Formalization of a Generalized Protocol for Clock Synchronization in Isabelle/HOL

Alwen Tiu
January 9, 2020

Abstract

We formalize the generalized Byzantine fault-tolerant clock synchronization protocol of Schneider. This protocol abstracts from particular algorithms or implementations for clock synchronization. This abstraction includes several assumptions on the behaviors of physical clocks and on general properties of concrete algorithms/implementations. Based on these assumptions the correctness of the protocol is proved by Schneider. His proof was later verified by Shankar using the theorem prover EHDM (precursor to PVS). Our formalization in Isabelle/HOL is based on Shankar’s formalization.

Contents

1 Introduction 1

2 Isar proof scripts 2
   2.1 Types and constants definitions ........................................... 2
   2.2 Clock conditions ............................................................ 3
      2.2.1 Some derived properties of clocks ..................................... 5
      2.2.2 Bounded-drift for logical clocks (IC) .................................... 5
   2.3 Agreement property ......................................................... 6

1 Introduction

In certain distributed systems, e.g., real-time process-control systems, the existence of a reliable global time source is critical in ensuring the correct functioning of the systems. This reliable global time source can be implemented using several physical clocks distributed on different nodes in the distributed system. Since physical clocks are by nature constantly drifting away from the “real time” and different clocks can have different drift rates, in such a scheme, it is important that these clocks are regularly adjusted so that they are closely synchronized within a certain application-specific safe bound. The design and verification of clock synchronization protocols are often complicated by the additional requirement that the protocols should work correctly under certain types of errors, e.g., failure of some clocks, error in communication network or corrupted messages, etc.

There has been a number of fault-tolerant clock synchronization algorithms studied in the literature, e.g., the Interactive Convergence Algorithm (ICA) by Lamport and Melliar-Smith [1], the Lundelius-Lynch algorithm [2], etc., each with its own degree of fault tolerance. One important property that
must be satisfied by a clock synchronization algorithm is the agreement property, i.e., at any time $t$, the difference of the clock readings of any two non-faulty processes must be bounded by a constant (which is fixed according to the domain of applications). At the core of these algorithms is the convergence function that calculates the adjustment to a clock of a process, based on the clock readings of all other processes. Schneider [3] gives an abstract characterization of a wide range of clock synchronization algorithms (based on the convergence functions used) and proves the agreement property in this abstract framework. Schneider’s proof was later verified by Shankar [4] in the theorem prover EHDM (precursor to PVS), where eleven axioms about clocks are explicitly stated.

We formalize Schneider’s proof in Isabelle/HOL, making use of Shankar’s formulation of the clock axioms. The particular formulation of axioms on clock conditions and the statements of the main theorems here are essentially those of Shankar’s [4], with some minor changes in syntax. For the full description of the protocol, the general structure of the proof and the meaning of the constants and function symbols used in this formalization, we refer readers to [4].

Acknowledgment I would like to thank Stephan Merz and Pascal Fontaine for useful tips on using Isabelle and particularly the Isar proof language.

2 Isar proof scripts

theory GenClock imports Complex-Main begin

2.1 Types and constants definitions

Process is represented by natural numbers. The type ‘event’ corresponds to synchronization rounds.

type-synonym process = nat
type-synonym event = nat
type-synonym time = real
type-synonym Clocktime = real

axiomatization
$\delta$ :: real and
$\mu$ :: real and
$\varrho$ :: real and
$rmin$ :: real and
$rmax$ :: real and
$\beta$ :: real and
$\Lambda$ :: real and

np :: process and
maxfaults :: process and

$PC$ :: [process, time] $\Rightarrow$ Clocktime and

$VC$ :: [process, time] $\Rightarrow$ Clocktime and

$te$ :: [process, event] $\Rightarrow$ time and
\[ \theta :: \{ \text{process, event} \} \Rightarrow \{ \text{process} \Rightarrow \text{Clocktime} \} \text{ and} \]

\[ IC :: \{ \text{process, event, time} \} \Rightarrow \text{Clocktime} \text{ and} \]

\[ \text{correct} :: \{ \text{process, time} \} \Rightarrow \text{bool} \text{ and} \]

\[ \text{cfn} :: \{ \text{process}, (\text{process} \Rightarrow \text{Clocktime}) \} \Rightarrow \text{Clocktime} \text{ and} \]

\[ \pi :: \{ \text{Clocktime, Clocktime} \} \Rightarrow \text{Clocktime} \text{ and} \]

\[ \alpha :: \text{Clocktime} \Rightarrow \text{Clocktime} \]

