Efficient Large-scale Trace Checking using MapReduce

Srđan Krstić

with

Marcello M. Bersani, Domenico Bianculli, Carlo Ghezzi and Pierluigi San Pietro

POLITECNICO MILANO

uni.lu
UNIVERSITÉ DU LUXEMBOURG

Efficient Large-scale Trace Checking using MapReduce

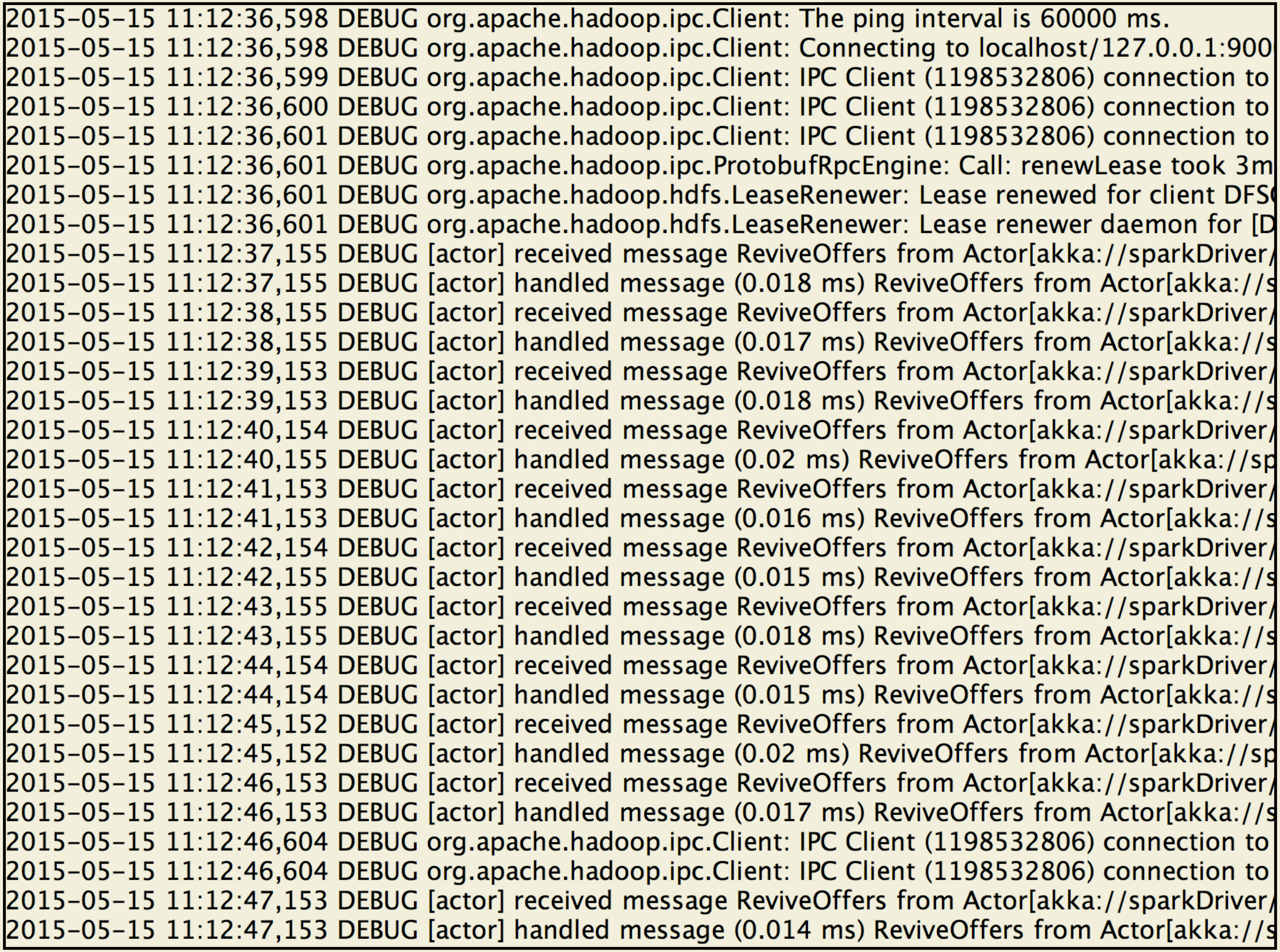
Srđan Krstić

with

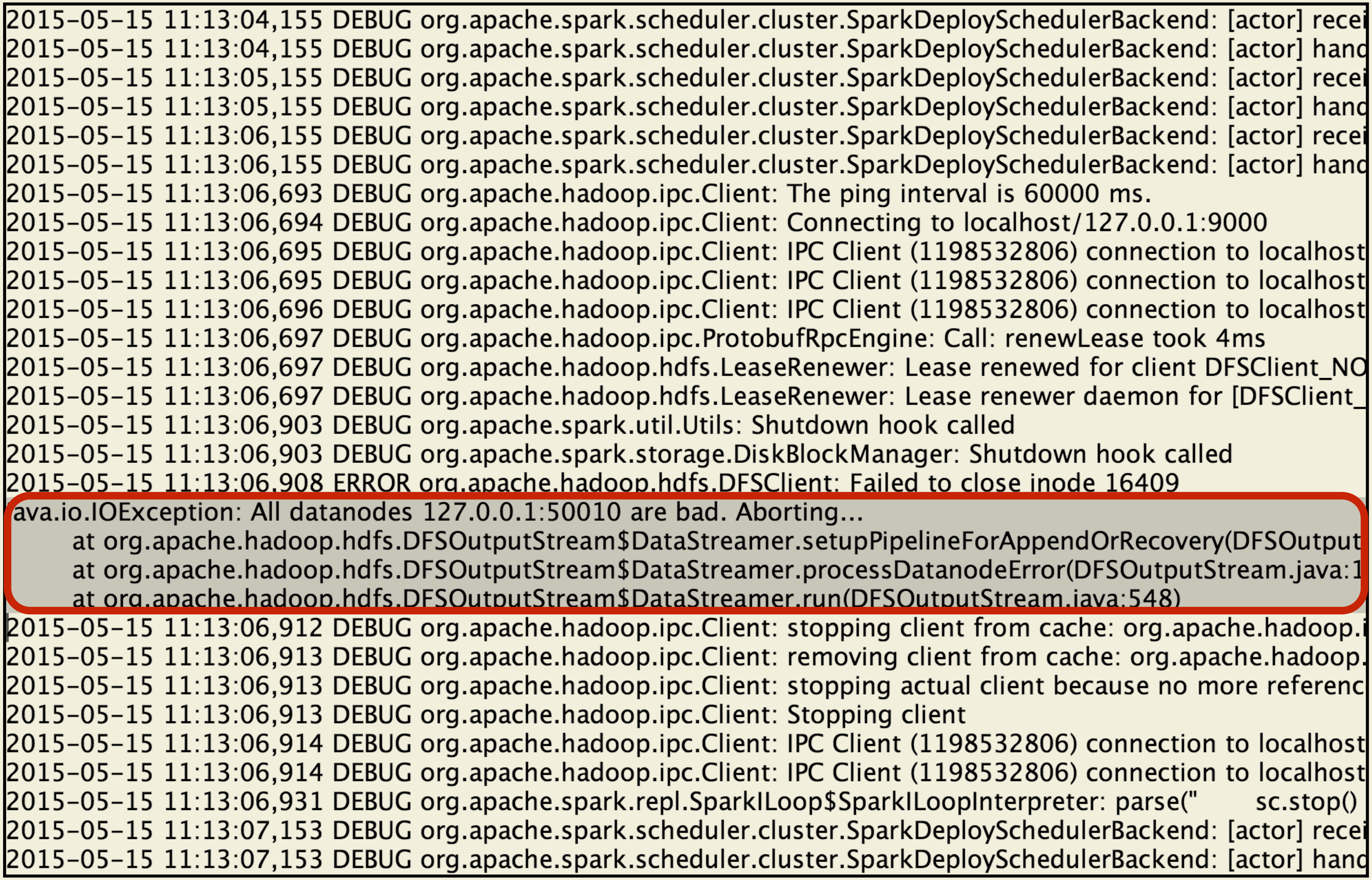
Marcello M. Bersani, Domenico Bianculli, Carlo Ghezzi and Pierluigi San Pietro

