Local Checkability, No Strings Attached

Klaus-Tycho Förster, Thomas Lüdi, Jochen Seidel, Roger Wattenhofer
January 06, 2016 @ ICDCN 2016 - Singapore
Towards Robust Distributed Systems

Eric A. Brewer
UC Berkeley and Inktomi

Current distributed systems, even the ones that work, tend to be very fragile: they are hard to keep up, hard to manage, hard to grow, hard to evolve, and hard to program. In this talk, I look at several issues in an attempt to clean up the way we think about these systems. These issues include the fault model, high availability, graceful degradation, data consistency, evolution, composition, and autonomy.

These are not (yet) provable principles, but merely ways to think about the issues that simplify design in practice. They draw on experience at Berkeley and with giant-scale systems built at Inktomi, including the system that handles 50% of all web searches.

Prove
Towards Robust Distributed Systems

Eric A. Brewer
UC Berkeley and Inktomi

Current distributed systems, even the ones that work, tend to be very fragile: they are hard to keep up, hard to manage, hard to grow, hard to evolve, and hard to program. In this talk, I look at several issues in an attempt to clean up the way we think about these systems. These issues include the fault model, high availability, graceful degradation, data consistency, evolution, composition, and autonomy.

These are not (yet) provably principles, but merely ways to think about the issues that simplify design in practice. They draw on experience at Berkeley and with large-scale systems built at Inktomi, including the system that handles 50% of all web searches.

Brewer’s Conjecture and the Feasibility of Consistent, Available, Partition-Tolerant Web Services

Seth Gilbert and Nancy Lynch
Laboratory for Computer Science
Massachusetts Institute of Technology
Cambridge, MA 02139
seth@mit.edu, lynch@theory.cs.mit.edu

Abstract

When designing distributed web services, there are three properties that are commonly desired: consistency, availability, and partition tolerance. It is impossible to achieve all three. In this note, we prove this conjecture in the asynchronous network model, and then discuss solutions to this dilemma in the partially synchronous model.

1 Introduction

At PODC 2000, Brewer¹, in an invited talk [2], made the following conjecture: it is impossible for a web service to provide the following three guarantees:

- Consistency
- Availability
- Partition-tolerance
Complexity Theory

P

NP

Prove In polynomial time

Verify In polynomial time
Overview

• Introduction
• Background & model
• Undirected vs directed communication
• Study of $s \rightarrow t$ reachability
• Conclusion
Let's get Distributed

- Is $n$ even?
- $\Omega(n)$ rounds, even with unique identifiers in the \textsc{local}-model
Let's get Distributed

- Is \( n \) even?
- \( \Omega(n) \) rounds, even with unique identifiers in the \text{LOCAL}\-model
- Prover assigns 1 bit
Let's get Distributed

- Is $n$ even?
- $\Theta(n)$ rounds in the \textsc{local}-model
- Prover assigns 1 bit $\rightarrow$ \texttt{Verify} in 1 round
Let's get Distributed

- Is $n$ even?
- $\Theta(n)$ rounds in the $\text{LOCAL}$-model
- Prover assigns 1 bit -> verify in 1 round
- Other way to think of it: 1 bit of non-determinism
- General question: How many bits necessary/sufficient?
Accepting a proof

- Every node outputs **Yes** -> Proof accepted
- One node outputs **No** -> Proof rejected
Accepting a proof

- Every node outputs **Yes** -> Proof accepted
- One node outputs **No** -> Proof rejected
  - _Prover_ chose the wrong proof
Accepting a proof

- Every node outputs **Yes** -> Proof accepted
- One node outputs **No** -> Proof rejected
  - **Prover** chose the wrong proof
  - Property does not hold
Overview

• Introduction
• Background & model
• Undirected vs directed communication
• Study of $s - t$ reachability
• Conclusion
Overview

• Introduction
• **Background & model**
• Undirected vs directed communication
• Study of $s \rightarrow t$ reachability
• Conclusion
Some Related Work

