# Dynamic Consistency Analysis for Convergent Operators


Alva L. Couch and Marc Chiarini

Tufts University

{couch,mchiar01}@cs.tufts.edu

# Overview

- Background
- Fixed-point operations
- Emergent consistency
- Practical considerations
- The Maelstrom Theorem
- Summary

# Background

- We can describe network management policies as sets of *convergent operators*.
- Sets of operators can *approximate autonomic computing* (by encapsulating control loops inside operators).
- This is the theoretical basis for Cfengine.

# Fixed point operators

- We define a *fixed point* as a clearly defined, stable, and policy-conformant state.
- A *fixed point operator* moves system state toward a fixed point, or leaves it unchanged if it is at a fixed point.
- A fixed point process is a series of invocations of one or more fixed point operators.
- Example: removal of unwanted rain-water.
  - Catch and remove individual raindrops (ECA).
  - Equip all streets with drains and gutters (FPRD).

# Consistency

- Centralized management strategies require defining overarching policies.
- Reasonable policies are consistent, in the sense that they *do not contain contradictions*.
- In the case of convergent operators, the *set of active operators is the policy*.
- Then w*hat does consistency mean?*

# A controversial claim

*Logical consistency is a useless concept* in a ubiquitous computing network, because:

- Operators can implement fixed points as *algorithms* rather than as rules.
- Codifying the results of the algorithms as rules may be *impossible* for sufficiently complex and/or non-deterministic algorithms.
- *One cannot have complete knowledge* of the set of operators in effect.

# A new "consistency"

Instead, we need *emergent consistency:*

- Consistency of operators is an *emergent property of their application.*
- A consistent set of operators converges to a *common fixed point.*
- We call this *reachable consistency*.
- Inconsistent sets of operators oscillate between conflicting fixed points.

# Reachability

- It is possible that reachability varies with system state, i.e., the starting point for operators.
- Operators can be reachably consistent even if we don’t know about all of them.
- If a set of operators is consistent in isolation, and is not consistent when deployed, then another unknown operator is present.

# Exists *vs* emerges

- In traditional policy theory, consistency is a property that either *exists* or *does not exist.*
- In our theory, consistency either *emerges* or *fails to emerge*.
- Thus it is a *time-varying phenomenon*.
- Purpose of this paper: discuss *when* consistency should emerge, and *with what probability*.

# Single-step operators

- To begin, let’s study perhaps the simplest kind of operator.
- A *convergent single-step operator* does one of two things:
  - Leaves any *acceptable state* alone without change.
  - Changes any *unacceptable state* to an acceptable state.
- In other words, all single-step operators o are *idempotent:* o(o(X))=o(X) for target system X.

# Emergent consistency

- Suppose we execute each of n fixed-point single-step operators once, in sequence.
- Then if consistency is not present, it will be present.
- Reason: if any operator is not at its fixed point, then there must be a conflict.

# Probabilistic execution

Suppose that:

- We have *n* convergent, single-step operators.
- Operator invocations are independent.
- The probability that each operator has been applied by time *t* is $1\text{-}e^{-\lambda t}$ (memoryless, exponential inter-arrival times).
- At time *t*, *we have observed* that some operators have not achieved a fixed point.

Then:

- Prob(operators consistent at time *t*) $\leq 1 - (1\text{-}e^{-\lambda t})^n$.

# Proof

- If the operators *are* consistent, then some operator must not have been applied yet.
- (operators consistent) $\rightarrow \neg$(all operators applied)
- Thus Prob(operators consistent)
  $\leq$ Prob($\neg$(all operators applied))
  = 1-Prob(all n operators applied)
  = $1-(1-e^{-\lambda t})^n$ (since operator invocations are independent).

# Subtleties of this approach

- This is not classical hypothesis testing.
- It is a simple result of implication:
  If for hypotheses A and B, A→ B:
  then States(A) ⊆ States(B)
  and thus Prob(A) ≤ Prob(B).
- This allows one to bound probabilities.
- Bounds are not tight, but may be useful nonetheless.

# In practice

- As time passes and consistency has not been observed, the *probability of inconsistency increases*.
- The previous result allows us to know *when to stop waiting for consistency to emerge*.

# Precedences

- Suppose we have *n* fixed-point operators with precedences between them.
- E.g., a package cannot be configured until it is installed.
- Each operator checks for its preconditions and does not become operative until they are satisfied.
- The system achieves a fixed point if all operators eventually become operative and idempotent.

# Emergent ordering of precedences

- Suppose you have n single-step fixed-point operators with precedences, and you execute the sequence of n operators n times.
- Then if consistency has not emerged, the operators cannot be consistent.
- Key to proof: “Maelstrom Theorem”.

# The Maelstrom Theorem

- If *n* operators are aware of their dependences, then all dependences are satisfied in at most $n^2$ operator invocations.
- Idea of proof: n=4, any permutation of four operators is contained in four sequences of four operators:

```
1 2 3 4 1 2 3 4 1 2 3 4 1 2 3 4
^ ^ ^ ^                          1234
^ ^   ^     ^                    1243
...
      ^     ^     ^     ^        4321
```

# Stochastic invocations

Theorem: suppose that:

- We have n fixed-point operators with precedences.
- Each operator is invoked repeatedly with exponential inter-arrival times with mean inter-arrival time $\lambda$.
- Then if consistency has not been observed at time t, then Prob(operators are consistent) $\leq 1-(1-e^{-\lambda t/n})^{n*n}$

# Proof(1)

- Suppose we have observed that no fixed point has emerged at time t.