POLITECNICO MILANO

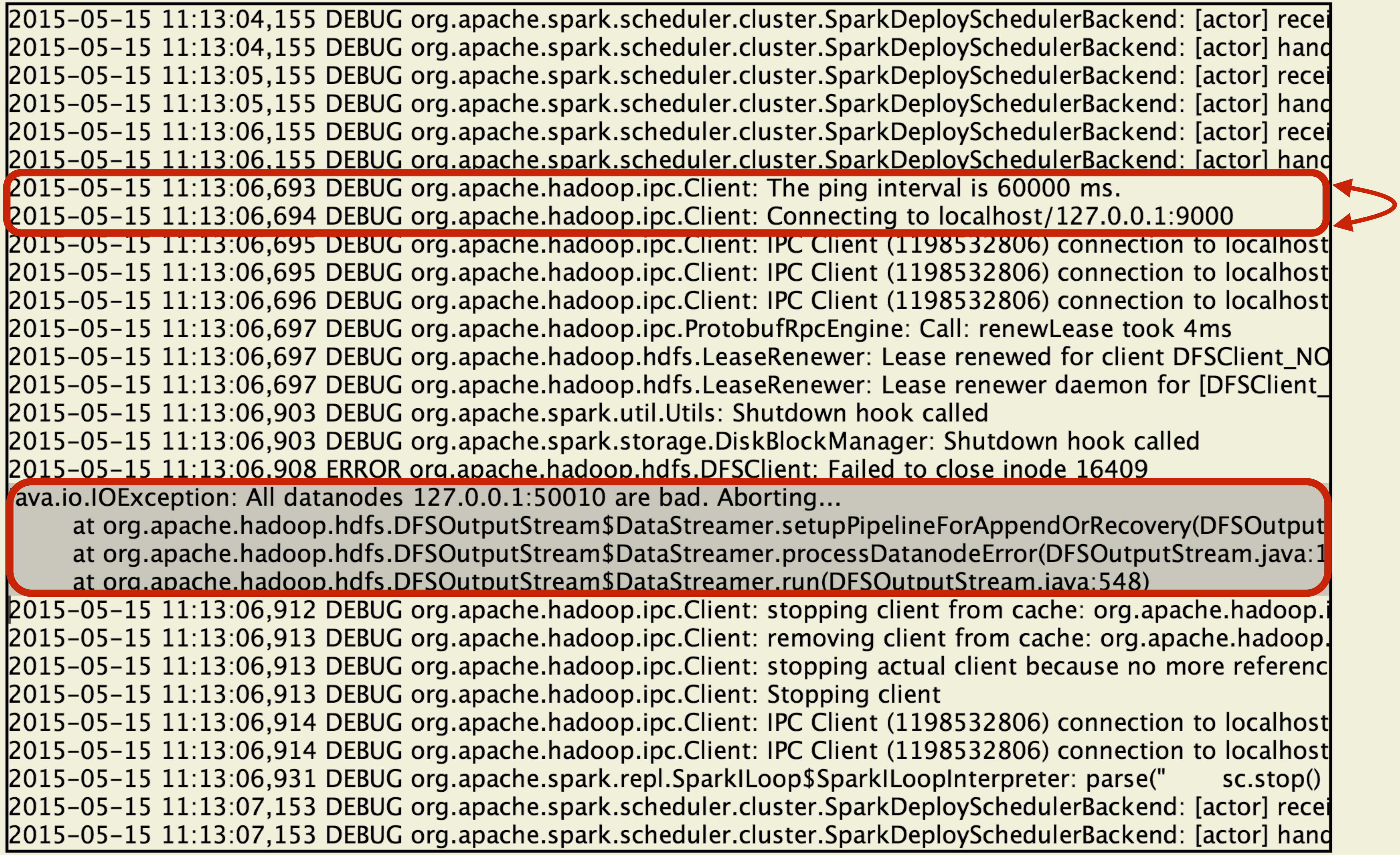
uni.lu UNIVERSITÉ DU LUXEMBOURG



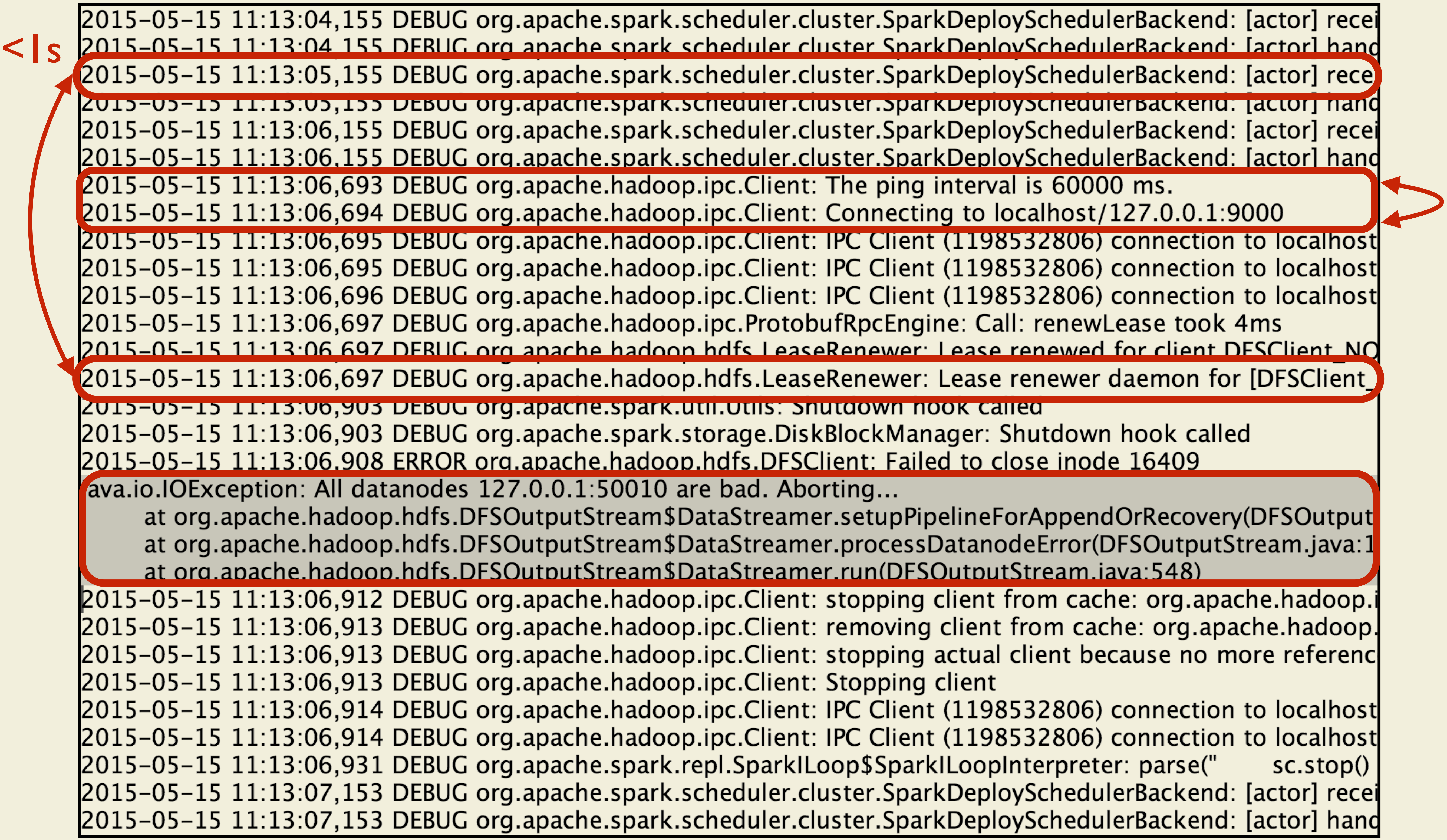
```
2015-05-15 11:12:36,598 DEBUG org.apache.hadoop.ipc.Client: The ping interval is 60000 ms.
2015-05-15 11:12:36,598 DEBUG org.apache.hadoop.ipc.Client: Connecting to localhost/127.0.0.1:900
2015-05-15 11:12:36,599 DEBUG org.apache.hadoop.ipc.Client: IPC Client (1198532806) connection to
2015-05-15 11:12:36,600 DEBUG org.apache.hadoop.ipc.Client: IPC Client (1198532806) connection to
2015-05-15 11:12:36,601 DEBUG org.apache.hadoop.ipc.Client: IPC Client (1198532806) connection to
2015-05-15 11:12:36,601 DEBUG org.apache.hadoop.ipc.ProtobufRpcEngine: Call: renewLease took 3m
2015-05-15 11:12:36,601 DEBUG org.apache.hadoop.hdfs.LeaseRenewer: Lease renewed for client DFS
2015-05-15 11:12:36,601 DEBUG org.apache.hadoop.hdfs.LeaseRenewer: Lease renewer daemon for [D
2015-05-15 11:12:37,155 DEBUG [actor] received message ReviveOffers from Actor[akka://sparkDriver/
2015-05-15 11:12:37,155 DEBUG [actor] handled message (0.018 ms) ReviveOffers from Actor[akka://s
2015-05-15 11:12:38,155 DEBUG [actor] received message ReviveOffers from Actor[akka://sparkDriver/
2015-05-15 11:12:38,155 DEBUG [actor] handled message (0.017 ms) ReviveOffers from Actor[akka://s
2015-05-15 11:12:39,153 DEBUG [actor] received message ReviveOffers from Actor[akka://sparkDriver/
2015-05-15 11:12:39,153 DEBUG [actor] handled message (0.018 ms) ReviveOffers from Actor[akka://s
2015-05-15 11:12:40,154 DEBUG [actor] received message ReviveOffers from Actor[akka://sparkDriver/
2015-05-15 11:12:40,155 DEBUG [actor] handled message (0.02 ms) ReviveOffers from Actor[akka://sp
2015-05-15 11:12:41,153 DEBUG [actor] received message ReviveOffers from Actor[akka://sparkDriver/
2015-05-15 11:12:41,153 DEBUG [actor] handled message (0.016 ms) ReviveOffers from Actor[akka://s
2015-05-15 11:12:42,154 DEBUG [actor] received message ReviveOffers from Actor[akka://sparkDriver/
2015-05-15 11:12:42,155 DEBUG [actor] handled message (0.015 ms) ReviveOffers from Actor[akka://s
2015-05-15 11:12:43,155 DEBUG [actor] received message ReviveOffers from Actor[akka://sparkDriver/
2015-05-15 11:12:43,155 DEBUG [actor] handled message (0.018 ms) ReviveOffers from Actor[akka://s
2015-05-15 11:12:44,154 DEBUG [actor] received message ReviveOffers from Actor[akka://sparkDriver/
2015-05-15 11:12:44,154 DEBUG [actor] handled message (0.015 ms) ReviveOffers from Actor[akka://s
2015-05-15 11:12:45,152 DEBUG [actor] received message ReviveOffers from Actor[akka://sparkDriver/
2015-05-15 11:12:45,152 DEBUG [actor] handled message (0.02 ms) ReviveOffers from Actor[akka://sp
2015-05-15 11:12:46,153 DEBUG [actor] received message ReviveOffers from Actor[akka://sparkDriver/
2015-05-15 11:12:46,153 DEBUG [actor] handled message (0.017 ms) ReviveOffers from Actor[akka://s
2015-05-15 11:12:46,604 DEBUG org.apache.hadoop.ipc.Client: IPC Client (1198532806) connection to
2015-05-15 11:12:46,604 DEBUG org.apache.hadoop.ipc.Client: IPC Client (1198532806) connection to
2015-05-15 11:12:47,153 DEBUG [actor] received message ReviveOffers from Actor[akka://sparkDriver/
2015-05-15 11:12:47,153 DEBUG [actor] handled message (0.014 ms) ReviveOffers from Actor[akka://s
```



