial velocity
- Location and movement of pedestrian
- Road conditions
- …

Outputs

- Hit a pedestrian?
- Speed if hit

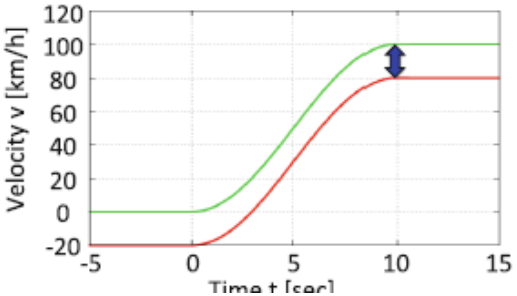
- *We have executables (simulation models or code)*
- *At least we can execute it to evaluate the output for a certain input*

➡ *Any useful problem settings & solution techniques?*

One Trend: Optimization-Driven Falsification

■Quantified properties

2 sample simulations



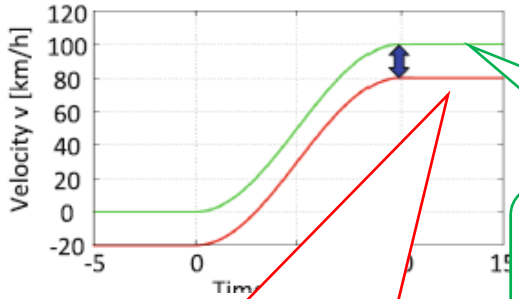
The velocity becomes ***<u>>= 80 km/h</u>***
<u>within 10 seconds</u> *after B occurs?*

[Fainekos et al., Robustness of temporal logic specifications for continuous-time signals, TheoCompSci'09]
Figure from [Akazaki et al, Time Robustness in MTL and Expressivity in Hybrid System Falsification, CAV'15]

One Trend: Optimization-Driven Falsification

■Quantified properties quantitatively evaluated

2 sample simulations



*The velocity becomes **>= 80 km/h** **within 10 seconds** after B occurs?*

YES, after 10 seconds +20km/h than required!

Robust Satisfaction

YES, after 10 seconds exactly the required value!

Fragile Satisfaction (close to Violation)

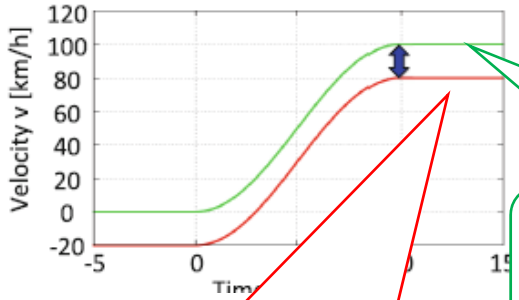
(similar evaluation can be done for time, too)

[Fainekos et al., Robustness of temporal logic specifications for continuous-time signals, TheoCompSci'09]
Figure from [Akazaki et al, Time Robustness in MTL and Expressivity in Hybrid System Falsification, CAV'15]

One Trend: Optimization-Driven Falsification

- Quantified properties quantitatively evaluated

2 sample simulations



*The velocity becomes **>= 80 km/h** **within 10 seconds** after B occurs?*

YES, after 10 seconds +20km/h than required!

Robust Satisfaction

YES, after 10 seconds exactly the required value!

Fragile Satisfaction (close to Violation)

(similar evaluation can be done for time, too)

