44

Simpler Proofs for Concurrent Reading and Writing

Fred B. Schneider[1]

Abstract

Simplified proofs are given for Lamport's protocols to coordinate concurrent reading and writing.

1 Introduction

In most computing systems, hardware ensures that read and write operations to some basic unit of memory can be considered mutually exclusive. As a result, a read that overlaps with a write is serialized and will appear either to precede that write or to follow it. Operations that make multiple accesses to memory are not serialized by the hardware. The programmer must ensure that when such operations overlap, they produce meaningful results.

In this paper, we give simplified proofs for two protocols proposed by Lamport [1] for coordinating read and write operations that involve multiple accesses to memory. The two key theorems in [1] are long and intricate. Here, we show that both are corollaries of a single, relatively simple theorem. Our facility with proofs and the use of formalism has improved significantly in a little over 15 years.[2] This is due, in part, to the influence of Edsger Dijkstra.

[1]This material is based on work supported in part by the Office of Naval Research under contract N00014-86-K-0092, the National Science Foundation under Grant No. CCR-8701103, and Digital Equipment Corporation. Any opinions, findings, and conclusions or recommendations expressed in this publication are those of the author and do not reflect the views of these agencies.

[2][1] was first submitted for publication in September 1974.

2 Words from digits

Consider a computing system i
and a digit can contain one of
finite set of values can be encoc
such a sequence of digits a *wor*
operations are performed on it
are performed. Observe that o
word will not be serialized by t
constraints on execution, it is p
obtain a meaningless value. For
from 0 through 9 and a word
encodes the value 099. A read t
might obtain any of the followin
100.

By constraining the order in w
digits are written, we can ensu
writes does obtain a meaningful
easily implemented and non-intr
operations should be delayed, nc
synchronization primitives.

In the protocols that follow,
$w[0]w[1]\ldots w[n]$ of digits. Think
digit and $w[n]$ as the least-signific
being stored by w. We assume
process. Define $w[i]^p$ to be the val
number p.[3] Also, for any sequen
the subsequence consisting of $s[i]$
Thus, $w[0..k]$ is the word constru
$k+1$ digits of w.

A read operation that overlap
value that corresponds to the resu
such values by using a *slice*, a sec
and a slice σ of equal length, defi

$$w^\sigma \equiv w[0]^{\sigma[0]}w[1]^{\sigma[1]}.$$

We write $(N) \otimes v$ to denote a le
equals $w[0]^p w[1]^p \ldots w[n]^p$, the val
ber p.

A slice σ is *non-decreasing* if (

[3]It will be convenient to assume
value to every digit. The new value c

2 Words from digits

Consider a computing system in which the basic unit of memory is a *digit*, and a digit can contain one of $B \geq 2$ distinct values. Any element from a finite set of values can be encoded using a finite sequence of digits. We call such a sequence of digits a *word*. To read the value stored by a word, read operations are performed on its digits; to write a value, write operations are performed. Observe that overlapping read and write operations to a word will not be serialized by the hardware. Therefore, without additional constraints on execution, it is possible for a read that overlaps a write to obtain a meaningless value. For example, suppose digits can encode integers from 0 through 9 and a word w constructed from three digits initially encodes the value 099. A read that is concurrent with a write of value 100 might obtain any of the following results: 099, 090, 009, 000, 199, 190, 109, 100.

By constraining the order in which digits are read and the order in which digits are written, we can ensure that a read overlapping one or more writes does obtain a meaningful value. Desired are constraints that are both easily implemented and non-intrusive. Execution of neither read nor write operations should be delayed, nor should the constraints require elaborate synchronization primitives.

In the protocols that follow, a word w is implemented by a sequence $w[0]w[1]\ldots w[n]$ of digits. Think of $w[0]$ as the most-significant (left-most) digit and $w[n]$ as the least-significant (right-most) digit of a base B number being stored by w. We assume that w is written by a single, sequential process. Define $w[i]^p$ to be the value written to digit $w[i]$ by write operation number p.[3] Also, for any sequence $s = s[0]s[1]\ldots s[n]$, define $s[i..j]$ to be the subsequence consisting of $s[i]\ldots s[j]$, and define $|s|$ to be the length of s. Thus, $w[0..k]$ is the word constructed from the most-significant (left-most) $k+1$ digits of w.

A read operation that overlaps with one or more writes can obtain a value that corresponds to the result of no write operation. We can describe such values by using a *slice*, a sequence of positive integers. For a word w and a slice σ of equal length, define:

$$w^\sigma \equiv w[0]^{\sigma[0]}w[1]^{\sigma[1]}\ldots w[n]^{\sigma[n]}$$

We write $(N) \otimes v$ to denote a length N sequence of v's. Thus, $w^{(n+1)\otimes p}$ equals $w[0]^p w[1]^p \ldots w[n]^p$, the value written to w by write operation number p.