definition

\[ \text{count} :: \{ \text{process} \Rightarrow \text{bool}, \text{process} \} \Rightarrow \text{nat} \text{ where} \]
\[ \text{count } f\ n = \text{card } \{ p. \ p < n \land f \ p \} \]

definition

\[ \text{Adj} :: \{ \text{process, event} \} \Rightarrow \text{Clocktime} \text{ where} \]
\[ \text{Adj} = (\lambda \ p \ i. \ \text{if } 0 < i \text{ then } \text{cfn } p \ (\vartheta \ p \ i) - PC \ p \ (te \ p \ i) \text{ else } 0) \]

definition

\[ \text{okRead1} :: \{ \text{process} \Rightarrow \text{Clocktime}, \text{Clocktime}, \text{process} \Rightarrow \text{bool} \} \Rightarrow \text{bool} \text{ where} \]
\[ \text{okRead1 } f\ x\ \text{ppred} \leftarrow\ (\forall \ l\ m. \ \text{ppred } l \land \text{ppred } m \rightarrow |f \ l - f \ m| \leq x) \]

definition

\[ \text{okRead2} :: \{ \text{process} \Rightarrow \text{Clocktime}, \text{process} \Rightarrow \text{Clocktime}, \text{process} \Rightarrow \text{bool} \} \Rightarrow \text{bool} \text{ where} \]
\[ \text{okRead2 } f\ g\ x\ \text{ppred} \leftarrow\ (\forall \ p. \ \text{ppred } p \rightarrow |f \ p - g \ p| \leq x) \]

definition

\[ \text{rho-bound1} :: \{ \text{[process, time]} \} \Rightarrow \text{Clocktime} \Rightarrow \text{bool} \text{ where} \]
\[ \text{rho-bound1 } C \leftarrow (\forall \ p\ s\ t. \ \text{correct } p \ t \land s \leq t \rightarrow C \ p \ t - C \ p \ s \leq (t - s)^*(1 + \varrho)) \]

definition

\[ \text{rho-bound2} :: \{ \text{[process, time]} \} \Rightarrow \text{Clocktime} \Rightarrow \text{bool} \text{ where} \]
\[ \text{rho-bound2 } C \leftarrow (\forall \ p\ s\ t. \ \text{correct } p \ t \land s \leq t \rightarrow (t - s)^*(1 - \varrho) \leq C \ p \ t - C \ p \ s) \]

### 2.2 Clock conditions

Some general assumptions

axiomatization where

\[ \text{constants-ax: } 0 < \beta \land 0 < \mu \land 0 < \varrho \]
\[ \land \varrho \leq r_{\text{max}} \land 0 < \rho \land 0 < np \land \text{maxfaults} \leq np \]

axiomatization where

\[ \text{PC-monotone: } \forall \ p\ s\ t. \ \text{correct } p \ t \land s \leq t \rightarrow PC \ p \ s \leq PC \ p \ t \]

axiomatization where

\[ \text{VClock: } \forall \ p\ t\ i. \ \text{correct } p \ t \land te \ p \ i \leq t \land t < te \ p \ (i + 1) \rightarrow VC \ p \ t = IC \ p \ i \ t \]
axiomatization where
IClock: \( \forall p t i. \这种事情 p t \rightarrow IC p i t = PC p t + Adj p i \)

Condition 1: initial skew
axiomatization where
init: \( \forall p. \这种事情 p 0 \rightarrow 0 \leq PC p 0 \land PC p 0 \leq \mu \)

Condition 2: bounded drift
axiomatization where
rate-1: \( \forall p s t. \这种事情 p t \land s \leq t \rightarrow PC p t - PC p s \leq (t - s) \ast (1 + \rho) \) and
rate-2: \( \forall p s t. \这种事情 p t \land s \leq t \rightarrow (t - s) \ast (1 - \rho) \leq PC p t - PC p s \)

Condition 3: bounded interval
axiomatization where
rts0: \( \forall p t i. \这种事情 p t \land t \leq te p (i+1) \rightarrow t - te p i \leq rmax \) and
rts1: \( \forall p t i. \这种事情 p t \land te p (i+1) \leq t \rightarrow rmin \leq t - te p i \)

Condition 4: bounded delay
axiomatization where
rts2a: \( \forall p q t i. \这种事情 p t \land \这种事情 q t \land te q i + \beta \leq t \rightarrow te p i \leq t \) and
rts2b: \( \forall p q t i. \这种事情 p t \land te p (i+1) \leq t \rightarrow abs(te p i - te q i) \leq \beta \)