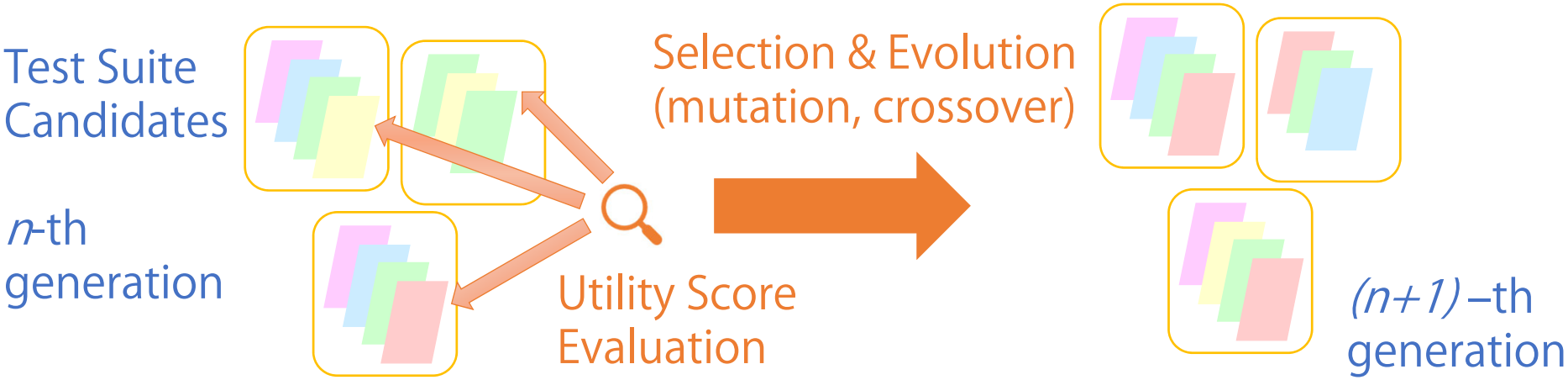
➡ *Optimization of "robustness score"*

- e.g., solved by "trials-evaluation" cycles

[Fainekos et al., Robustness of temporal logic specifications for continuous-time signals, TheoCompSci'09]
Figure from [Akazaki et al, Time Robustness in MTL and Expressivity in Hybrid System Falsification, CAV'15]

Another Trend: Search-based Testing

- Use of metaheuristic for test generation
  - Again, "trials-evaluation" cycles
  - Used also for generating a "good test list/suite/vector"



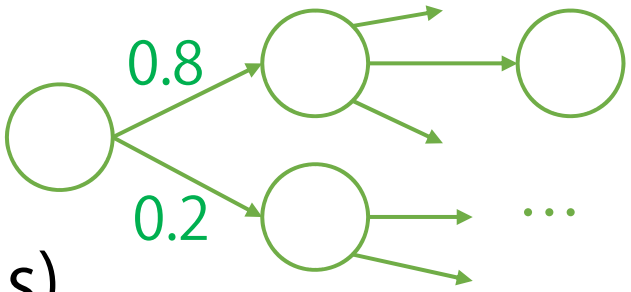
  - e.g., 10 min. to produce the "same-level" test suite as human (in terms of code coverage and mutation score)
  - e.g., use in Facebook (Sapienz)

[Molina et al, Java Unit Testing Tool Competition - Sixth Round]
[https://code.fb.com/developer-tools/sapienz-intelligent-automated-software-testing-at-scale/]

Yet Another: Statistical Model Checking

- Computational models with probabilities

➡ "Probability calculation" of (un)desirable situations is too heavy (e.g., on Marcov models)



➡ *Hypothesis testing* or *probability estimation* by a lot of trials (executions/simulations)

- Capability (and demand) to tailor to the problem by using prior knowledge on the domain/problem

[Jha et al., A Bayesian Approach to Model Checking Biological Systems, CMSB'11]
[Zuliani et al, Bayesian statistical model checking with application to Stateflow/Simulink verification, FMSD'13]

ERATO-MMSD Project (Complete)

led by Ichiro Hasuo (NII) (2016-2022)