Then:

- All operators applied each t/n seconds

$\rightarrow$ All permutations have been tried

(by maelstrom argument)

$\rightarrow$ Operators not consistent.

# Proof(2)

- Suppose we have observed that no fixed point has emerged at time t.

Then:

- Prob(All operators applied each t/n seconds)

$\leq$ Prob(all permutations have been tried)

$\leq$ Prob(operators not consistent).

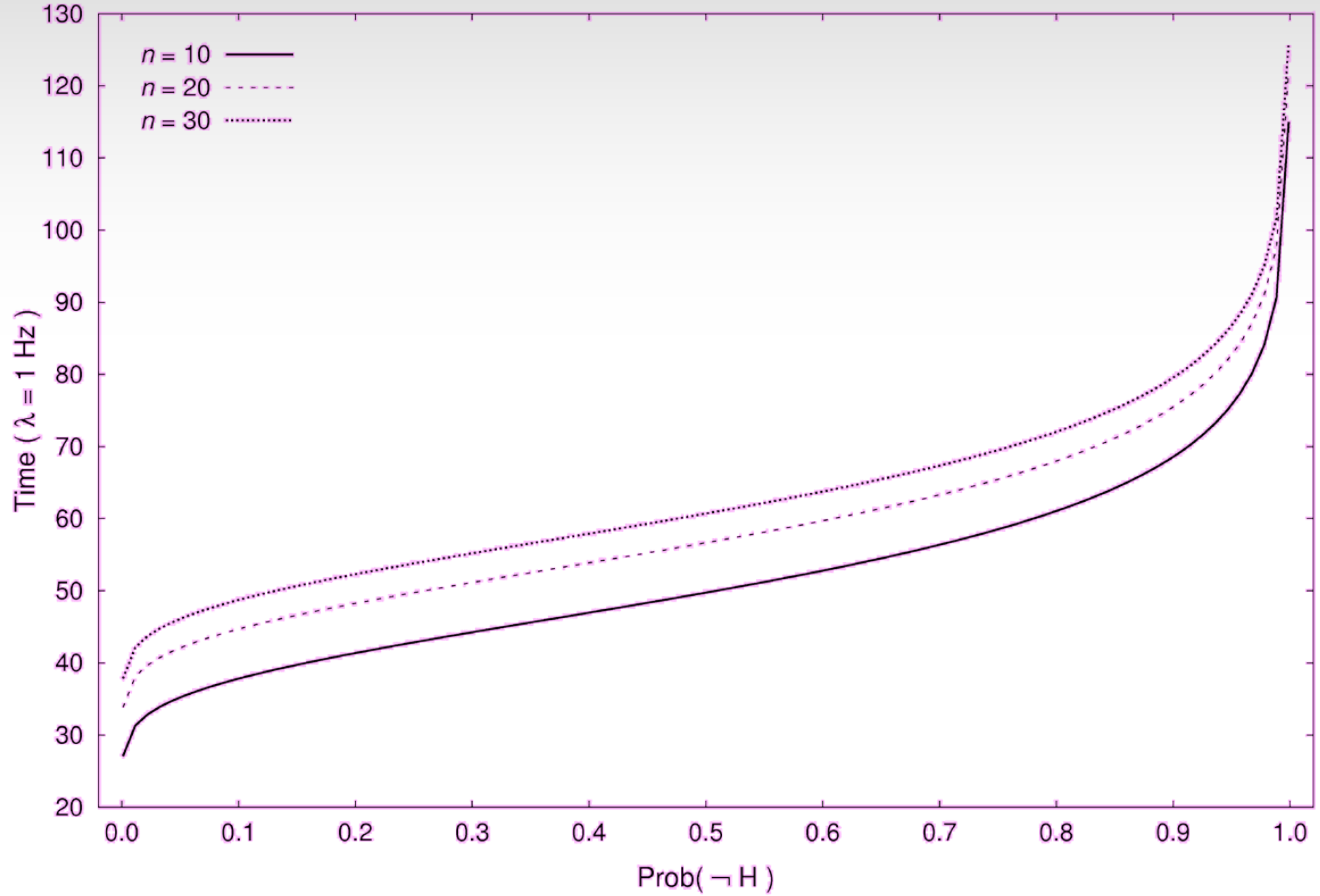
# Proof(3)

- But
  Prob(all operators applied each t/n seconds)
  $= (1\text{-}e^{-\lambda t/n})^{n*n}$ (invoking independence).
- So Prob(operators consistent) $\leq 1 - (1\text{-}e^{-\lambda t/n})^{n*n}$

# The big deal

- As $t \to \infty$, Prob(consistency)$\to 0$, and one can decide when to give up on consistency!

# Title



130
120
110
100
90
80
70
60
50
40
30
20
Time ( λ = 1 Hz )
n = 10
n = 20
n = 30
0.0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1.0
Prob( ¬ H )

# Applying the maelstrom theorem

- Suppose we have *n* single-step operators with precedence chains of at most k operators.
- Suppose we apply all operators at rate $\lambda$ with exponential inter-arrival times.
- Suppose we observe at time *t* that consistency has not been achieved.
- Then Prob(operators are consistent) $\leq 1\text{-}(1\text{-}e^{-\lambda t})^{kn}$
- Idea of proof: as before, bound by implication.