A slice σ is *non-decreasing* if $(\forall i : 0 < i < |\sigma| : \sigma[i-1] \leq \sigma[i])$ and

[3] It will be convenient to assume that a write operation to a word writes a value to every digit. The new value can, of course, be the same as the old.

non-increasing if $(\forall i : 0 < i < |\sigma| : \sigma[i-1] \geq \sigma[i])$. For slices σ and τ such that $|\sigma| = |\tau|$, define

$$\sigma \subseteq \tau \quad \equiv \quad (\forall i : 0 \leq i < |\sigma| : \sigma[i] \leq \tau[i]).$$

Finally, in order to reason about the relative order in which operations occur, define $\mu_i(x)$ to be the number of writes that have been made to digit $w[i]$ as of time x. Observe that $x \leq x'$ implies that $\mu_i(x) \leq \mu_i(x')$ is valid.

3 The main result

We first show that if slices σ and τ satisfy certain restrictions (H1–H3) and values written to w are non-decreasing (H4), then $w^\sigma \leq w^\tau$ where "$\leq$" denotes lexicographic ordering.

Theorem 1 Let σ and τ be slices such that $|\sigma| = |\tau| \geq N+1$. Then,

$$w[0..N]^\sigma \quad \leq \quad w[0..N]^\tau$$

provided:

(H1) σ is non-decreasing,
(H2) τ is non-increasing,
(H3) $\sigma \subseteq \tau$, and
(H4) $w[0..N]^{(N+1)\otimes i} \leq w[0..N]^{(N+1)\otimes j}$ for all $i \leq j$.

Proof From hypothesis H4 and the definition of lexicographic ordering, we conclude that for all $i \leq j$ and any m such that $0 \leq m \leq N$:

$$\text{LO:} \quad w[0..m]^{(m+1)\otimes i} \leq w[0..m]^{(m+1)\otimes j}$$

The proof now proceeds by induction on the number of digits in w.

Base Case: Assume w is constructed using a single digit.

w^σ

$=$ {By assumption that w is a single digit and hypothesis that $|\sigma| \geq N+1$.}

$w[0..0]^{\sigma[0]}$

$\leq$ {By LO, since $\sigma \subseteq \tau$ by H3.}

$w[0..0]^{\tau[0]}$

$=$ {By assumption that w is a single digit and hypothesis that $|\tau| \geq N+1$.}

w^τ

Induction Case: Assume the I
where $0 \leq n < N$. We show th

1. {From H1, σ is
$\sigma[0..n]$ is non-decreas

2. {Definition of nc
$\alpha = (n+1) \otimes \sigma[n+$

{By constructior
non-decreasing.
3. $\sigma[0..n] \subseteq \alpha$.

{By induction hy
1, 2, 3, and LO.
4. $w[0..n]^{\sigma[0..n]} \leq w[0.$

{Definition of lex
5. $w[0..n]^{\sigma[0..n]} w[n+1]^{\sigma}$

{By LO, since $\sigma[$
6. $w[0..n]^{\alpha} w[n+1]^{\sigma[n+1]}$

{Transitivity with
7. $w[0..n]^{\sigma[0..n]} w[n+1]^{\sigma[}$

{Definition of nor
8. $\beta = (n+1) \otimes \tau[n+1]$

{From H2, τ is nc
9. $\tau[0..n]$ is non-increasin

{By construction
non-increasing.}
10. $\beta \subseteq \tau[0..n]$

{By induction hyp
8, 9, 10, and LO.
11. $w[0..n]^{\beta} \leq w[0..n]^{\tau[0}$

{Definition of lexic
12. $w[0..n]^{\beta} w[n+1]^{\tau[n+1]}$

{Transitivity with
13. $w[0..n]^{\sigma[0..n]} w[n+1]^{\sigma[n}$

{$\sigma[0..n+1] = \sigma[0..$
14. $w[0..n+1]^{\sigma[0..n+1]} \leq$

□

Induction Case: Assume the Theorem holds for any $n+1$ digit word $w[0..n]$, where $0 \leq n < N$. We show that it holds for the $n+2$ digit word $w[0..n+1]$.

{From H1, σ is non-decreasing. Therefore, any prefix is.}

1. $\sigma[0..n]$ is non-decreasing.

{Definition of non-increasing.}

2. $\alpha = (n+1) \otimes \sigma[n+1]$ is non-increasing.

{By construction of $\sigma[0..n]$ and α, since (H1) σ is non-decreasing.}

3. $\sigma[0..n] \subseteq \alpha$.