Group 0: Metamathematical Integration

$T \quad + \quad e \quad \rightarrow \quad T(e)$

Category Theory

Group 1: Heterogenous Formal Methods

$T_1 \quad + \quad e_1 \quad \rightarrow \quad T_1(e_1)$

$T_2 \quad + \quad e_2 \quad \rightarrow \quad T_2(e_2)$

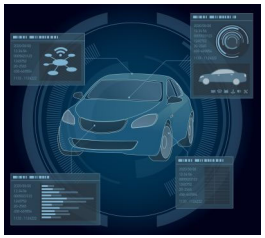
…

Computer Science

Control Theory

Group 2: Formal Methods in Industry

Advanced setting in autonomous driving



Automotive Industry

Practical setting to improve present practices

Group 3: Formal Methods and Intelligence

Heuristics, Evolutionary, Search-based approaches

Optimiz ation

Software Engineering

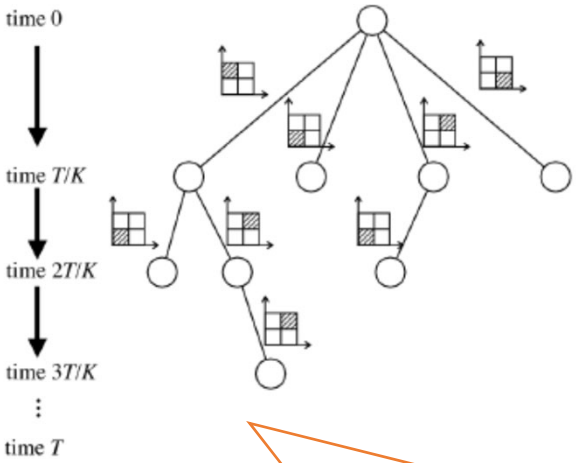
Machine Learning

Topic Example: More Exploratory Falsification

- Apply a systematic exploration method
  - Avoid local optima (too much exploitation)
  - Also obtain informative data on the search space

➡ Monte-Carlo Tree Search and its algorithms

- Common for bandit problems, AI Go game player, etc.
- Used to record the "smells" over the search space (time-staged) in the "trials-evaluation" cycles



What about making full throttle in the first time slot, …

[Zhang et al., Two-Layered Falsification of Hybrid Systems Guided by Monte Carlo Tree Search, EMSOFT'18]

Pragmatic Side, Integrated

What I mentioned so far

Theory Side
(Formal)