Condition 5: initial synchronization
axiomatization where
synch0: \( \forall p. te p 0 = 0 \)

Condition 6: nonoverlap
axiomatization where
nonoverlap: \( \beta \leq rmin \)

Condition 7: reading errors
axiomatization where
readerror: \( \forall p q i. \这种事情 p (te p (i+1)) \land \这种事情 q (te p (i+1)) \rightarrow abs(\theta p (i+1) q - IC q i (te p (i+1))) \leq \Lambda \)

Condition 8: bounded faults
axiomatization where
correct-closed: \( \forall p s t. s \leq t \land \这种事情 p t \rightarrow \这种事情 p s \) and
correct-count: \( \forall t. np - maxfaults \leq count (\lambda p. \这种事情 p t) np \)

Condition 9: Translation invariance
axiomatization where
trans-inv: \( \forall f x. 0 \leq x \rightarrow cfn p (\lambda y. f y + x) = cfn p f + x \)

Condition 10: precision enhancement
axiomatization where
prec-enh:
\( \forall ppred p q f g x y. np - maxfaults \leq count ppred np \land okRead1 f y ppred \land okRead1 g y ppred \land \)
\[ \text{okRead2 } f \ g \ x \ \text{ppred} \land \text{ppred } p \land \text{ppred } q \\
\rightarrow \text{abs}(cfn } p \ f - cfn \ q \ g) \leq \pi \ x \ y \]

Condition 11: accuracy preservation

axiomatization where

\[ \forall \ p \ q \ f \ x. \ \text{okRead1 f } x \ \text{ppred} \land \text{np} - \text{maxfaults} \leq \text{count } p \text{pred } np \land \text{ppred } p \land \text{ppred } q \rightarrow \text{abs}(cfn } p \ f - f \ q) \leq \alpha \ x \]

2.2.1 Some derived properties of clocks

lemma \text{rts0d}:
assumes \( cp \): correct \( p \) \((te \ p \ (i+1))\)
shows \( te \ p \ (i+1) - te \ p \ i \leq \text{rmax} \)
(proof)

lemma \text{rts1d}:
assumes \( cp \): correct \( p \) \((te \ p \ (i+1))\)
shows \( \text{rmin} \leq te \ p \ (i+1) - te \ p \ i \)
(proof)

lemma \text{rte}:
assumes \( cp \): correct \( p \) \((te \ p \ (i+1))\)
shows \( te \ p \ i \leq te \ p \ (i+1) \)
(proof)

lemma \text{beta-bound1}:
assumes \( corr-p \): correct \( p \) \((te \ p \ (i+1))\)
and \( corr-q \): correct \( q \) \((te \ p \ (i+1))\)
shows \( \theta \leq te \ p \ (i+1) - te \ q \ i \)
(proof)

lemma \text{beta-bound2}:
assumes \( corr-p \): correct \( p \) \((te \ p \ (i+1))\)
and \( corr-q \): correct \( q \) \((te \ q \ i)\)
shows \( te \ p \ (i+1) - te \ q \ i \leq \text{rmax} + \beta \)
(proof)

2.2.2 Bounded-drift for logical clocks (IC)

lemma \text{bd}:
assumes \( ie \): \( s \leq t \)
and \( rb1 \): \( \text{rho-bound1 } C \)
and \( rb2 \): \( \text{rho-bound2 } D \)
and \( PC-ie \): \( D \ q \ t - D \ q \ s \leq C p \ t - C p \ s \)
and \( corr-p \): correct \( p \ t \)
and \( corr-q \): correct \( q \ t \)
shows \( | C p \ t - D q \ t | \leq | C p \ s - D q \ s | + 2 \varrho \ast (t - s) \)
(proof)

lemma \text{bounded-drift}:
assumes \( ie \): \( s \leq t \)
and \( rb1 \): \( \text{rho-bound1 } C \)
and \( rb_2: \rho\)-bound2 \( C \)
and \( rb_3: \rho\)-bound1 \( D \)
and \( rb_4: \rho\)-bound2 \( D \)
and \( corr-p: \) correct \( p \ t \)
and \( corr-q: \) correct \( q \ t \)
shows \(| C \ p \ t - D \ q \ t | \leq | C \ p \ s - D \ q \ s | + 2g*(t - s) \)
(sort)

Drift rate of logical clocks

\textbf{lemma} IC-rate1:
\( \rho\)-bound1 \( (\lambda \ p \ t. \ IC \ p \ i \ t) \)
(sort)

\textbf{lemma} IC-rate2:
\( \rho\)-bound2 \( (\lambda \ p \ t. \ IC \ p \ i \ t) \)
(sort)

Auxiliary function \( ICf \): we introduce this to avoid some unification problem in some tactic of isabelle.