{By induction hypothesis, since H1–H4 are satisfied due to 1, 2, 3, and LO.}

4. $w[0..n]^{\sigma[0..n]} \leq w[0..n]^{\alpha}$

{Definition of lexicographic order.}

5. $w[0..n]^{\sigma[0..n]} w[n+1]^{\sigma[n+1]} \leq w[0..n]^{\alpha} w[n+1]^{\sigma[n+1]}$

{By LO, since $\sigma[n+1] \leq \tau[n+1]$ because (H3) $\sigma \subseteq \tau$.}

6. $w[0..n]^{\alpha} w[n+1]^{\sigma[n+1]} \leq w[0..n+1]^{(n+2)\otimes\tau[n+1]}$

{Transitivity with 5 and 6.}

7. $w[0..n]^{\sigma[0..n]} w[n+1]^{\sigma[n+1]} \leq w[0..n+1]^{(n+2)\otimes\tau[n+1]}$

{Definition of non-decreasing.}

8. $\beta = (n+1) \otimes \tau[n+1]$ is non-decreasing.

{From H2, τ is non-increasing. Therefore, any prefix is.}

9. $\tau[0..n]$ is non-increasing.

{By construction of β and $\tau[0..n]$, since (H2) t is non-increasing.}

10. $\beta \subseteq \tau[0..n]$

{By induction hypothesis, since H1–H4 are satisfied due to 8, 9, 10, and LO.}

11. $w[0..n]^{\beta} \leq w[0..n]^{\tau[0..n]}$

{Definition of lexicographic order.}

12. $w[0..n]^{\beta} w[n+1]^{\tau[n+1]} \leq w[0..n]^{\tau[0..n]} w[n+1]^{\tau[n+1]}$

{Transitivity with 7 and 12.}

13. $w[0..n]^{\sigma[0..n]} w[n+1]^{\sigma[n+1]} \leq w[0..n]^{\tau[0..n]} w[n+1]^{\tau[n+1]}$

{$\sigma[0..n+1] = \sigma[0..n]\sigma[n+1]$ and $\tau[0..n+1] = \tau[0..n]\tau[n+1]$}

14. $w[0..n+1]^{\sigma[0..n+1]} \leq w[0..n+1]^{\tau[0..n+1]}$

□

4 Reading to the left, writing to the right

We can now show that if the digits of w are read from right to left (i.e. $w[n], w[n-1], \ldots, w[0]$) but written from left to right (i.e. $w[0], \ldots, w[n-1]$, $w[n]$) then only certain mixtures of values can be obtained from overlapping writes. In particular, the value read is bounded from below by the value written by the earliest write whose digit is obtained by this read.

Read-Left, Write-Right:
If (i) the sequence of values written to w is non-decreasing, (ii) digits are written from left to right, and (iii) digits are read from right to left, then the value w^τ obtained by the read satisfies $w^{(N+1)\otimes\tau[n]} \leq w^\tau$.

Proof We first show that τ is non-increasing. Let x_i be the time that digit $w[i]$ is read. Thus, $\tau[i] = \mu_i(x_i)$ and, due to hypothesis (iii) that digits are read from right to left, $x_n \leq x_{n-1} \leq \ldots \leq x_0$. For any i, $0 \leq i < n$:

$$
\begin{array}{ll}
& \tau[i] \\
= & \quad \{\text{Assumption that } \tau[i] = \mu_i(x_i).\} \\
& \mu_i(x_i) \\
\geq & \quad \{\text{Digits are written from left to right due to hypothesis (ii).}\} \\
& \mu_{i+1}(x_i) \\
\geq & \quad \{x_i \geq x_{i+1}.\} \\
& \mu_{i+1}(x_{i+1}) \\
= & \quad \{\text{Assumption that } \tau[i] = \mu_i(x_i).\} \\
& \tau[i+1]
\end{array}
$$

The correctness of Read-Left, Write-Right now follows from Theorem 1. Choose $(N+1) \otimes \tau[n]$ for σ; this choice for σ satisfies H1 and H3. We showed above that τ satisfies H2. H4 is satisfied by hypothesis (i). Thus, from Theorem 1 we conclude $w^{(N+1)\otimes\tau[n]} \leq w^\tau$.
□

There are two interesting things to note about this protocol. First, exclusive access to digits is the only synchronization required. Second, read operations and write operations do not delay each other.