Falsification
Search-based Testing
Statistical Model Checking

More Exploratory
Falsification

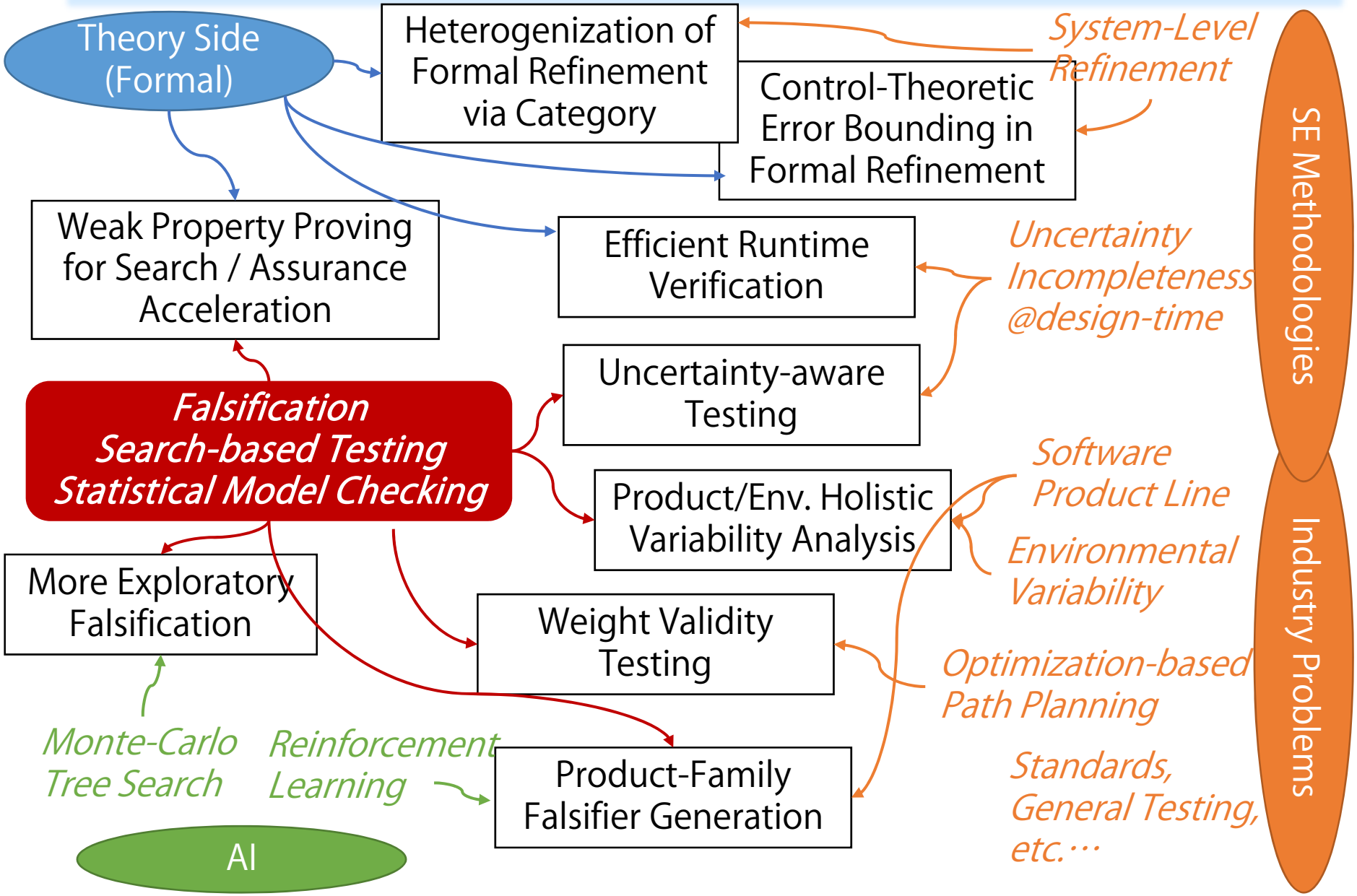
Monte-Carlo
Tree Search

AI

SE Methodologies

Industry Problems

Ongoing Work (Group 3 Viewpoint)



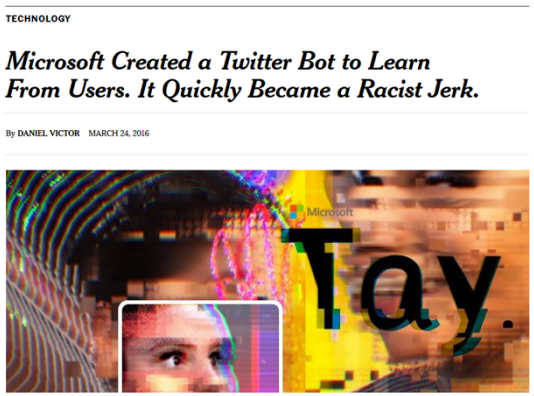
A Little on Machine Learning

AI and ML

- Everyone is talking about AI and ML, recently more about risks and concerns in terms of dependability

Accidents of autonomous cars

[http://www.dailymail.co.uk/news/article-3677101/Tesla-told-regulators-fatal-Autopilot-crash-nine-days-happened.html]


TECHNOLOGY

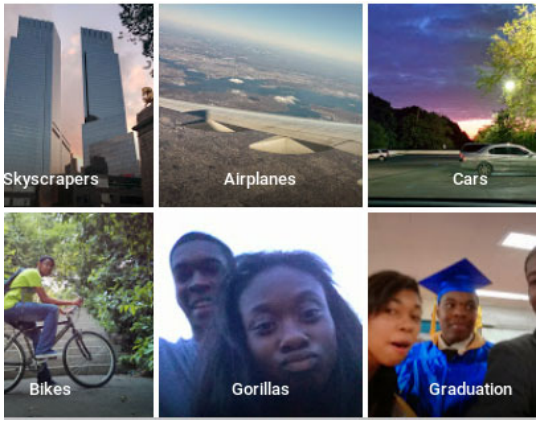
Microsoft Created a Twitter Bot to Learn From Users. It Quickly Became a Racist Jerk.

By DANIEL VICTOR MARCH 24, 2016

Improper online learning

[https://www.nytimes.com/2016/03/25/technology/microsoft-created-a-twitter-bot-to-learn-from-users-it-quickly-became-a-racist-jerk.html]

Technically unsolved problems at Google Photo

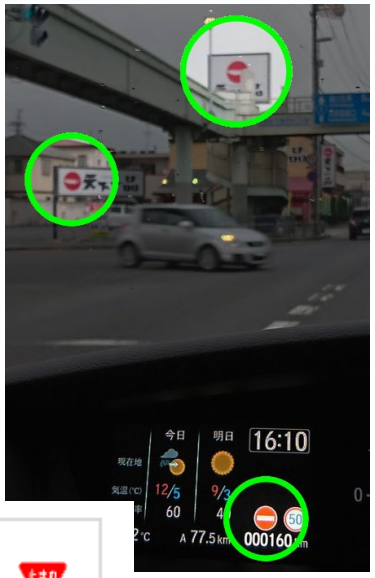


[https://www.theguardian.com/technology/2015/jul/01/google-sorry-racist-auto-tag-photo-app]
[https://www.theguardian.com/technology/2018/jan/12/google-racism-ban-gorilla-black-people]

What will you do if you are responsible? (1)