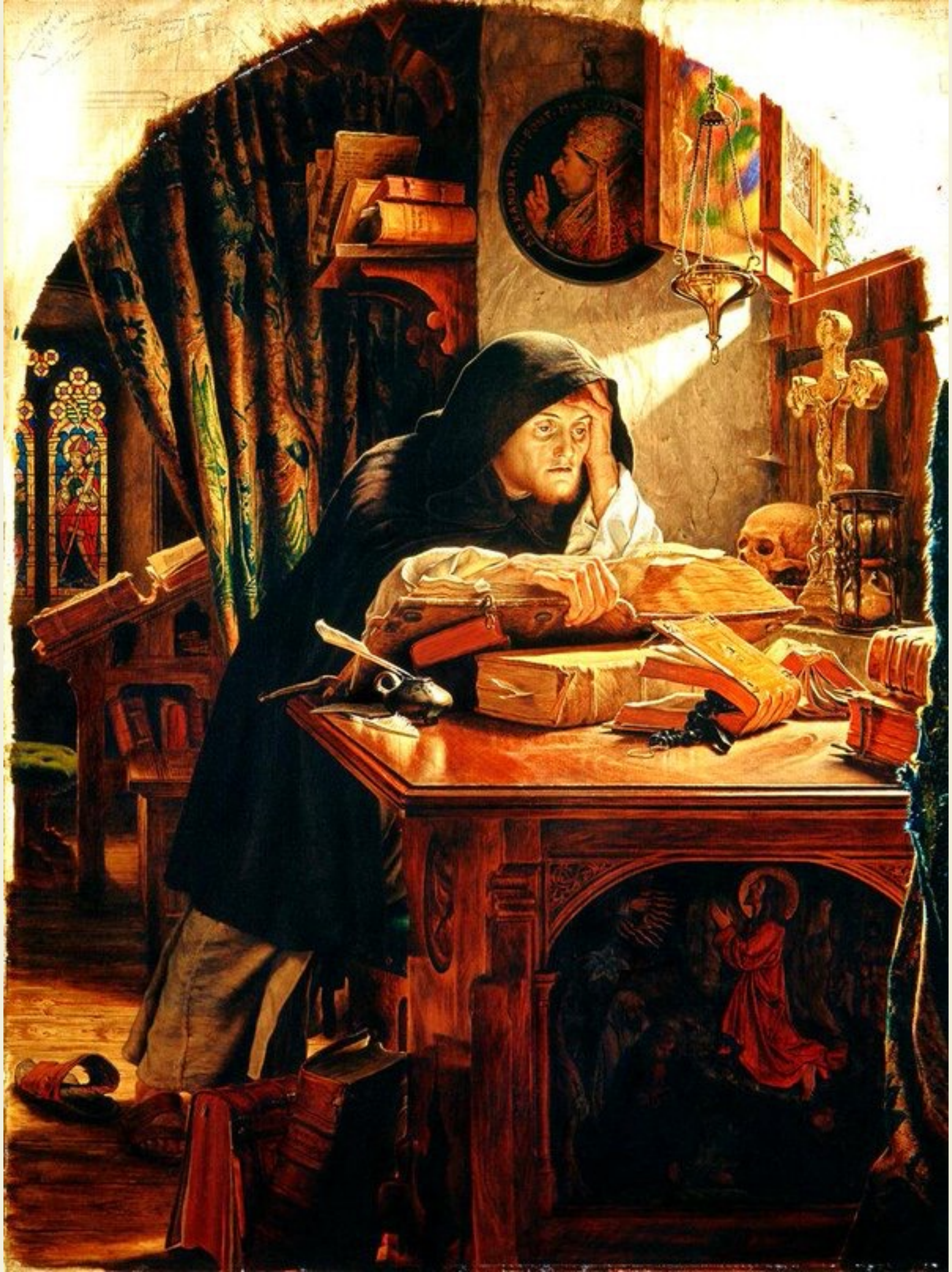
```
2015-05-15 11:13:04,155 DEBUG org.apache.spark.scheduler.cluster.SparkDeploySchedulerBackend: [actor] recei
2015-05-15 11:13:04,155 DEBUG org.apache.spark.scheduler.cluster.SparkDeploySchedulerBackend: [actor] hand
2015-05-15 11:13:05,155 DEBUG org.apache.spark.scheduler.cluster.SparkDeploySchedulerBackend: [actor] recei
2015-05-15 11:13:05,155 DEBUG org.apache.spark.scheduler.cluster.SparkDeploySchedulerBackend: [actor] hand
2015-05-15 11:13:06,155 DEBUG org.apache.spark.scheduler.cluster.SparkDeploySchedulerBackend: [actor] recei
2015-05-15 11:13:06,155 DEBUG org.apache.spark.scheduler.cluster.SparkDeploySchedulerBackend: [actor] hand
2015-05-15 11:13:06,693 DEBUG org.apache.hadoop.ipc.Client: The ping interval is 60000 ms.
2015-05-15 11:13:06,694 DEBUG org.apache.hadoop.ipc.Client: Connecting to localhost/127.0.0.1:9000
2015-05-15 11:13:06,695 DEBUG org.apache.hadoop.ipc.Client: IPC Client (1198532806) connection to localhost
2015-05-15 11:13:06,695 DEBUG org.apache.hadoop.ipc.Client: IPC Client (1198532806) connection to localhost
2015-05-15 11:13:06,696 DEBUG org.apache.hadoop.ipc.Client: IPC Client (1198532806) connection to localhost
2015-05-15 11:13:06,697 DEBUG org.apache.hadoop.ipc.ProtobufRpcEngine: Call: renewLease took 4ms
2015-05-15 11:13:06,697 DEBUG org.apache.hadoop.hdfs.LeaseRenewer: Lease renewed for client DFSClient_NO
2015-05-15 11:13:06,697 DEBUG org.apache.hadoop.hdfs.LeaseRenewer: Lease renewer daemon for [DFSClient_
2015-05-15 11:13:06,903 DEBUG org.apache.spark.util.Utils: Shutdown hook called
2015-05-15 11:13:06,903 DEBUG org.apache.spark.storage.DiskBlockManager: Shutdown hook called
2015-05-15 11:13:06,908 ERROR org.apache.hadoop.hdfs.DFSClient: Failed to close inode 16409
ava.io.IOException: All datanodes 127.0.0.1:50010 are bad. Aborting...
        at org.apache.hadoop.hdfs.DFSOutputStream$DataStreamer.setupPipelineForAppendOrRecovery(DFSOutput
        at org.apache.hadoop.hdfs.DFSOutputStream$DataStreamer.processDatanodeError(DFSOutputStream.java:1
        at org.apache.hadoop.hdfs.DFSOutputStream$DataStreamer.run(DFSOutputStream.java:548)
2015-05-15 11:13:06,912 DEBUG org.apache.hadoop.ipc.Client: stopping client from cache: org.apache.hadoop.i
2015-05-15 11:13:06,913 DEBUG org.apache.hadoop.ipc.Client: removing client from cache: org.apache.hadoop.
2015-05-15 11:13:06,913 DEBUG org.apache.hadoop.ipc.Client: stopping actual client because no more referenc
2015-05-15 11:13:06,913 DEBUG org.apache.hadoop.ipc.Client: Stopping client
2015-05-15 11:13:06,914 DEBUG org.apache.hadoop.ipc.Client: IPC Client (1198532806) connection to localhost
2015-05-15 11:13:06,914 DEBUG org.apache.hadoop.ipc.Client: IPC Client (1198532806) connection to localhost
2015-05-15 11:13:06,931 DEBUG org.apache.spark.repl.SparkILoop$SparkILoopInterpreter: parse("       sc.stop()
2015-05-15 11:13:07,153 DEBUG org.apache.spark.scheduler.cluster.SparkDeploySchedulerBackend: [actor] recei
2015-05-15 11:13:07,153 DEBUG org.apache.spark.scheduler.cluster.SparkDeploySchedulerBackend: [actor] hand
```



```
2015-05-15 11:13:04,155 DEBUG org.apache.spark.scheduler.cluster.SparkDeploySchedulerBackend: [actor] recei
2015-05-15 11:13:04,155 DEBUG org.apache.spark.scheduler.cluster.SparkDeploySchedulerBackend: [actor] hand
2015-05-15 11:13:05,155 DEBUG org.apache.spark.scheduler.cluster.SparkDeploySchedulerBackend: [actor] recei
2015-05-15 11:13:05,155 DEBUG org.apache.spark.scheduler.cluster.SparkDeploySchedulerBackend: [actor] hand
2015-05-15 11:13:06,155 DEBUG org.apache.spark.scheduler.cluster.SparkDeploySchedulerBackend: [actor] recei
2015-05-15 11:13:06,155 DEBUG org.apache.spark.scheduler.cluster.SparkDeploySchedulerBackend: [actor] hand
2015-05-15 11:13:06,693 DEBUG org.apache.hadoop.ipc.Client: The ping interval is 60000 ms.
2015-05-15 11:13:06,694 DEBUG org.apache.hadoop.ipc.Client: Connecting to localhost/127.0.0.1:9000
2015-05-15 11:13:06,695 DEBUG org.apache.hadoop.ipc.Client: IPC Client (1198532806) connection to localhost
2015-05-15 11:13:06,695 DEBUG org.apache.hadoop.ipc.Client: IPC Client (1198532806) connection to localhost
2015-05-15 11:13:06,696 DEBUG org.apache.hadoop.ipc.Client: IPC Client (1198532806) connection to localhost
2015-05-15 11:13:06,697 DEBUG org.apache.hadoop.ipc.ProtobufRpcEngine: Call: renewLease took 4ms
2015-05-15 11:13:06,697 DEBUG org.apache.hadoop.hdfs.LeaseRenewer: Lease renewed for client DFSClient_NO
2015-05-15 11:13:06,697 DEBUG org.apache.hadoop.hdfs.LeaseRenewer: Lease renewer daemon for [DFSClient_
2015-05-15 11:13:06,903 DEBUG org.apache.spark.util.Utils: Shutdown hook called
2015-05-15 11:13:06,903 DEBUG org.apache.spark.storage.DiskBlockManager: Shutdown hook called
2015-05-15 11:13:06,908 ERROR org.apache.hadoop.hdfs.DFSClient: Failed to close inode 16409
ava.io.IOException: All datanodes 127.0.0.1:50010 are bad. Aborting...
	at org.apache.hadoop.hdfs.DFSOutputStream$DataStreamer.setupPipelineForAppendOrRecovery(DFSOutput
	at org.apache.hadoop.hdfs.DFSOutputStream$DataStreamer.processDatanodeError(DFSOutputStream.java:1
	at org.apache.hadoop.hdfs.DFSOutputStream$DataStreamer.run(DFSOutputStream.java:548)
2015-05-15 11:13:06,912 DEBUG org.apache.hadoop.ipc.Client: stopping client from cache: org.apache.hadoop.i
2015-05-15 11:13:06,913 DEBUG org.apache.hadoop.ipc.Client: removing client from cache: org.apache.hadoop.
2015-05-15 11:13:06,913 DEBUG org.apache.hadoop.ipc.Client: stopping actual client because no more referenc
2015-05-15 11:13:06,913 DEBUG org.apache.hadoop.ipc.Client: Stopping client
2015-05-15 11:13:06,914 DEBUG org.apache.hadoop.ipc.Client: IPC Client (1198532806) connection to localhost
2015-05-15 11:13:06,914 DEBUG org.apache.hadoop.ipc.Client: IPC Client (1198532806) connection to localhost
2015-05-15 11:13:06,931 DEBUG org.apache.spark.repl.SparkILoop$SparkILoopInterpreter: parse("       sc.stop()
2015-05-15 11:13:07,153 DEBUG org.apache.spark.scheduler.cluster.SparkDeploySchedulerBackend: [actor] recei
2015-05-15 11:13:07,153 DEBUG org.apache.spark.scheduler.cluster.SparkDeploySchedulerBackend: [actor] hand
```



```
2015-05-15 11:13:04,155 DEBUG org.apache.spark.scheduler.cluster.SparkDeploySchedulerBackend: [actor] recei
2015-05-15 11:13:04,155 DEBUG org.apache.spark.scheduler.cluster.SparkDeploySchedulerBackend: [actor] hand
2015-05-15 11:13:05,155 DEBUG org.apache.spark.scheduler.cluster.SparkDeploySchedulerBackend: [actor] rece
2015-05-15 11:13:05,155 DEBUG org.apache.spark.scheduler.cluster.SparkDeploySchedulerBackend: [actor] hand
2015-05-15 11:13:06,155 DEBUG org.apache.spark.scheduler.cluster.SparkDeploySchedulerBackend: [actor] recei
2015-05-15 11:13:06,155 DEBUG org.apache.spark.scheduler.cluster.SparkDeploySchedulerBackend: [actor] hand
2015-05-15 11:13:06,693 DEBUG org.apache.hadoop.ipc.Client: The ping interval is 60000 ms.
2015-05-15 11:13:06,694 DEBUG org.apache.hadoop.ipc.Client: Connecting to localhost/127.0.0.1:9000
2015-05-15 11:13:06,695 DEBUG org.apache.hadoop.ipc.Client: IPC Client (1198532806) connection to localhost
2015-05-15 11:13:06,695 DEBUG org.apache.hadoop.ipc.Client: IPC Client (1198532806) connection to localhost
2015-05-15 11:13:06,696 DEBUG org.apache.hadoop.ipc.Client: IPC Client (1198532806) connection to localhost
2015-05-15 11:13:06,697 DEBUG org.apache.hadoop.ipc.ProtobufRpcEngine: Call: renewLease took 4ms
2015-05-15 11:13:06,697 DEBUG org.apache.hadoop.hdfs.LeaseRenewer: Lease renewed for client DFSClient_NO
2015-05-15 11:13:06,697 DEBUG org.apache.hadoop.hdfs.LeaseRenewer: Lease renewer daemon for [DFSClient_
2015-05-15 11:13:06,903 DEBUG org.apache.spark.util.Utils: Shutdown hook called
2015-05-15 11:13:06,903 DEBUG org.apache.spark.storage.DiskBlockManager: Shutdown hook called
2015-05-15 11:13:06,908 ERROR org.apache.hadoop.hdfs.DFSClient: Failed to close inode 16409
ava.io.IOException: All datanodes 127.0.0.1:50010 are bad. Aborting...
        at org.apache.hadoop.hdfs.DFSOutputStream$DataStreamer.setupPipelineForAppendOrRecovery(DFSOutput
        at org.apache.hadoop.hdfs.DFSOutputStream$DataStreamer.processDatanodeError(DFSOutputStream.java:1
        at org.apache.hadoop.hdfs.DFSOutputStream$DataStreamer.run(DFSOutputStream.java:548)
2015-05-15 11:13:06,912 DEBUG org.apache.hadoop.ipc.Client: stopping client from cache: org.apache.hadoop.i
2015-05-15 11:13:06,913 DEBUG org.apache.hadoop.ipc.Client: removing client from cache: org.apache.hadoop.
2015-05-15 11:13:06,913 DEBUG org.apache.hadoop.ipc.Client: stopping actual client because no more referenc
2015-05-15 11:13:06,913 DEBUG org.apache.hadoop.ipc.Client: Stopping client
2015-05-15 11:13:06,914 DEBUG org.apache.hadoop.ipc.Client: IPC Client (1198532806) connection to localhost
2015-05-15 11:13:06,914 DEBUG org.apache.hadoop.ipc.Client: IPC Client (1198532806) connection to localhost
2015-05-15 11:13:06,931 DEBUG org.apache.spark.repl.SparkILoop$SparkILoopInterpreter: parse("       sc.stop()
2015-05-15 11:13:07,153 DEBUG org.apache.spark.scheduler.cluster.SparkDeploySchedulerBackend: [actor] recei
2015-05-15 11:13:07,153 DEBUG org.apache.spark.scheduler.cluster.SparkDeploySchedulerBackend: [actor] hand
```



“Developer looking at production logs after a regression with downtime”