• [Naor and Stockmeyer, STOC 1993]: *What can be computed locally?*

• [Göös and Suomela, PODC 2011]: *Locally Checkable Proofs (LCP)*

• [Korman et al., ICDCN 2006, ...]: *Proof Labeling Schemes (PLS)*

• [Fraigniaud et al., FOCS 2011,...]: *Nondeterministic Local Decision (NLD)*
  — [Fraigniaud et al., DISC 2012,...]: “Randomization”
“No Strings attached”

• No knowledge of $n$
• No identifiers
• No port numbers
• No relaying of messages - just one round
Graphs and Communication

- (Weakly) Connected graphs $G = (V, E)$ with $|V| = n$
  - **Yes** instances $G \in Y$ & **No** instances $G \notin Y$

- **Undirected**: $U(v)$ for every $v \in V$
  - multiset of labels of all neighbors

- **Directed**: $D_1(v)$ for every $v \in V$
  - Multiset $I$ of labels of all incoming-neighbors

- **Directed**: $D_2(v)$ for every $v \in V$
  - two multisets $(I, O)$ of labels of all
    - incoming-neighbors
    - outgoing-neighbors
Local Checkability

• **Prover** \( \mathcal{P} \) gets as input \( G \in Y \)
  – Assigns a labels \( \ell(v) \) for every \( v \in V \)

• **Verifier** \( \mathcal{V} \) is a distributed algorithm that gets as input at node \( v \) both \( \ell(v) \) & \( U(v) \) (or \( D_1(v) / D_2(v) \))
  – Outputs either **Yes** or **No**

• A Prover-Verifier pair \( (\mathcal{P}, \mathcal{V}) \) is correct for \( Y \) if:
  – \( G \in Y \) & labels from \( \mathcal{P} \): \( \mathcal{V} \) outputs **Yes** at all nodes
  – \( G \notin Y \): \( \mathcal{V} \) outputs **No** for at least one node
Prover-Verifier Pairs

• We investigate if there are correct $(\mathcal{P}, \mathcal{V})$ for some $Y$
  – (abbreviated by U-PVP, $D_1$-PVP, $D_2$-PVP)

• The quality of a PVP is its proof size
  – $f(n)$, if the PVP uses at most $f(n)$ bits for each label in any Yes instance with at most $n$ nodes

• The U-proof size of $Y$ is the smallest proof size for which there exists a correct U-PVP
  – Analogous for $D_1$-proof size / $D_2$-proof size

• In this talk: All logarithms are of base 2 and rounded up to be of integer value
Overview

- Introduction
- Background & model
- Undirected vs directed communication
- Study of $s \rightarrow t$ reachability
- Outlook
Overview

• Introduction
• Background & model
• **Undirected vs directed communication**
• Study of $s - t$ reachability
• Outlook
Undirected vs Directed Communication

• The different models can induce different amount of bits required in the proof size
  – Or might even render a problem impossible

• Example problem $Y : CYCLE$
  – $U-CYCLE$: all undirected graphs containing a cycle
  – $D-CYCLE$: all directed graphs containing a directed cycle
$D$-CYCLE: Is there a $D_1$-PVP?

$G$: 

![Graph Image]
**D-CYCLE**: Is there a $D_1$-PVP?

$G$: 

```
c_2 -> c_1 <- A -> B
```

- $c_2$: Yes
- $c_1$: Yes
- $A$: Yes
- $B$: Yes
**D-CYCLE**: Is there a $D_1$-PVP?

**$G$:**

- $c_2$ → $c_1$ → $a$ → $b$

**$H$:**

- $b'$ → $a$ → $b$
**D-CYCLE**: Is there a $D_1$-PVP?

$G:$

- $c_2$ -> $A$
- $c_1$ <- $A$
- $a$ -> $B$

$H:$

- $b'$ <- $B$
- $a$ -> $A$
- $b$ -> $B$
$D$-CYCLE: Is there a $D_1$-PVP?

There is no $D_1$-PVP for $D$-CYCLE
## CYCLE

<table>
<thead>
<tr>
<th>Problem</th>
<th>Directed one-way</th>
<th>Directed two-way</th>
<th>Undirected</th>
</tr>
</thead>
<tbody>
<tr>
<td>CYCLE</td>
<td>Impossible</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
$D$-$CYCLE$: Is there a $D_2$-PVP?
**D-CYCLE:** Is there a $D_2$-PVP?