- When Honda sees ramen shop sign
  - First buzz in Dec 2017

    [https://twitter.com/_gyochan_/status/938240168078622720]
    [https://twitter.com/Bleu_kakeru727/status/937680760491753473]



  - Now a caution on the web site

    http://www.honda.co.jp/hondasensing/feature/srf/

色や形の判別が
つきにくく、
象の標識が無いのに、
標識を表示する

/www.tenkaippin.co.jp/
ny.html]

  - Second buzz in Sep 2018

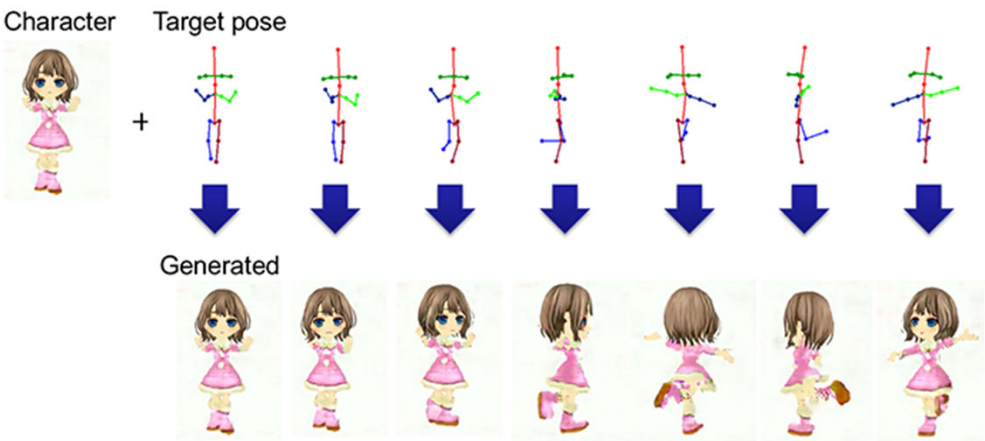
8:24 PM - 12 Sep 2018

15,099 Retweets 20,908 Likes

Can you find beforehand or prevent adverse (?) news??

What will you do if you are responsible? (2)

- From DeNA (May 2018)
  - Generate an image of a certain pose
  - Generate a movie given a pose sequence while changing the character



[https://dena.com/intl/anime-generation/]

What do you ensure to sell this to anime companies?

Essential Difference in ML

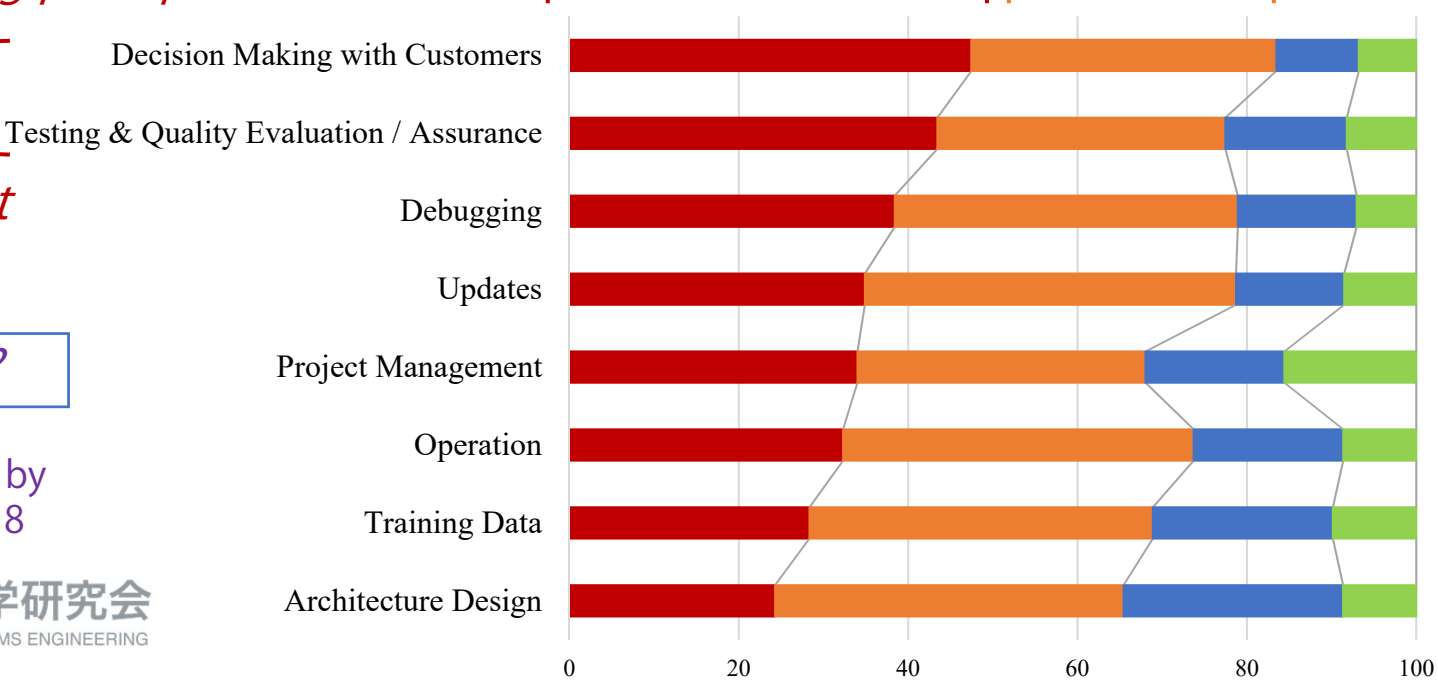
- With ML, we obtain the behavior of a component (e.g., a neural net) **inductively** from training data