\textbf{definition}
\( ICf :: \text{nat} \Rightarrow (\text{process} \Rightarrow \text{time} \Rightarrow \text{Clocktime}) \) where
\( ICf \ i = (\lambda \ p \ t. \ IC \ p \ i \ t) \)

\textbf{lemma} IC-bd:
\textbf{assumes} \ie: \( s \leq t \)
and \( corr-p: \) correct \( p \ t \)
and \( corr-q: \) correct \( q \ t \)
shows \(| IC \ p \ i \ t - IC \ q \ j \ t | \leq | IC \ p \ i \ s - IC \ q \ j \ s | + 2g*(t - s) \)
(sort)

\textbf{lemma} event-bound:
\textbf{assumes} \ie1: \( 0 \leq (t::\text{real}) \)
and \( corr-p: \) correct \( p \ t \)
and \( corr-q: \) correct \( q \ t \)
shows \( \exists \ i. \ t < \max (te \ p \ i) (te \ q \ i) \)
(sort)

2.3 Agreement property

\textbf{definition} \( \gamma_1 \ x = \pi \ (2g*\beta + 2*\Lambda) \ (2*\Lambda + x + 2g*(r\max + \beta)) \)
\textbf{definition} \( \gamma_2 \ x = x + 2g*r\max \)
\textbf{definition} \( \gamma_3 \ x = \alpha \ (2*\Lambda + x + 2g*(r\max + \beta)) + \Lambda + 2g*\beta \)

\textbf{definition}
\( ok\maxsync :: [\text{nat}, \text{Clocktime}] \Rightarrow \text{bool} \) where
\( ok\maxsync \ i \ x \leftarrow (\forall \ p \ q. \ \text{correct} \ p \ (\max (te \ p \ i) (te \ q \ i)) \)
\( \land \ \text{correct} \ q \ (\max (te \ p \ i) (te \ q \ i)) \)
\( \rightarrow | IC \ p \ i (\max (te \ p \ i) (te \ q \ i)) - IC \ q \ i (\max (te \ p \ i) (te \ q \ i))| \leq x \)

\textbf{definition}
\( ok\Clocks :: [\text{process}, \text{process}, \text{nat}] \Rightarrow \text{bool} \) where
\( ok\Clocks \ p \ q \ i \leftarrow (\forall \ t. \ 0 \leq t \land t < \max (te \ p \ i) (te \ q \ i)) \)
\( \land \ \text{correct} \ p \ t \land \text{correct} \ q \ t \)
\( \rightarrow | \text{VCon} \ p \ t - \text{VCon} \ q \ t | \leq \delta \)
lemma okClocks-sym:
assumes ok-pq: okClocks p q i
shows okClocks q p i
(\proof)

lemma ICP-Suc:
assumes corr-p: correct p (te p (i+1))
shows IC p (i+1) (te p (i+1)) = cfn p (θ p (i+1))
(\proof)

lemma IC-trans-inv:
assumes ie1: te q (i+1) ≤ te p (i+1)
and corr-p: correct p (te p (i+1))
and corr-q: correct q (te p (i+1))
shows IC q (i+1) (te p (i+1)) =
cfn (λ n. θ q (i+1) n + (PC q (te p (i+1)) − PC q (te q (i+1))))
(is ?T1 = ?T2)
(\proof)

lemma beta-rho:
assumes ie: te q (i+1) ≤ te p (i+1)
and corr-p: correct p (te p (i+1))
and corr-q: correct q (te p (i+1))
and corr-l: correct l (te p (i+1))
shows |PC l (te p (i+1)) − PC l (te q (i+1))) − (te p (i+1) − te q (i+1))| ≤ β*ϱ + 2*Λ
(\proof)

This lemma (and the next one pe-cond2) proves an assumption used in the precision enhancement.