5 Reading to the right, writing to the left

By reversing the order in which digits are read and written, we obtain another protocol for concurrent reading and writing. With this protocol,

the value read is bounded fr
write whose digit is obtained

Read-Right, Write-Left:
If (i) the sequence of values w
written from right to left, and
the value w^σ obtained by any

Proof We first show that σ
digit $w[i]$ is read. Thus, $\sigma[i] =$
are read from left to right, x_0

$$
\begin{array}{ll}
& \sigma[i] \\
= & \quad \{\text{Assumption that } \tau \\
& \mu_i(x_i) \\
\leq & \quad \{\text{Digits are written f} \\
& \mu_{i+1}(x_i) \\
\leq & \quad \{x_i \leq x_{i+1}.\} \\
& \mu_{i+1}(x_{i+1}) \\
= & \quad \{\text{Assumption that } \sigma[\\
& \sigma[i+1]
\end{array}
$$

The correctness of Read-Right,
Choose $(N+1) \otimes \sigma[n]$ for τ;
showed above that σ satisfies H
from Theorem 1 we conclude $w^{\cdot}$
□

As before, exclusive access to
and operations are never delaye

6 Final remarks

This paper is now in its third re
and informal proofs. These, like
induction on the number of digit
wrong — the informality let deta
sion of the paper contained corre
total of four lemmas were requir
though the two pairs of lemmas
Theorem 1 of the current version
mas, and its proof results from c

the value read is bounded from above by the value written by the latest write whose digit is obtained by this read.

Read-Right, Write-Left:
If (i) the sequence of values written to w is non-decreasing, (ii) digits are written from right to left, and (iii) digits are read from left to right, then the value w^σ obtained by any read satisfies $w^\sigma \leq w^{(N+1)\otimes\sigma[n]}$.

Proof We first show that σ is non-decreasing. Let x_i be the time that digit $w[i]$ is read. Thus, $\sigma[i] = \mu_i(x_i)$ and, due to hypothesis (iii) that digits are read from left to right, $x_0 \leq x_1 \leq \ldots \leq x_n$. For any i, $0 \leq i < n$:

$$
\begin{array}{ll}
 & \sigma[i] \\
= & \quad \{\text{Assumption that } \tau[i] = \mu_i(x_i).\} \\
 & \mu_i(x_i) \\
\leq & \quad \{\text{Digits are written from right to left due to hypothesis (ii).}\} \\
 & \mu_{i+1}(x_i) \\
\leq & \quad \{x_i \leq x_{i+1}.\} \\
 & \mu_{i+1}(x_{i+1}) \\
= & \quad \{\text{Assumption that } \sigma[i] = \mu_i(x_i).\} \\
 & \sigma[i+1]
\end{array}
$$

The correctness of Read-Right, Write-Left now follows from Theorem 1. Choose $(N+1) \otimes \sigma[n]$ for τ; this choice for τ satisfies H2 and H3. We showed above that σ satisfies H1. H4 is satisfied by hypothesis (i). Thus, from Theorem 1 we conclude $w^\sigma \leq w^{(N+1)\otimes\sigma[n]}$.
□

As before, exclusive access to digits is the only synchronization required, and operations are never delayed.

6 Final remarks

This paper is now in its third revision. The first version contained simple and informal proofs. These, like the proof of Theorem 1 given above, used induction on the number of digits in a word. Unfortunately, the proofs were wrong — the informality let details slip through the cracks. The second version of the paper contained correct and formal versions of those proofs. A total of four lemmas were required —two lemmas for each protocol— although the two pairs of lemmas had proofs that were disturbingly similar. Theorem 1 of the current version of the paper generalizes two of those lemmas, and its proof results from combining the proofs of those two lemmas.

Acknowledgments: David Gries read and commented on many earlier versions of this paper. Jay Misra pointed out the errors in the first version of the paper and proposed the statement of Theorem 1 along with a (long) proof. Avoiding a case analysis in that proof led to the proof finally given above.

References

[1] Leslie Lamport. Concurrent reading and writing. *Comm. ACM*, 20(11):806–811, Nov. 1977.

Fred B. Schneider,
Department of Computer Science,
Cornell University,
4130 Upson Hall,
Ithaca, New York 14853–7501,
U.S.A.

45

Goodbye Junct

Carel S. Scholten

In [2] the notions conjunctivity a
tivity"— are introduced. For th
duction of conjunctivity. Prima
former f and a set V of predica

(0) (f is conjunctive over
$[f.(\mathbf{A}\, X : X \in V : X)$

In the above, the infix dot der
brackets denote universal quanti

Conjunctivity properties of f alo
over all V of a certain type. We

(f is universally conjun
(f is positively conjunc
(f is conjunctive over a

In [1] Dijkstra raises the quest
the right notion to introduce, m
better to consider the two implic
hand side of (0) separately.

Well, to begin with, the notic
faithfully for about eight years I
thank you" in the title of this no

Nevertheless, let us follow Dijk
ing two properties of f and V:

(1) $[f.(\mathbf{A}\, X : X \in V : X) =$
(2) $[f.(\mathbf{A}\, X : X \in V : X) \Leftarrow$

For (1), no special name is pro
is satisfied by all pairs f, V satis
any f, the three assertions "(1) h
V", and "f is monotonic" are equ
mention