  Black-box, imperfect, non-testable (no or costly oracle), unexplainable, has adversarial examples, …

Similar principles but methods unavailable

Existing principles do not work

Especially when arguing the product and its quality



How difficult??

Questionnairesurvey by SIG-MLSE, Japan, 2018

機械学習工学研究会
MACHINE LEARNING SYSTEMS ENGINEERING

Resulting Characteristics (1)

Often difficult/costly or impossible to define the right output for each arbitrary input
(no deductive/logical specification)

- The “unit testing” principle invalidated
- Obvious faults (in training data, configuration, learning-algorithm code) not detected, possibly
- Fault localization (debugging) very difficult
- Not to straightforward to have enormous number of test cases to increase the confidence (e.g., random testing)

Resulting Characteristics (2)

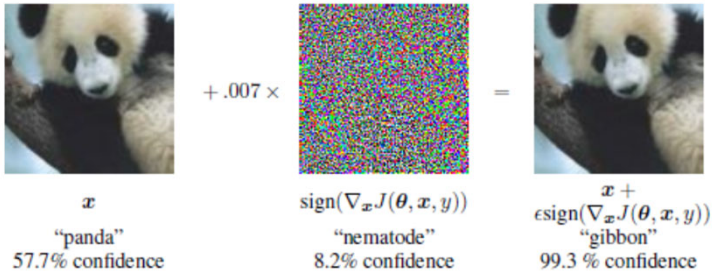
Imperfect and has limitation on performance, impossible to estimate the performance before construction or changes

- Contract not based on the specification but only on “the best effort together”
- Half-a-year effort to find “we should give up”
- Requiring much courage and creative ideas to find acceptable usages with the imperfectness

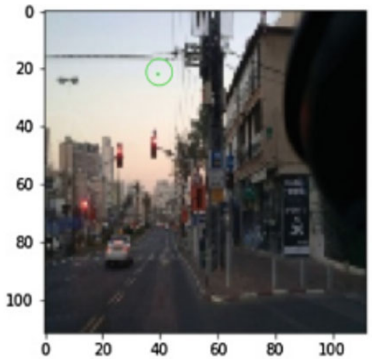
Resulting Characteristics (3)

Impossible to describe the boundary of what can be done and what cannot be done

Basically no confidence on how it works with untested data



Have adversarial examples (slight input changes cause large output change)



- Very difficult to have confidence on the quality
- The "equivalence class" principle invalidated

[Goodfellow et al., Explaining and Harnessing Adversarial Examples, 2015]

[Wicker et al., Feature-Guided Black-Box Safety Testing of Deep Neural Networks, 2018]