- **Prover $P$** labels nodes as follows:
  - In a directed cycle? -> 0
  - Else: Minimum distance to a cycle
    - (in the underlying undirected graph)
  - Proof size: $\log n$ bits
**D-CYCLE**: Is there a $D_2$-PVP?

- **Prover $P$** labels nodes as follows:
  - In a directed cycle? \(\rightarrow 0\)
  - Else: Minimum distance to a cycle
    - (in the underlying undirected graph)
  - Proof size: $\log n$ bits
**D-CYCLE**: Is there a $D_2$-PVP?

- **Verifier $V$ returns Yes**
  - For nodes $v_c$ with label $\ell(v_c)=0$ if for $(I,O)$ holds:
    - $0 \in O$ and $0 \in I$
  - For the other nodes $v$ with label $\ell(v)$ if
    1. There is a label $\ell(u)$ in $(I,O)$ with $\ell(v)=\ell(u)+1$, and
    2. There is no label $\ell(u')$ in $(I,O)$ with $\ell(v)>\ell(u')+1$
**D-CYCLE**: Is there a $D_2$-PVP?

- **Verifier $V$ returns Yes**
  - For nodes $v_c$ with label $\ell(v_c)=0$ if for $(I,O)$ holds:
    - $0 \in O$ and $0 \in I$
  - For the other nodes $v$ with label $\ell(v)$ if
    1. There is a label $\ell(u)$ in $(I,O)$ with $\ell(v)=\ell(u)+1$, and
    2. There is no label $\ell(u')$ in $(I,O)$ with $\ell(v)>\ell(u')+1$
Is the described $D_2$-PVP correct?

• **Yes** instances labeled by $\mathcal{P}$:
  – Only nodes in directed cycles labeled with 0 -> **Yes**
  – All other nodes: Label is defined by minimum distance to a directed cycle -> **Yes**

• **No** instances:
  – Is there a node with label 0? Follow “0-path” -> **No**
  – No node with label 0, but one with label k?
    • Follow “descending path” -> **No**
D₂-proof size: $\Omega(\log n)$ bits
**$D_2$-proof size: $\Omega(\log n)$ bits**

**$G$:**

\[ v_1 \rightarrow v_{i-1} \rightarrow v_i \rightarrow v_{i+1} \rightarrow v_{i+2} \rightarrow v_{j-1} \rightarrow v_j \rightarrow v_{j+1} \rightarrow v_{j+2} \rightarrow v_{n-2} \rightarrow v_{n-1} \rightarrow v_n \]
\textbf{D}_2\text{-proof size: } \Omega(\log n) \text{ bits}

\[G:\]
\[
\begin{array}{ccccccccccccc}
  & v_1 & \rightarrow & v_{i-1} & \rightarrow & v_i & \rightarrow & v_{i+1} & \rightarrow & v_{i+2} & \rightarrow & v_{j-1} & \rightarrow & v_j & \rightarrow & v_{j+1} & \rightarrow & v_{j+2} & \rightarrow & v_{n-2} & \rightarrow & v_{n-1} & \rightarrow & v_n
\end{array}
\]

\[H:\]
\[
\begin{array}{ccccccccccccc}
  A & \rightarrow & B & \rightarrow & \cdots & \rightarrow & A & \rightarrow & B & \rightarrow & \cdots & \rightarrow & A & \rightarrow & B & \rightarrow & \cdots & \rightarrow & \text{Yes}
\end{array}
\]
D_2-proof size: \( \Omega(\log n) \) bits

\[ G: \quad v_1 \xrightarrow{} v_{i-1} \xrightarrow{} v_i \xrightarrow{} v_{i+1} \xrightarrow{} v_{i+2} \xrightarrow{} v_{j-1} \xrightarrow{} v_j \xrightarrow{} v_{j+1} \xrightarrow{} v_{j+2} \xrightarrow{} v_{n-2} \xrightarrow{} v_{n-1} \xrightarrow{} v_n \]

\[ H: \quad u_i \xrightarrow{} u_{i+1} \xrightarrow{} u_{i+2} \xrightarrow{} u_{j-1} \xrightarrow{} u_j \xrightarrow{} u_{j+1} \xrightarrow{} u_{i+2} \xrightarrow{} u_{j-1} \]