lemma pe-cond1:
assumes ie: te q (i+1) ≤ te p (i+1)
and corr-p: correct p (te p (i+1))
and corr-q: correct q (te p (i+1))
and corr-l: correct l (te p (i+1))
shows |θ q (i+1) l + (PC q (te p (i+1)) − PC q (te q (i+1))) −
θ p (i+1) l| ≤ 2*ϱ*β + 2*Λ
(is ?M ≤ ?N)
(\proof)

lemma pe-cond2:
assumes ie: te m i ≤ te l i
and corr-k: correct k (te k (i+1))
and corr-l-tk: correct l (te k (i+1))
and corr-m-tk: correct m (te k (i+1))
and ind-hyp: |IC l i (te l i) − IC m i (te l i)| ≤ δS
shows |θ k (i+1) l − θ k (i+1) m| ≤ 2*Λ + δS + 2*ϱ*(rmax + β)
(\proof)

lemma theta-bound:
assumes corr-l: correct l (te p (i+1))
and corr-m: correct m (te p (i+1))
and corr-p: correct p (te p (i+1))

and IC-bound:
\[ |IC_l i (max (te l i) (te m i)) - IC_m i (max (te l i) (te m i))| \leq \delta S \]
shows \[ |\vartheta p (i+1) l - \vartheta p (i+1) m| \leq 2*\Lambda + \delta S + 2*\varrho*(r_{max} + \beta) \]
\(\langle proof \rangle\)

**lemma four-one-ind-half:**
assumes ie1: \( \beta \leq \text{rmin} \)
and ie2: \( \mu \leq \delta S \)
and ie3: \( \gamma t \delta S \leq \delta S \)
and ind-hyp: okmaxsync i \( \delta S \)
and ie4: te q (i+1) \leq te p (i+1)
and corr-p: correct p (te p (i+1))
and corr-q: correct q (te p (i+1))
shows \[ |IC p (i+1) (te p (i+1)) - IC q (i+1) (te p (i+1))| \leq \delta S \]
\(\langle proof \rangle\)

Theorem 4.1 in Shankar's paper.

**theorem four-one:**
assumes ie1: \( \beta \leq \text{rmin} \)
and ie2: \( \mu \leq \delta S \)
and ie3: \( \gamma t \delta S \leq \delta S \)
shows okmaxsync i \( \delta S \)
\(\langle proof \rangle\)

**lemma VC-cfn:**
assumes corr-p: correct p (te p (i+1))
and ie: te p (i+1) < te p (i+2)
sshows VC p (te p (i+1)) = cfn p (\( \vartheta p (i+1) \))
\(\langle proof \rangle\)

Lemma for the inductive case in Theorem 4.2

**lemma four-two-ind:**
assumes ie1: \( \beta \leq \text{rmin} \)
and ie2: \( \mu \leq \delta S \)
and ie3: \( \gamma t \delta S \leq \delta S \)
and ie4: \( \gamma t \delta S \leq \delta \)
and ie5: \( \gamma t \delta S \leq \delta \)
and ie6: te q (i+1) \leq te p (i+1)
and ind-hyp: okClocks p q i
and t-bound1: 0 \leq t
and t-bound2: t < max (te p (i+1)) (te q (i+1))
and t-bound3: max (te p i) (te q i) \leq t
and tpq-bound: max (te p i) (te q i) < max (te p (i+1)) (te q (i+1))
and corr-p: correct p t
and corr-q: correct q t
shows \[ |VC p t - VC q t| \leq \delta \]
\(\langle proof \rangle\)

Theorem 4.2 in Shankar's paper.

**theorem four-two:**
assumes ie1: \( \beta \leq \text{rmin} \)
and \( i e2: \mu \leq \delta S \)
and \( i e3: \gamma_1 \delta S \leq \delta S \)
and \( i e4: \gamma_2 \delta S \leq \delta \)
and \( i e5: \gamma_3 \delta S \leq \delta \)
shows \( \text{okClocks } p q i \)
(\( \text{proof} \))

The main theorem: all correct clocks are synchronized within the bound delta.

\textbf{theorem} \textit{agreement}:
\begin{itemize}
  \item \textbf{assumes} \( i e1: \beta \leq r_{\text{min}} \)
  \item \( i e2: \mu \leq \delta S \)
  \item \( i e3: \gamma_1 \delta S \leq \delta S \)
  \item \( i e4: \gamma_2 \delta S \leq \delta \)
  \item \( i e5: \gamma_3 \delta S \leq \delta \)
  \item \( i e6: 0 \leq t \)
  \item \( cpq: \text{correct } p t \land \text{correct } q t \)
\end{itemize}
shows \( |VC p t - VC q t| \leq \delta \)
(\( \text{proof} \))

\begin{thebibliography}{9}

ibitem{LamportMelliar-Smith:1985}

ibitem{LundeliusLynch:1984}

ibitem{Schneider:1987}

ibitem{Shankar:1992}

\end{thebibliography}