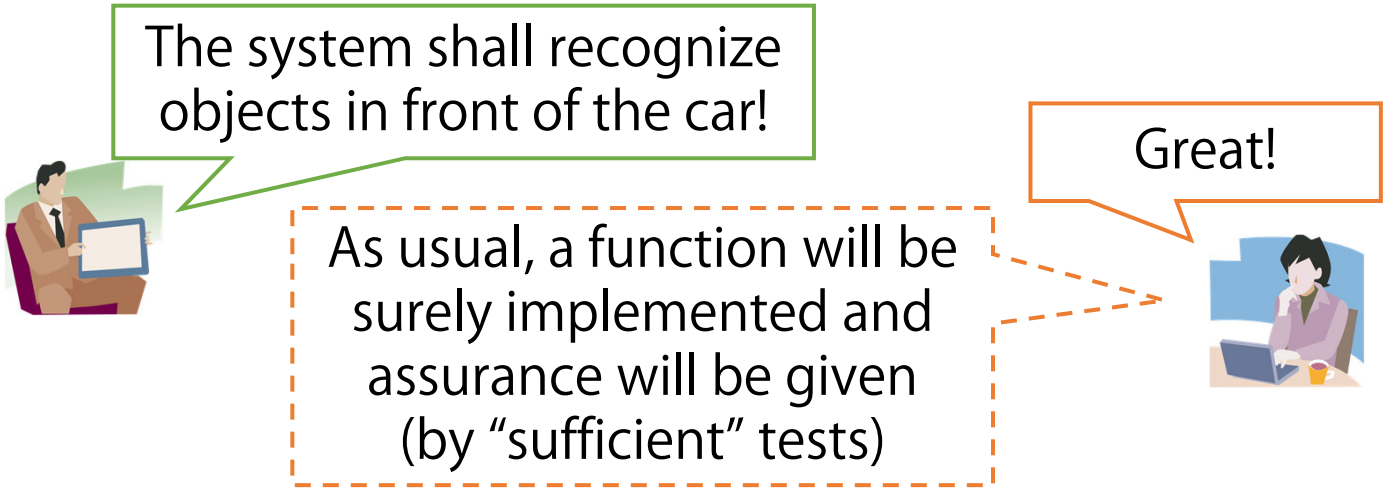
V&V Research Emerging in SE Community

- SMT-based verification [CAV'17]
- Search-based testing with "neuron coverage" criteria [SOSP'17] [ICSE'18]
- Testing based on System-Level Requirements [NFM'17]
- Safe reinforcement learning by formal methods (a few papers on very similar goals) [AAAI'18]
- Verification by stochastic game [TACAS'18]
- Metamorphic testing [ISSTA'18]
- Empirical study on bug statistics [ISSTA'18]
- Mutation Analysis [ISSRE'18]
- Updated coverage criteria [ASE'18]
- Fairness testing [ASE'18]
- (and more) [https://github.com/TrustAI/Literature-on-DNN-Verification-and-Testing]

Example of Change: Requirements & Tests

- Argue the product and its quality!

  (engineer-engineer or engineer-customer)



Example of Change: Requirements & Tests

- Argue the product and its quality!

  (engineer-engineer or engineer-customer)

The system shall recognize objects in front of the car!

~~As usual, a function will be surely implemented and assurance will be given (by "sufficient" tests)~~

Great!

It's imperfect and we cannot make assurance on the behavior for untested data

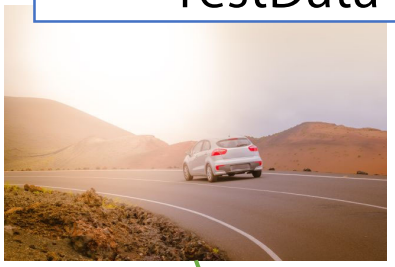
Oh, then we need to carefully <u>assess</u> the quality based on <u>what are actually tested</u>

Topic Example: Attributed Tests to Arguments

- To specify & check constraints
- To describe current status
- To discuss validity
- To compare with operational data

We tested with 100,000 data!

What data … ? Did you test misty days?



carCameraImg1 : TestData

atr1 : Attribute

atr2 : Attribute

weatherReport : AttributionMethod

middle : Confidence

situation : AttributeType

sunrise : AttributeValue

weatherCondition : AttributeType

cloudy : AttributeValue

imgSet1 : TestDataSet

#misty = 2000 : DataSetProperty

Actual representation may be spreadsheet, GUI, or whatever

[Ishikawa, Concepts in Quality Assessment for Machine Learning - From Test Data to Arguments, ER'18]

Not Sufficient (Always, Forever)

- This attribution/requirement-based testing is necessary
  - Convincing with human-explainable "exhaustiveness"
  - To collect "missing" test data
- But this is just a "hope" and cannot be perfect
  - The implementation (neural net) may be looking at different aspects