Yes
## CYCLE

<table>
<thead>
<tr>
<th>Problem</th>
<th>Directed one-way</th>
<th>Directed two-way</th>
<th>Undirected</th>
</tr>
</thead>
<tbody>
<tr>
<td>CYCLE</td>
<td>Impossible</td>
<td>$\Theta(\log n)$</td>
<td></td>
</tr>
</tbody>
</table>
U-proof size: At least 2 Bits

\[ G_1: \quad \begin{array}{cccc} & & & \\ & & 0 & \\ & 0 & 0 & 0 \\ \end{array} \quad \begin{array}{c} H_1: \quad 0 \\ 0 \end{array} \]

\[ G_2: \quad \begin{array}{cccc} & & & \\ & & 0 & \\ & 0 & 1 & 0 \\ \end{array} \quad \begin{array}{c} H_2: \quad 0 \\ 1 \\ 0 \end{array} \]

\[ G_3: \quad \begin{array}{cccc} & & & \\ & & 1 & \\ & 1 & 1 & 0 \\ \end{array} \quad \begin{array}{c} H_3: \quad 0 \\ 1 \\ 1 \\ 0 \end{array} \]

\[ G_4: \quad \begin{array}{cccc} & & & \\ & & 1 & \\ & 1 & 0 & 0 \\ \end{array} \quad \begin{array}{c} H_4: \quad 0 \\ 0 \end{array} \]
U-PVP for CYCLE with 2 bits

- **Prover $P$** labels nodes as follows:
  - In a cycle? $\rightarrow$ 3
  - Else: Remove all cycles, remaining graph is a forest
    - For each tree $T$:
      » Create a root $r$ adjacent to a cycle in $G$ with label 0
      » Other nodes: Distance to $r$ modulo 3

- **Proof size:** 2 bits
U-PVP for CYCLE with 2 bits

• **Verifier** $V$ returns **Yes**
  
  – For nodes $v_c$ with label $\ell(v_c)=3$ if holds:
    • Two neighbors with label 3 exist
  
  – For the other nodes $v$ with label $\ell(v) \in \{0,1,2\}$ if
    1. There is no neighbor with label $\ell(v)$, and
    2. Exactly one neighbor exists with label $\ell(v)−1 \mod 3$ or at least one neighbor with label of 3
Is the described U-PVP correct?

- **Yes** instances labeled by $\mathcal{P}$:
  - Only nodes in cycles labeled with 3 -> **Yes**
  - Without the cycles, all other nodes are in a tree with labels as distance to root mod 3, and root is adjacent to a cycle -> **Yes**
Is the described U-PVP correct?

• **Yes** instances labeled by $\mathcal{P}$:
  – Only nodes in cycles labeled with 3 -> **Yes**
  – Without the cycles, all other nodes are in a tree with labels as distance to root mod 3, and root is adjacent to a cycle -> **Yes**

• **No** instances (without a cycle):
  – Is there a node with label 3? They form a forest, consider any leaf-> **No**
  – Else: follow “descending path” -> **No**
## CYCLE, ACYCLIC, TREE

<table>
<thead>
<tr>
<th>Problem</th>
<th>Directed one-way</th>
<th>Directed two-way</th>
<th>Undirected</th>
</tr>
</thead>
<tbody>
<tr>
<td>CYCLE</td>
<td>Impossible</td>
<td>$\Theta(\log n)$</td>
<td>2</td>
</tr>
</tbody>
</table>
CYCLE, ACYCLIC, TREE

<table>
<thead>
<tr>
<th>Problem</th>
<th>Directed one-way</th>
<th>Directed two-way</th>
<th>Undirected</th>
</tr>
</thead>
<tbody>
<tr>
<td>CYCLE</td>
<td>Impossible</td>
<td>$\Theta(\log n)$</td>
<td>2</td>
</tr>
<tr>
<td>TREE</td>
<td>$\Theta(\log n)^*$</td>
<td>$\Theta(\log n)$</td>
<td>$\Theta(\log n)^*$</td>
</tr>
<tr>
<td>ACYCLIC</td>
<td>$\Theta(\log n)$</td>
<td>$\Theta(\log n)$</td>
<td>same as Tree</td>
</tr>
</tbody>
</table>