Sir Joseph Noel Paton, Oil on Canvas, 1861

Trace Checking

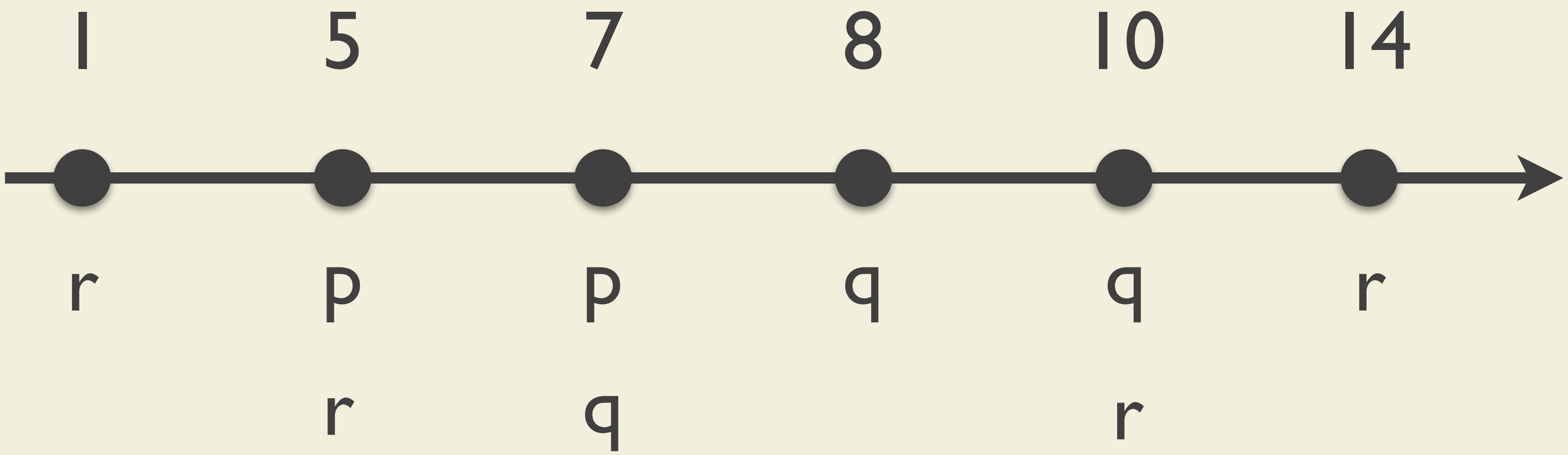
“Automatic procedure for evaluating a formal specification over a trace of recorded events produced by a system”

How do we specify properties to check?

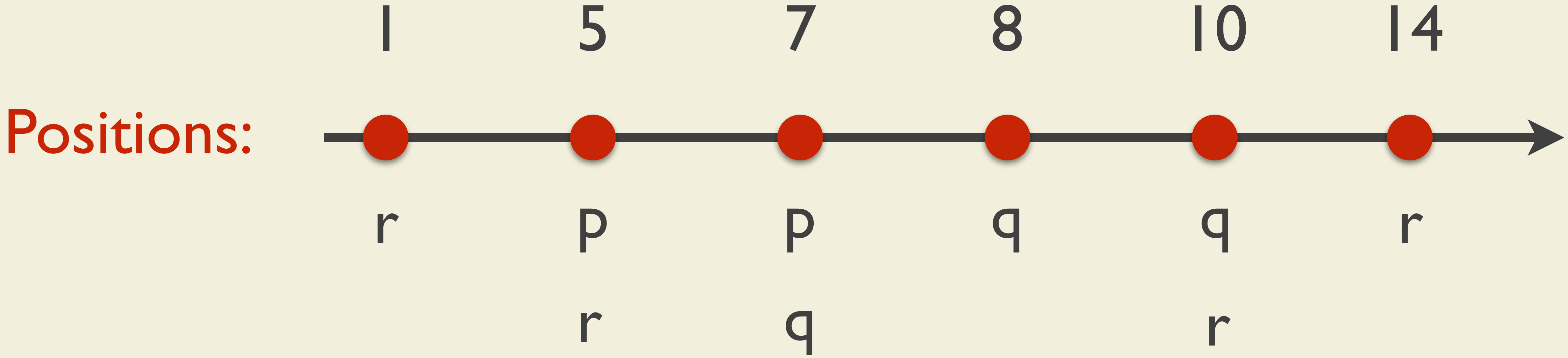
Metric Temporal Logic

$$\phi ::= p \mid \neg\phi \mid \phi \vee \phi \mid \phi \mathsf{U}_{\mathsf{I}} \phi$$

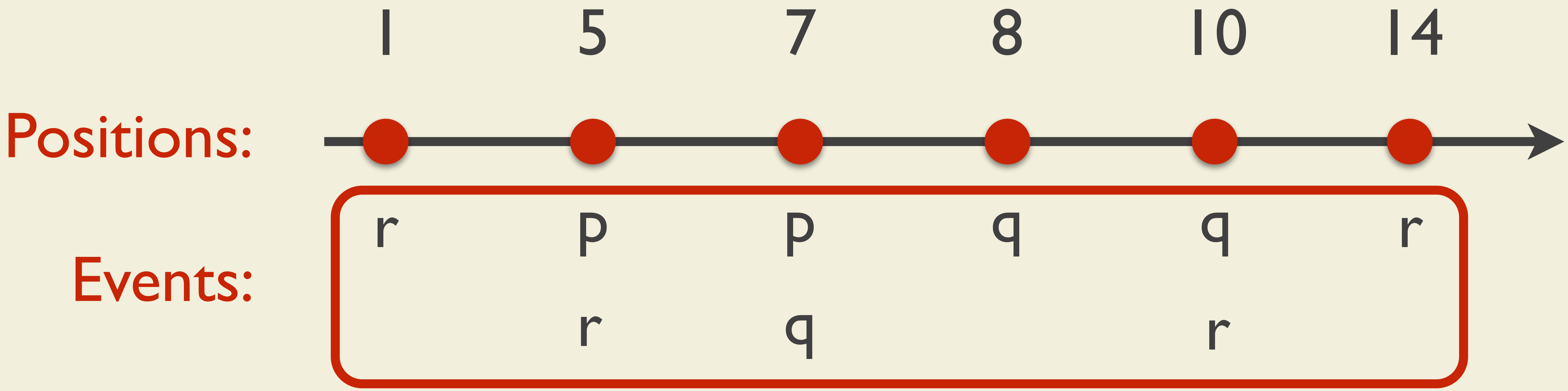
Model: Timed Word



Model: Timed Word

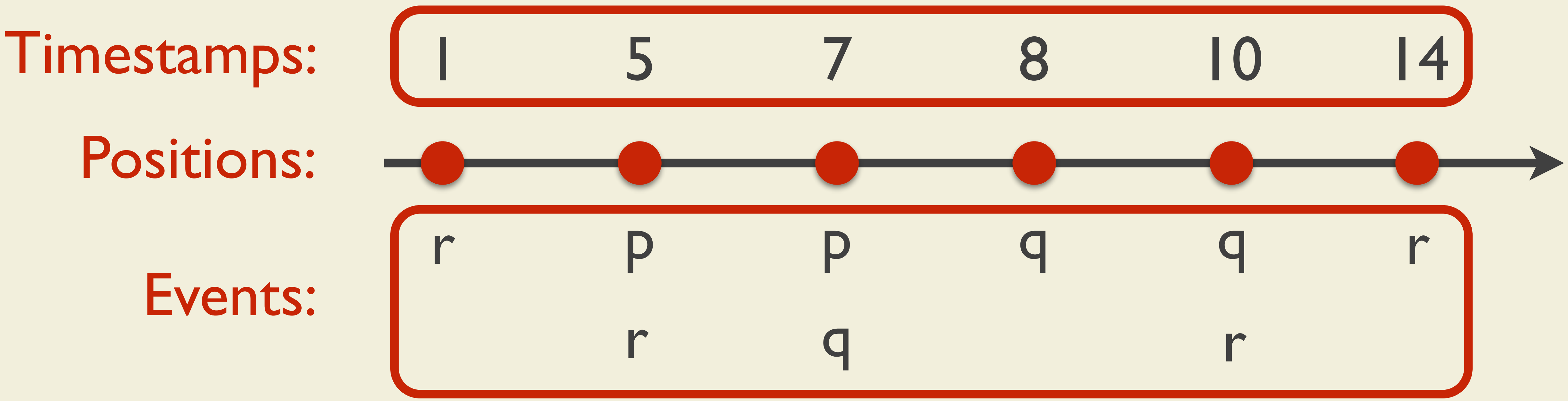


Model: Timed Word



	1	5	7	8	10	14
Positions:	●	●	●	●	●	●
Events:	r	p r	p q	q	q r	r

Model: Timed Word



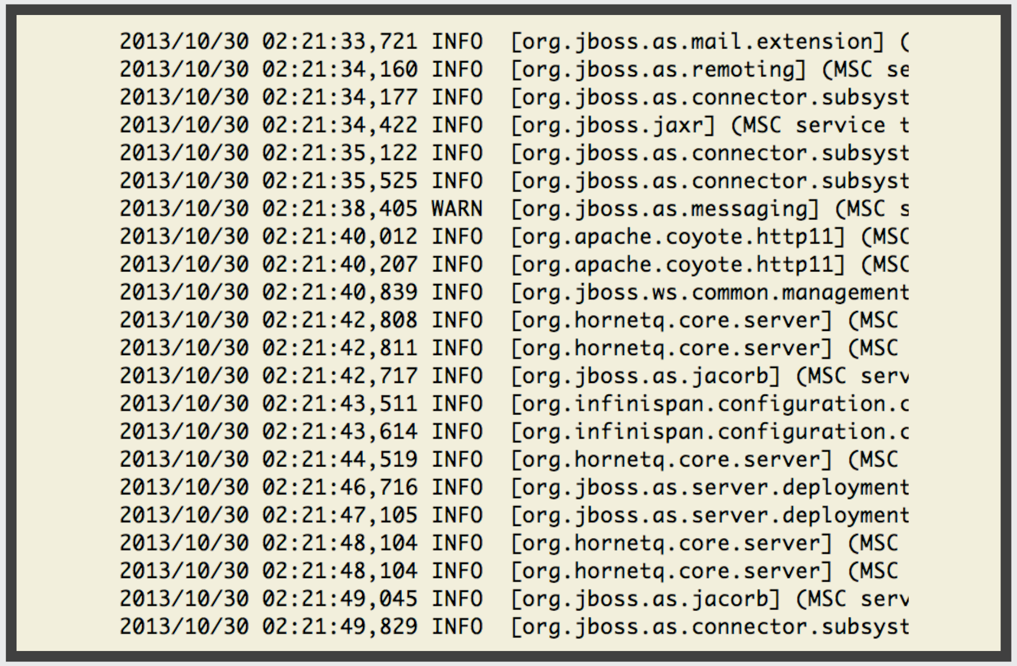
Timestamps:	1	5	7	8	10	14
Positions:	●	●	●	●	●	●
Events:	r	p r	p q	q	q r	r

MTL Semantics

$$\mathsf{F}_I(\phi)$$

"Formula ϕ holds eventually in the future within a time window I"