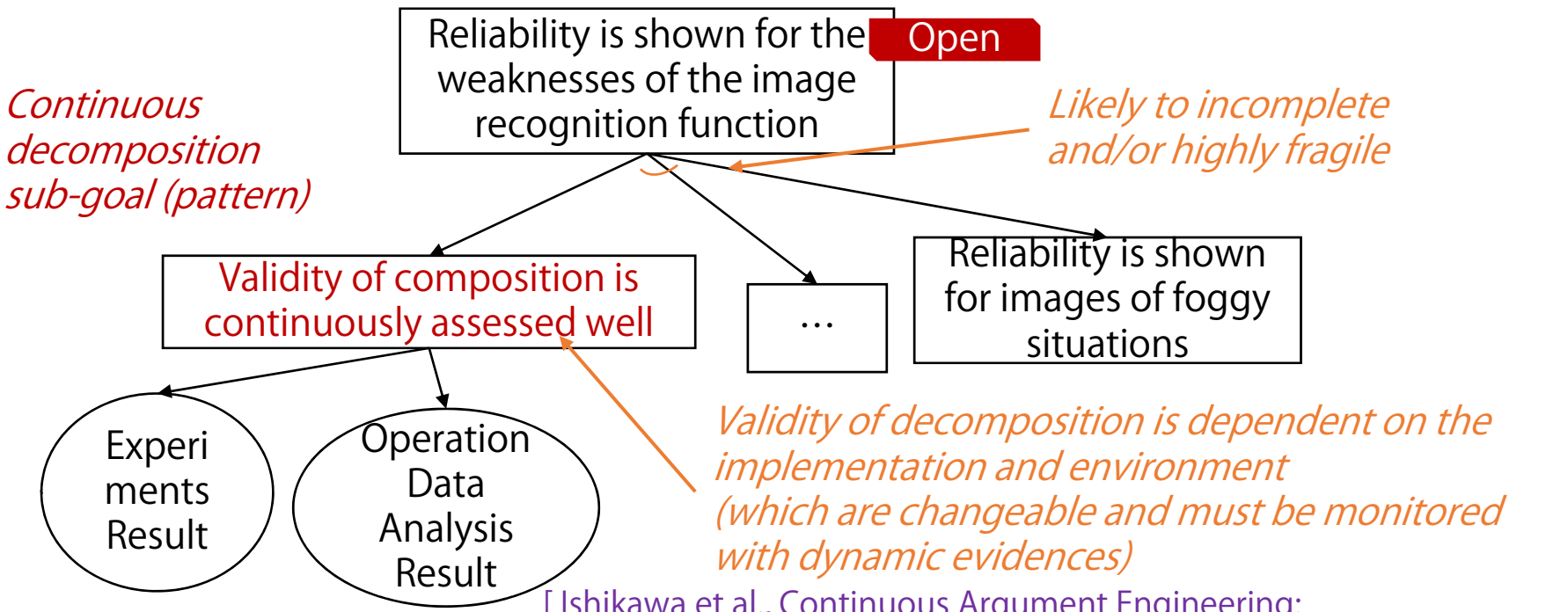
(a) Husky classified as wolf (b) Explanation

[Ribeiro et. al., KDD'16]

- Need to look at experimental and operational results and update the test plan, continuously

Topic Example: Continuous Arguments Eng.

- Explicitly model intrinsic uncertainty in arguments to be aware of risk and need for continuous update
  - Uncertainty in goal decomposition, evidence contribution, and feasibility of goals



[Ishikawa et al., Continuous Argument Engineering: Tackling Uncertainty in Machine Learning based Systems, ASSURE'18]

Toward (Another) Integrated Approach

- QAML: a new project for Quality Assurance on Machine Learning-based Systems

ML Theory

Statistical Assurance

Requirements
& Risk Analysis

e.g., Theoretical results of post-selection inference -> useful for testing!

System

ML Model

Programming Language & Theory

Testing, V&V

Collaboration with
- Social scientist
- Medical industry

Summary

- Integrated Approach to (Intelligent) CPS
  - Necessary to be pragmatically effective for the very difficult problems
  - Team prepared to answer to various queries from the industry (with collective expertise)
  - Just a fun !

Long ago I was talking about "categories" ,
when my core area was the web: like "travel" and "sport",
and now again!