Idea for Tree:
• Label root as 0
• Other nodes: Label is distance from root

Idea for Acyclicity:
• Label nodes without incoming edges as 0
• Other nodes: Max. incoming label plus 1

*: [Korman et al., Distributed Computing 2010]: Proof labeling schemes
Overview

• Introduction
• Background & model
• Undirected vs directed communication
• Study of $s - t$ reachability
• Conclusion
Overview

• Introduction
• Background & model
• Undirected vs directed communication
• **Study of \( s \rightarrow t \) reachability**
• Conclusion
$s \rightarrow t$ Reachability

• Is there a (directed) path from $s$ to $t$?

“To ask meaningful questions about connectivity [...] we have the promise that there is exactly one node with label $s$ and exactly one node with label $t$.”

[Göös and Suomela, PODC 2011]

• We thus assume that there are two nodes with the unique labels $s$ and $t$

• U-proof size of 1 bit (e.g., [Immermann, 1999]):
  – Label nodes along a shortest $s \rightarrow t$ path with 1, else 0
Directed $s - t$ Reachability

• $D_2$-PVP with port numbers: $O(\log \Delta)$ bits
  – With $\Delta$ being max degree
  – Idea: “Point at successor and predecessor” along a shortest $s - t$ path

• Open question:
  
  “Is there a proof labelling scheme with $O(1)$-bit proofs?”

[Göös and Suomela, PODC 2011]
D₁-PVP for $s - t$ Reachability

• We don’t have port numbers...

• Idea: Take a shortest $s - t$ path $s, v_1, ..., v_j, t$
  – Label according to distance to $s$ along the path
  – All other nodes: Label of 0

• Proof size of $\log n$
$D_1$-proof size: $\Omega(\log n)$ bits
$D_1$-proof size: $\Omega(\log n)$ bits
D₁-proof size: \( \Omega(\log n) \) bits

\[
\begin{align*}
G:\quad & s & \rightarrow & B & \rightarrow & A & \quad & t & \rightarrow & A & \rightarrow & C
\end{align*}
\]
$D_1$-proof size: $\Omega(\log n)$ bits

$G$:
$D_1$-proof size: $\Omega(\log n)$ bits

$G$: $s 
\xrightarrow{} B \xrightarrow{} A$
$t 
\xleftarrow{} A \xleftarrow{} C$

$H$: $s 
\xrightarrow{} B \xrightarrow{} A$
$t 
\xleftarrow{} A \xleftarrow{} C$
D$_1$-proof size: $\Omega(\log n)$ bits

There is no D$_1$-PVP with $f(\Delta)$ bits!
D₂-PVP for s − t Reachability

• As we don’t have port numbers, we could use the D₁-PVP with \( \log n \) bits

• With port numbers: \( O(\log \Delta) \) bits

• Let us create port numbers!
D$_2$-PVP for $s \rightarrow t$ Reachability

- Idea: A 2-hop coloring needs $\leq \Delta^2 + 1$ colors
  - Encoding each color: $O(\log \Delta)$ bits

- 2-hop coloring can be checked locally
  - All colors in the 1-hop neighborhood different?

- Thus, we can point “back and forth” along edges, by emulating port numbers with $O(\log \Delta)$ bits
Conclusion

• Summary
  – All three models of communication differ
  – Our lower bound examples have constant degree
    • Can drop the 1 round restriction and go local
  – Directed $s - t$ reachability:
    • One-Way: Proof size of $\Theta(\log n)$ bits, $f(\Delta)$ bits don’t suffice
    • Two-Way: Emulating port numbers -> $O(\log \Delta)$ bits proof size

• Open Questions
  – What happens in biologically inspired systems?
    • E.g., no multisets but sets & finite automata verifier?
  – What is the correct answer to $D_2 s - t$ reachability?
  – Can similar techniques be deployed in production networks?
Thank you

Klaus-Tycho Förster, Thomas Lüdi, Jochen Seidel, Roger Wattenhofer
January 06, 2016 @ ICDCN 2016 - Singapore