MTL Trace Checking: Challenges



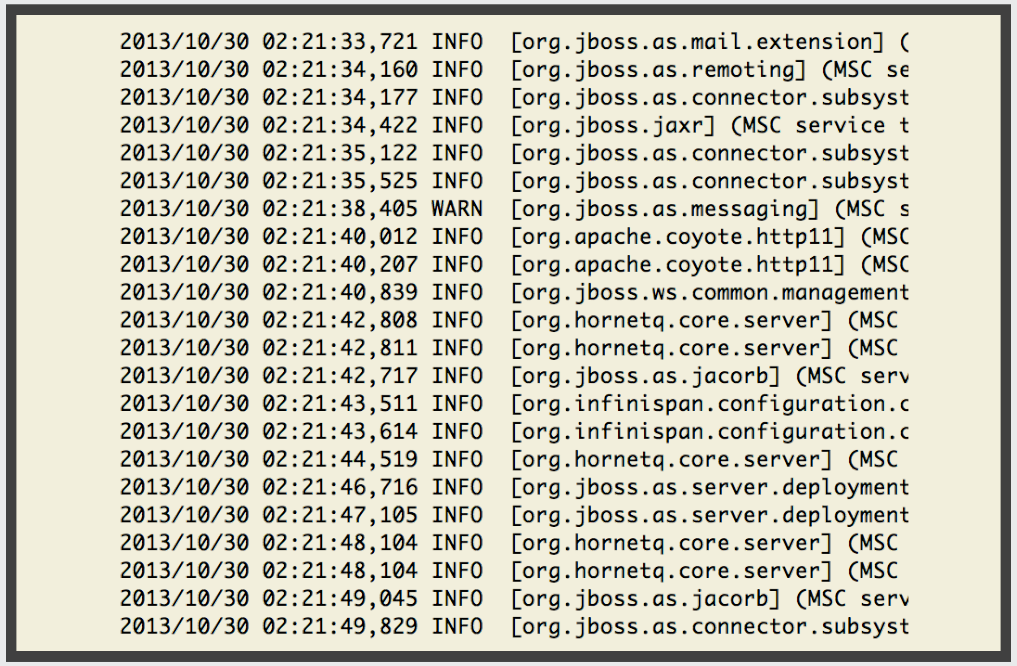
```
2013/10/30 02:21:33,721 INFO  [org.jboss.as.mail.extension] (
2013/10/30 02:21:34,160 INFO  [org.jboss.as.remoting] (MSC se
2013/10/30 02:21:34,177 INFO  [org.jboss.as.connector.subsyst
2013/10/30 02:21:34,422 INFO  [org.jboss.jaxr] (MSC service t
2013/10/30 02:21:35,122 INFO  [org.jboss.as.connector.subsyst
2013/10/30 02:21:35,525 INFO  [org.jboss.as.connector.subsyst
2013/10/30 02:21:38,405 WARN  [org.jboss.as.messaging] (MSC s
2013/10/30 02:21:40,012 INFO  [org.apache.coyote.http11] (MSC
2013/10/30 02:21:40,207 INFO  [org.apache.coyote.http11] (MSC
2013/10/30 02:21:40,839 INFO  [org.jboss.ws.common.management
2013/10/30 02:21:42,808 INFO  [org.hornetq.core.server] (MSC
2013/10/30 02:21:42,811 INFO  [org.hornetq.core.server] (MSC
2013/10/30 02:21:42,717 INFO  [org.jboss.as.jacorb] (MSC serv
2013/10/30 02:21:43,511 INFO  [org.infinispan.configuration.c
2013/10/30 02:21:43,614 INFO  [org.infinispan.configuration.c
2013/10/30 02:21:44,519 INFO  [org.hornetq.core.server] (MSC
2013/10/30 02:21:46,716 INFO  [org.jboss.as.server.deployment
2013/10/30 02:21:47,105 INFO  [org.jboss.as.server.deployment
2013/10/30 02:21:48,104 INFO  [org.hornetq.core.server] (MSC
2013/10/30 02:21:48,104 INFO  [org.hornetq.core.server] (MSC
2013/10/30 02:21:49,045 INFO  [org.jboss.as.jacorb] (MSC serv
2013/10/30 02:21:49,829 INFO  [org.jboss.as.connector.subsyst
```

Scalability with respect to the size of the trace

$$\neg\ \mathsf{F}_{(0,3333]}(\phi) \wedge \mathsf{G}_{(0,5000]}(\psi) \rightarrow \mathsf{F}\ \phi \mathsf{U}_{(0,105000]}(\varphi)$$

Scalability with respect to the size of the timing intervals in the formula

MTL Trace Checking: Challenges



```
2013/10/30 02:21:33,721 INFO  [org.jboss.as.mail.extension] (
2013/10/30 02:21:34,160 INFO  [org.jboss.as.remoting] (MSC se
2013/10/30 02:21:34,177 INFO  [org.jboss.as.connector.subsyst
2013/10/30 02:21:34,422 INFO  [org.jboss.jaxr] (MSC service t
2013/10/30 02:21:35,122 INFO  [org.jboss.as.connector.subsyst
2013/10/30 02:21:35,525 INFO  [org.jboss.as.connector.subsyst
2013/10/30 02:21:38,405 WARN  [org.jboss.as.messaging] (MSC s
2013/10/30 02:21:40,012 INFO  [org.apache.coyote.http11] (MSC
2013/10/30 02:21:40,207 INFO  [org.apache.coyote.http11] (MSC
2013/10/30 02:21:40,839 INFO  [org.jboss.ws.common.management
2013/10/30 02:21:42,808 INFO  [org.hornetq.core.server] (MSC
2013/10/30 02:21:42,811 INFO  [org.hornetq.core.server] (MSC
2013/10/30 02:21:42,717 INFO  [org.jboss.as.jacorb] (MSC serv
2013/10/30 02:21:43,511 INFO  [org.infinispan.configuration.c
2013/10/30 02:21:43,614 INFO  [org.infinispan.configuration.c
2013/10/30 02:21:44,519 INFO  [org.hornetq.core.server] (MSC
2013/10/30 02:21:46,716 INFO  [org.jboss.as.server.deployment
2013/10/30 02:21:47,105 INFO  [org.jboss.as.server.deployment
2013/10/30 02:21:48,104 INFO  [org.hornetq.core.server] (MSC
2013/10/30 02:21:48,104 INFO  [org.hornetq.core.server] (MSC
2013/10/30 02:21:49,045 INFO  [org.jboss.as.jacorb] (MSC serv
2013/10/30 02:21:49,829 INFO  [org.jboss.as.connector.subsyst
```

Scalability with respect to the size of the trace

Wikipedia Page Traffic Statistics Dataset

Contains 7 months of hourly page view statistics for all articles in Wikipedia

Size: 320 GB
Created On: June 9, 2009

DARPA Scalable Network Monitoring (SNM) Program Traffic

Contains 9 days of captured network traffic

Size: 7083.4 TB
Created On: November 12, 2009

Scalability with respect to the size of the trace

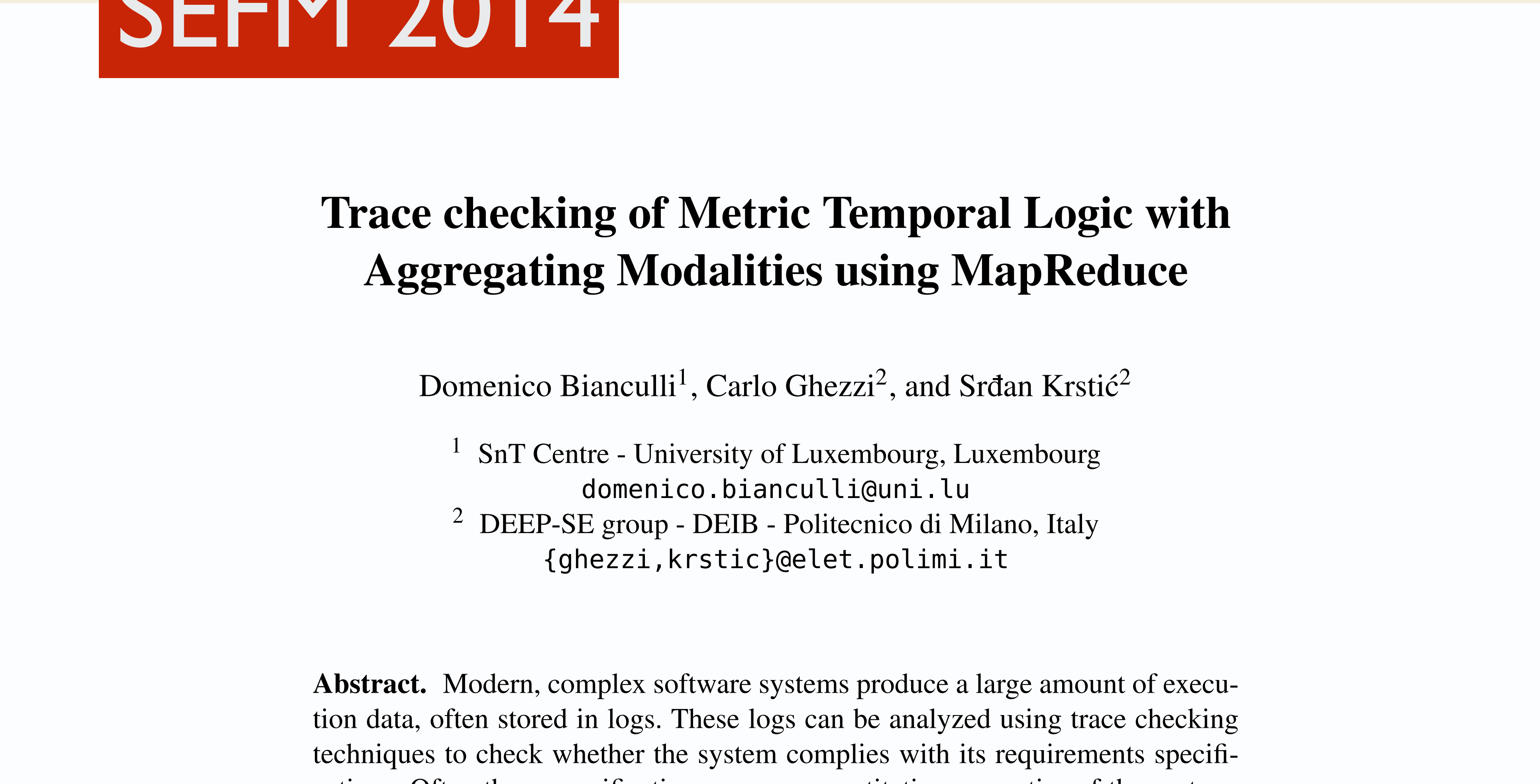
Scalability with respect to the size of the trace

Solution: Distributed Trace Checking

Scalability with respect to the size of the trace

Solution: Distributed Trace Checking

SEFM 2014



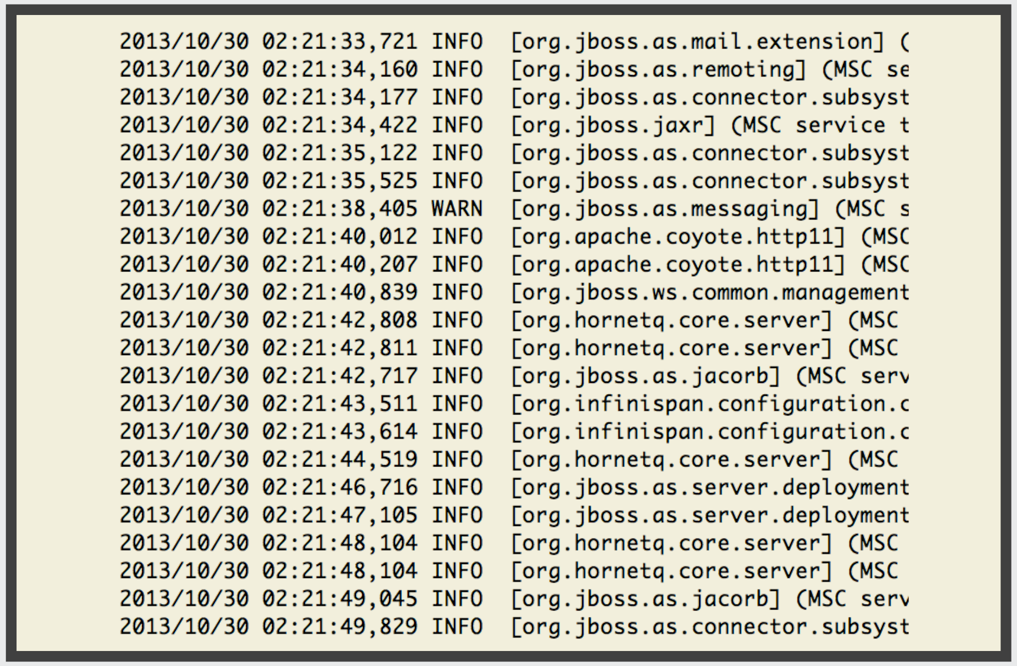
Trace checking of Metric Temporal Logic with Aggregating Modalities using MapReduce

Domenico Bianculli[1], Carlo Ghezzi[2], and Srđan Krstić[2]

[1] SnT Centre - University of Luxembourg, Luxembourg
domenico.bianculli@uni.lu
[2] DEEP-SE group - DEIB - Politecnico di Milano, Italy
{ghezzi,krstic}@elet.polimi.it

Abstract. Modern, complex software systems produce a large amount of execution data, often stored in logs. These logs can be analyzed using trace checking techniques to check whether the system complies with its requirements specifi-

MTL Trace Checking: Challenges



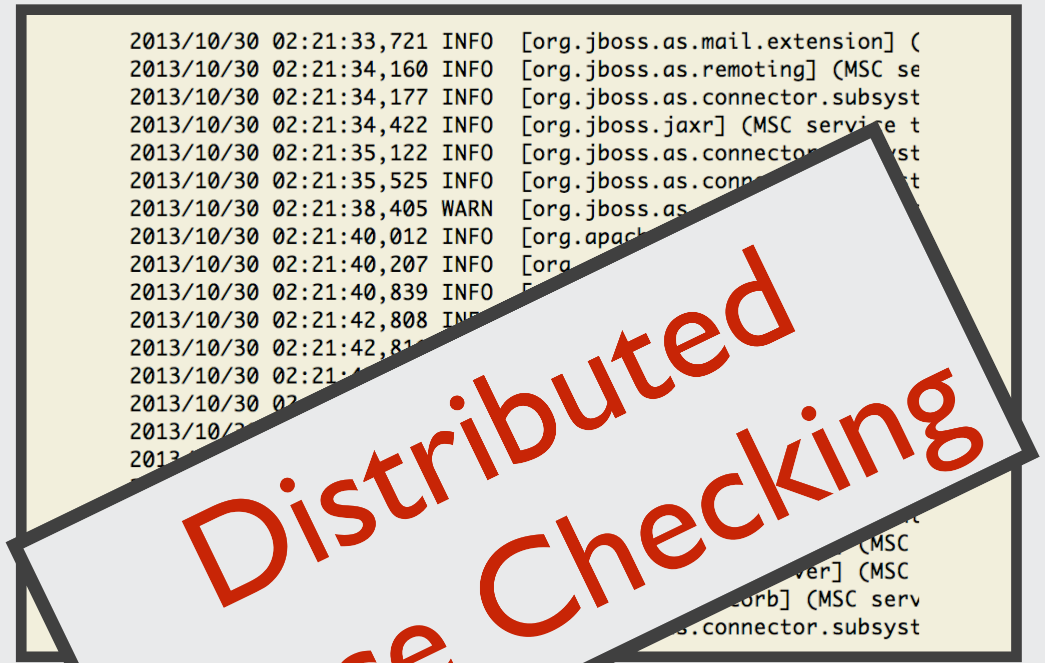
```
2013/10/30 02:21:33,721 INFO  [org.jboss.as.mail.extension] (
2013/10/30 02:21:34,160 INFO  [org.jboss.as.remoting] (MSC se
2013/10/30 02:21:34,177 INFO  [org.jboss.as.connector.subsyst
2013/10/30 02:21:34,422 INFO  [org.jboss.jaxr] (MSC service t
2013/10/30 02:21:35,122 INFO  [org.jboss.as.connector.subsyst
2013/10/30 02:21:35,525 INFO  [org.jboss.as.connector.subsyst
2013/10/30 02:21:38,405 WARN  [org.jboss.as.messaging] (MSC s
2013/10/30 02:21:40,012 INFO  [org.apache.coyote.http11] (MSC
2013/10/30 02:21:40,207 INFO  [org.apache.coyote.http11] (MSC
2013/10/30 02:21:40,839 INFO  [org.jboss.ws.common.management
2013/10/30 02:21:42,808 INFO  [org.hornetq.core.server] (MSC
2013/10/30 02:21:42,811 INFO  [org.hornetq.core.server] (MSC
2013/10/30 02:21:42,717 INFO  [org.jboss.as.jacorb] (MSC serv
2013/10/30 02:21:43,511 INFO  [org.infinispan.configuration.c
2013/10/30 02:21:43,614 INFO  [org.infinispan.configuration.c
2013/10/30 02:21:44,519 INFO  [org.hornetq.core.server] (MSC
2013/10/30 02:21:46,716 INFO  [org.jboss.as.server.deployment
2013/10/30 02:21:47,105 INFO  [org.jboss.as.server.deployment
2013/10/30 02:21:48,104 INFO  [org.hornetq.core.server] (MSC
2013/10/30 02:21:48,104 INFO  [org.hornetq.core.server] (MSC
2013/10/30 02:21:49,045 INFO  [org.jboss.as.jacorb] (MSC serv
2013/10/30 02:21:49,829 INFO  [org.jboss.as.connector.subsyst
```

Scalability with respect to the **size of the trace**

$$
\neg\ \mathsf{F}_{(0,3333]}(\phi) \wedge \\
\mathsf{G}_{(0,5000]}(\psi) \rightarrow \mathsf{F} \\
\phi \mathsf{U}_{(0,105000]}(\varphi)
$$

Scalability with respect to **the size of the timing intervals in the formula**

MTL Trace Checking: Challenges



```
2013/10/30 02:21:33,721 INFO  [org.jboss.as.mail.extension] (
2013/10/30 02:21:34,160 INFO  [org.jboss.as.remoting] (MSC se
2013/10/30 02:21:34,177 INFO  [org.jboss.as.connector.subsyst
2013/10/30 02:21:34,422 INFO  [org.jboss.jaxr] (MSC servi e t
2013/10/30 02:21:35,122 INFO  [org.jboss.as.connecto      yst
2013/10/30 02:21:35,525 INFO  [org.jboss.as.conn           t
2013/10/30 02:21:38,405 WARN  [org.jboss.as
2013/10/30 02:21:40,012 INFO  [org.apac
2013/10/30 02:21:40,207 INFO  [org
2013/10/30 02:21:40,839 INFO
2013/10/30 02:21:42,808 IN
2013/10/30 02:21:42,81
2013/10/30 02:21:
2013/10/30 0
2013/10/
2013
                                              (MSC
                                          ver] (MSC
                                       orb] (MSC serv
                                  s.connector.subsyst
```

Distributed Trace Checking

Scalability with respect to the size of the trace

$$\neg\, F_{(0,3333]}(\phi) \wedge G_{(0,5000]}(\psi) \rightarrow F\, \phi U_{(0,105000]}(\varphi)$$

Scalability with respect to the size of the timing intervals in the formula

MTL Trace Checking: Challenges

$$\begin{gathered}\neg\ \mathsf{F}_{(0,3333]}(\phi) \wedge \\ \mathsf{G}_{(0,5000]}(\psi) \rightarrow \mathsf{F} \\ \phi\mathsf{U}_{(0,105000]}(\varphi)\end{gathered}$$

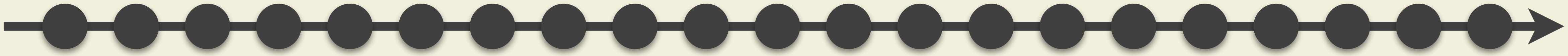
Scalability with respect to the size of the timing intervals in the formula

Health Insurance Portability and Accountability Act of 1996

"Retain the documentation [...] for **6 years** from the date of its creation or the date when it last was in effect, whichever is later"

Trace Checking Temporal operators

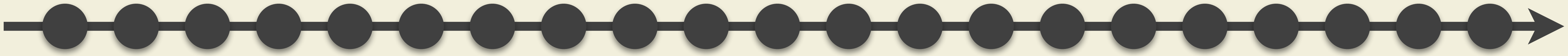
$\mathsf{F}_I(\phi)$



Trace Checking Temporal operators

Metric Eventually operator

$\mathsf{F}_I(\phi)$

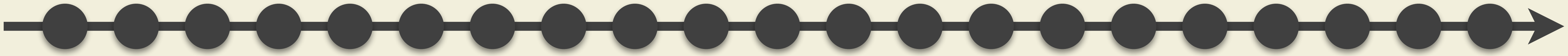


Trace Checking Temporal operators

Metric Eventually operator

Reverse scanning

$\mathsf{F}_I(\phi)$



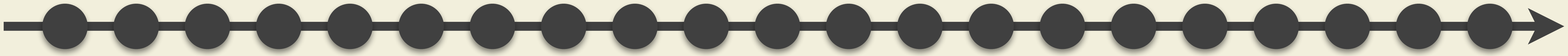
Trace Checking Temporal operators

Metric Eventually operator

Reverse scanning

Incremental verdict

$F_I(\phi)$



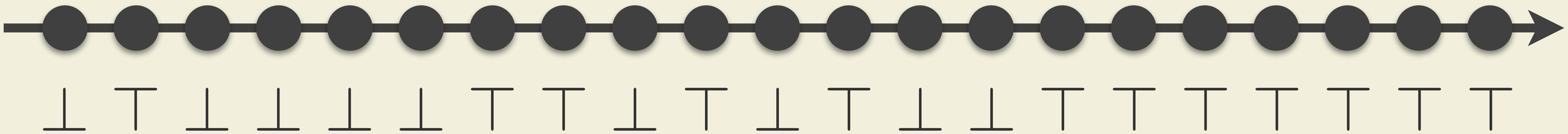
Trace Checking Temporal operators

Metric Eventually operator

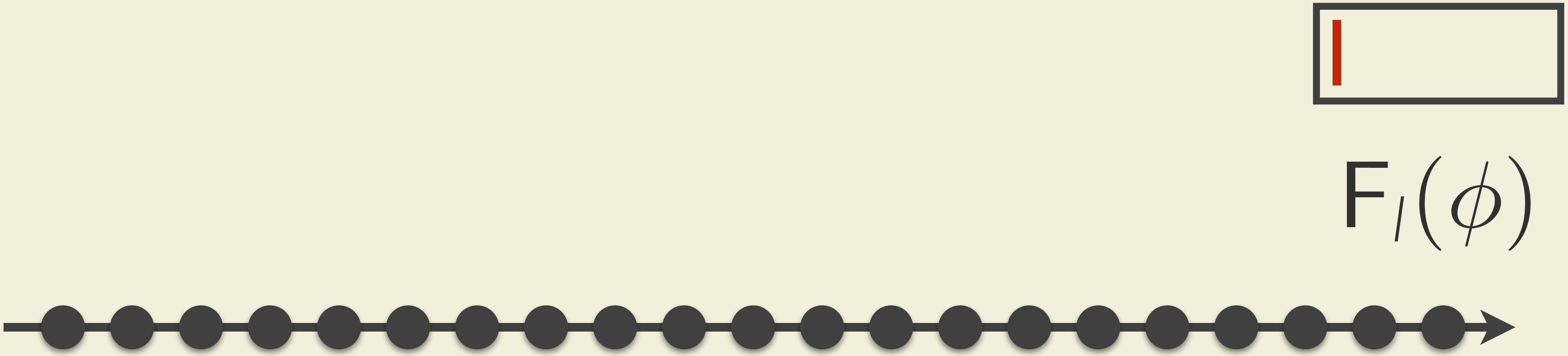
Reverse scanning

Incremental verdict

$$\mathsf{F}_I(\phi)$$



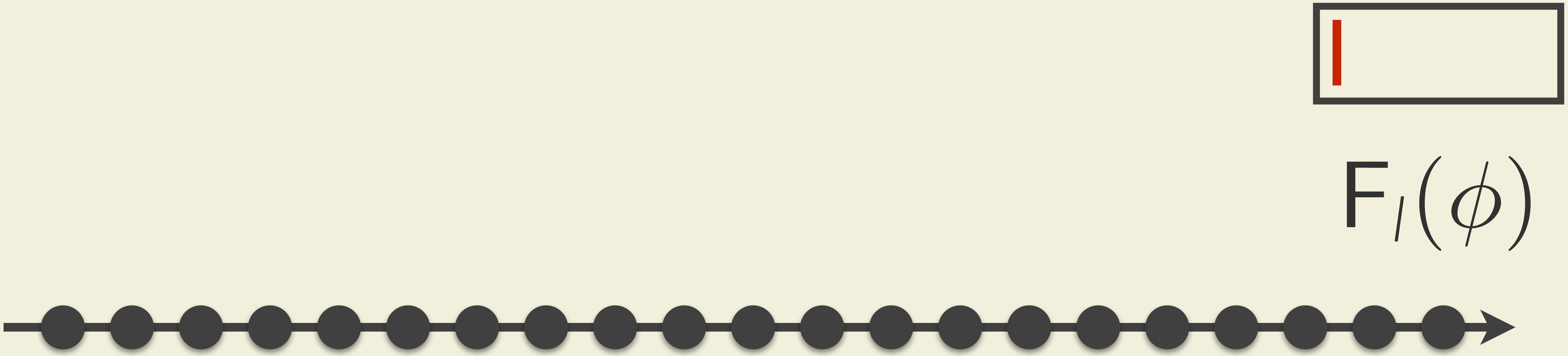
Trace Checking Temporal operators



$F_I(\phi)$

Trace Checking Temporal operators

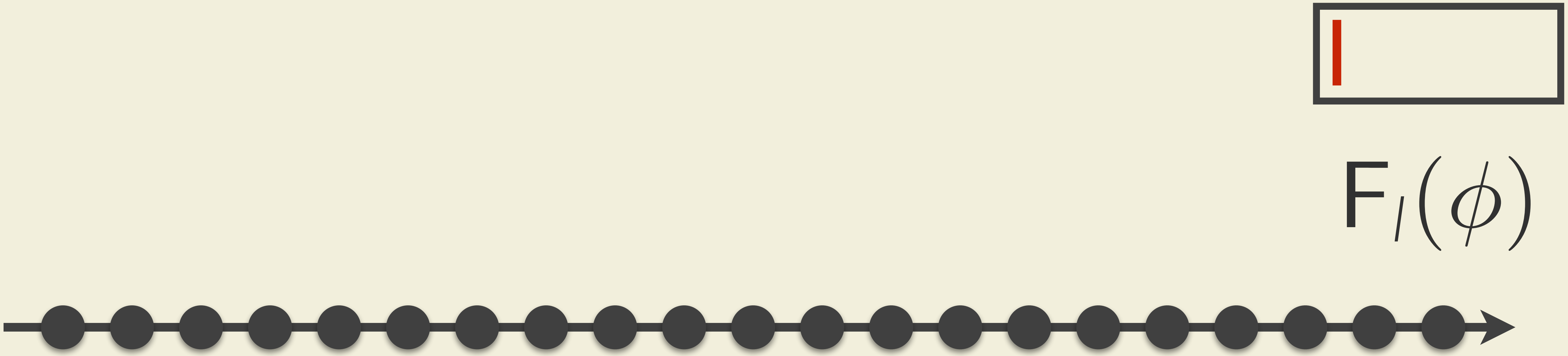
Queue-like data structure



Trace Checking Temporal operators

Queue-like data structure

Size of the temporal interval



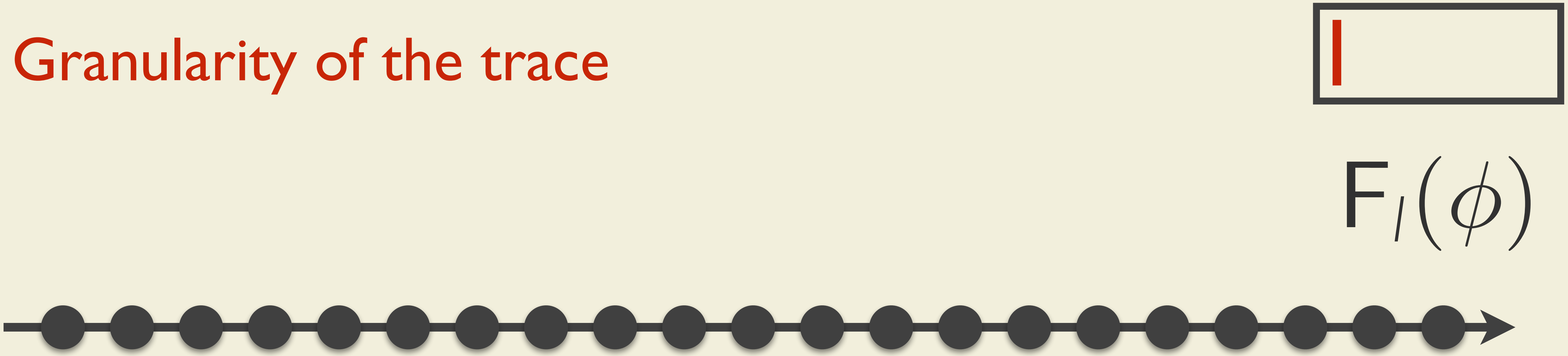
$F_I(\phi)$

Trace Checking Temporal operators

Queue-like data structure

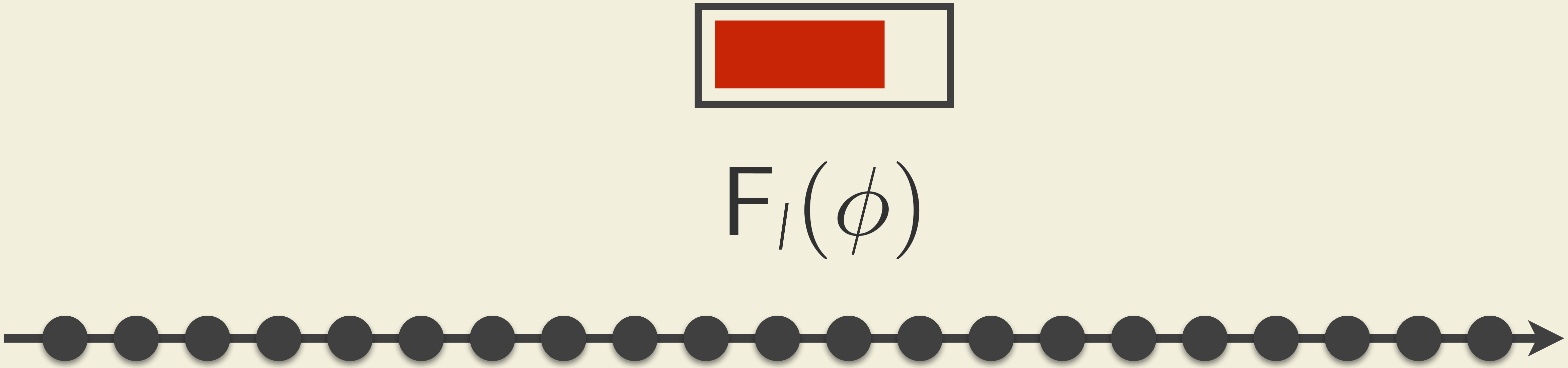
Size of the temporal interval

Granularity of the trace



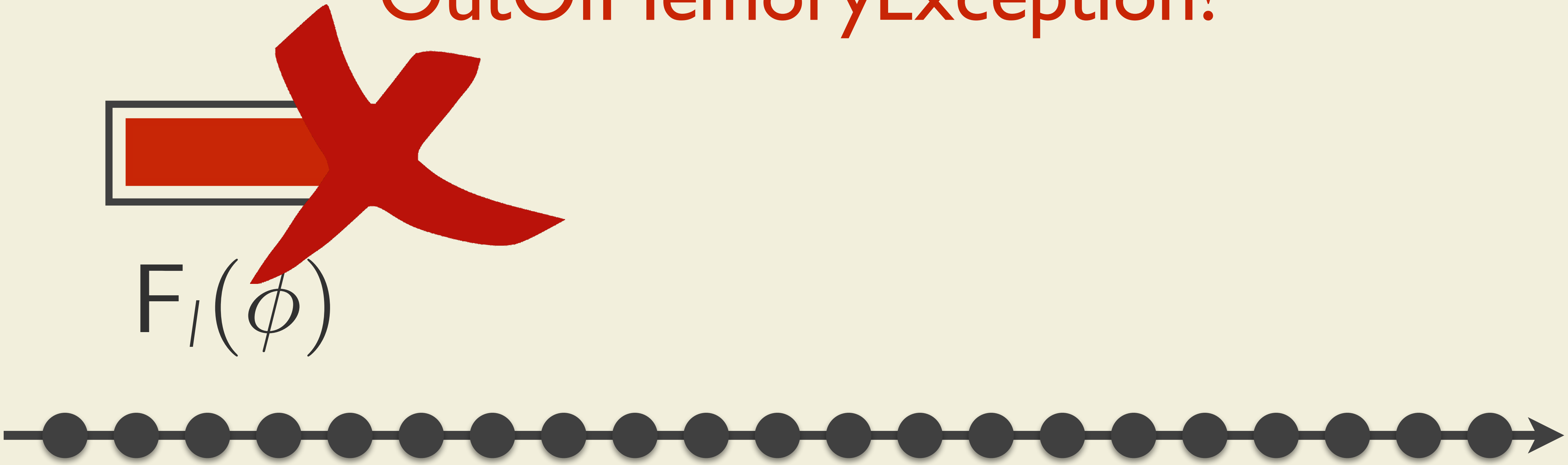
$\mathsf{F}_I(\phi)$

Trace Checking Temporal operators



$\mathsf{F}_I(\phi)$

Trace Checking Temporal operators



OutOfMemoryException!

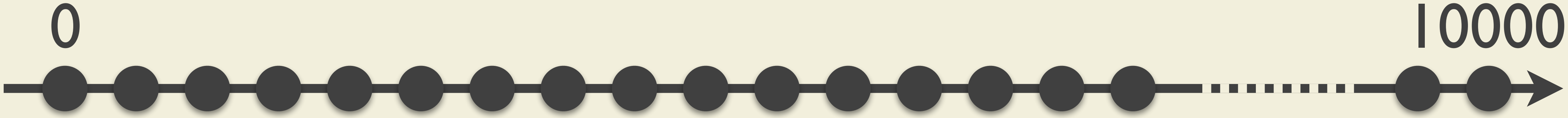
$F_l(\phi)$

Scalability with respect to the size of the temporal intervals

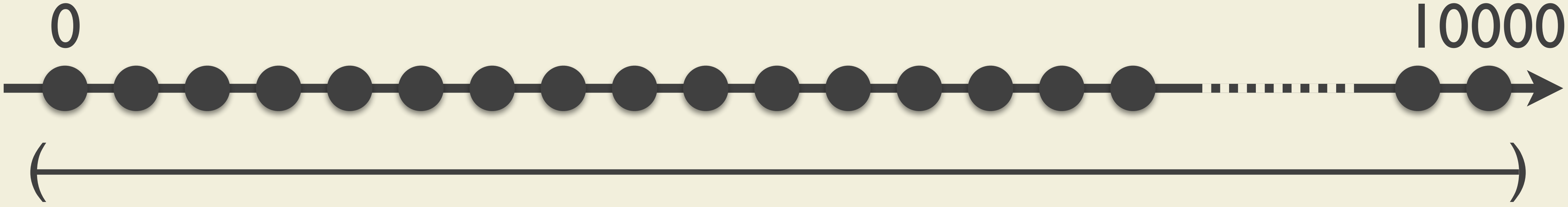
Scalability with respect to the size of the temporal intervals

Solution: Decomposing formulae with large intervals

Decomposition of temporal formulae

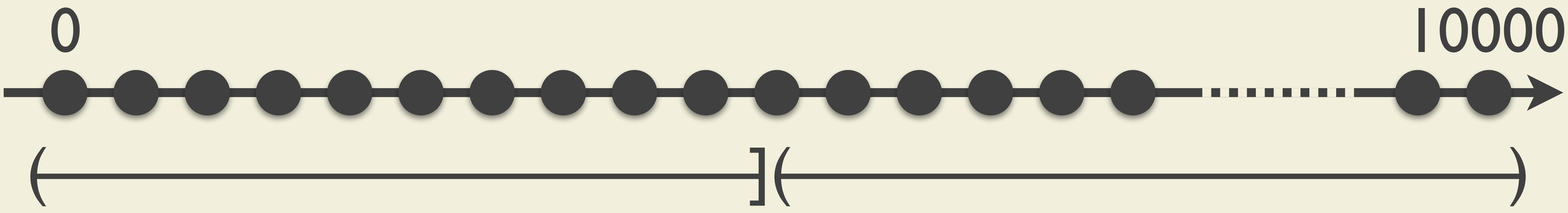


Decomposition of temporal formulae



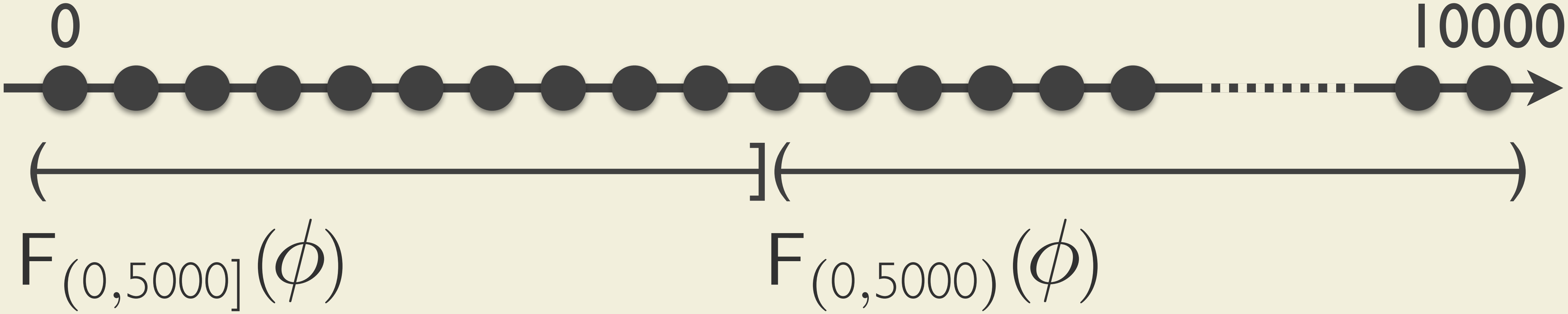
$F_{(0,10000]}(\phi)$

Decomposition of temporal formulae



$F_{(0,5000]}(\phi)$ $F_{(0,5000)}(\phi)$

Decomposition of temporal formulae



$$\mathsf{F}_{(0,10000)}(\phi) \equiv \mathsf{F}_{(0,5000]}(\phi) \vee \mathsf{F}_{=5000}(\mathsf{F}_{(0,5000)}(\phi))$$

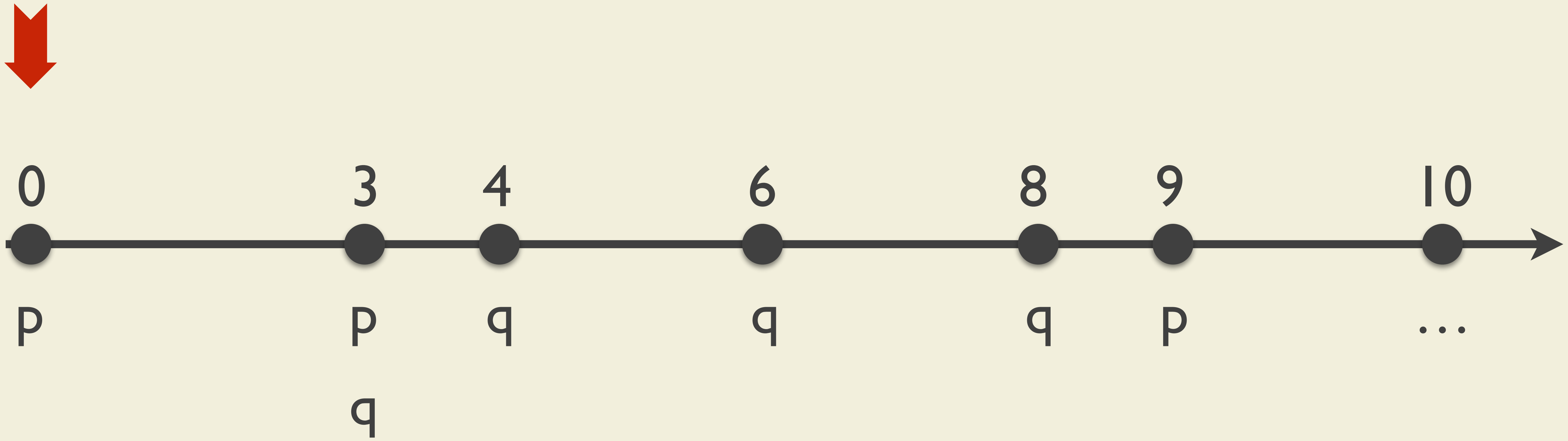
Decomposition of temporal formulae

$$F_{(0,10000)}(\phi) \equiv F_{(0,5000]}(\phi) \vee F_{=5000}(F_{(0,5000)}(\phi))$$

Equivalent?

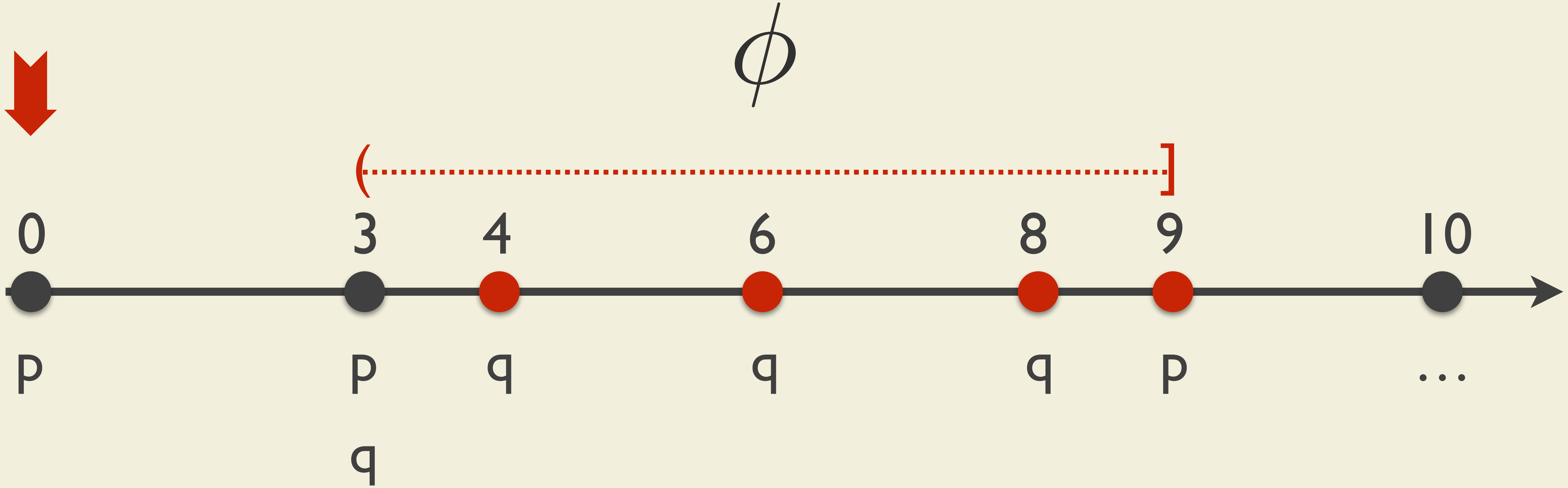
Point-based MTL Semantics

$$F_{(3,9]}(\phi)$$



Point-based MTL Semantics

$F_{(3,9]}(\phi)$



Point-based MTL Semantics

$$(\sigma, \tau, i) \models \mathsf{F}_I \phi \Leftrightarrow \exists i'.(i' \geq i \;\wedge \tau_{i'} - \tau_i \in I \wedge (\sigma, \tau, i') \models \phi)$$

Point-based semantics relies on
the explicit existence of a position for every formula

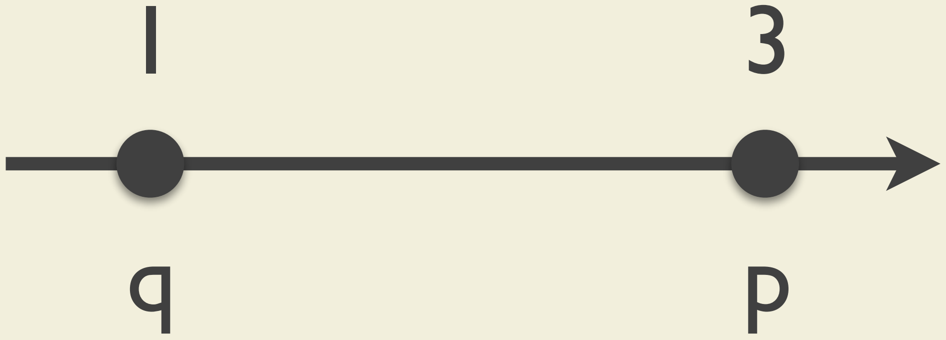
Point-based MTL Semantics

$$(\sigma, \tau, i) \models \mathsf{F}_I \phi \Leftrightarrow \boxed{\exists i'}.(i' \geq i \ \wedge \tau_{i'} - \tau_i \in I \wedge (\sigma, \tau, i') \models \phi)$$

Point-based semantics relies on
the explicit existence of a position for every formula

Point-based MTL Semantics

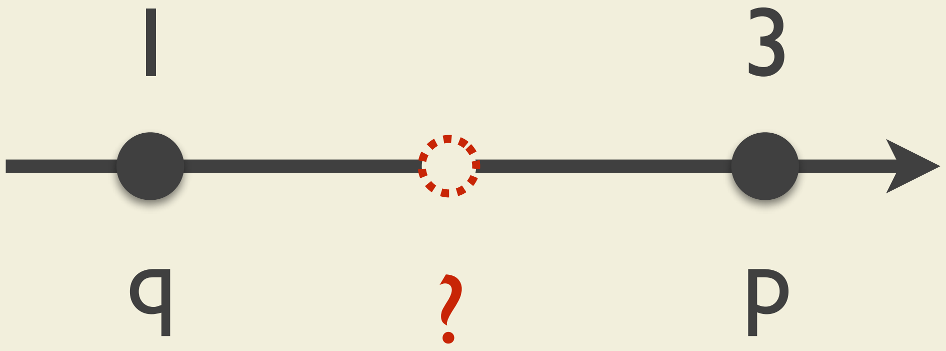
$\mathsf{F}_{=2}(p)$ $\mathsf{F}_{=1}\mathsf{F}_{=1}(p)$



Point-based MTL Semantics

$\mathsf{F}_{=2}(p)$ ✓

$\mathsf{F}_{=1}\mathsf{F}_{=1}(p)$ ✗

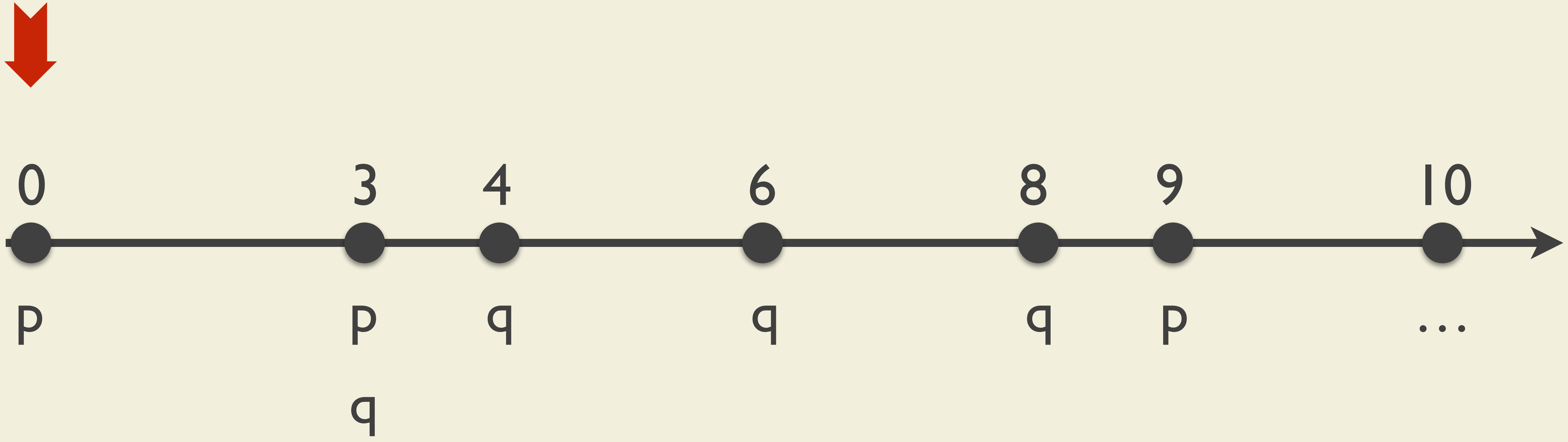


Our Proposal

Lazy MTL Semantics

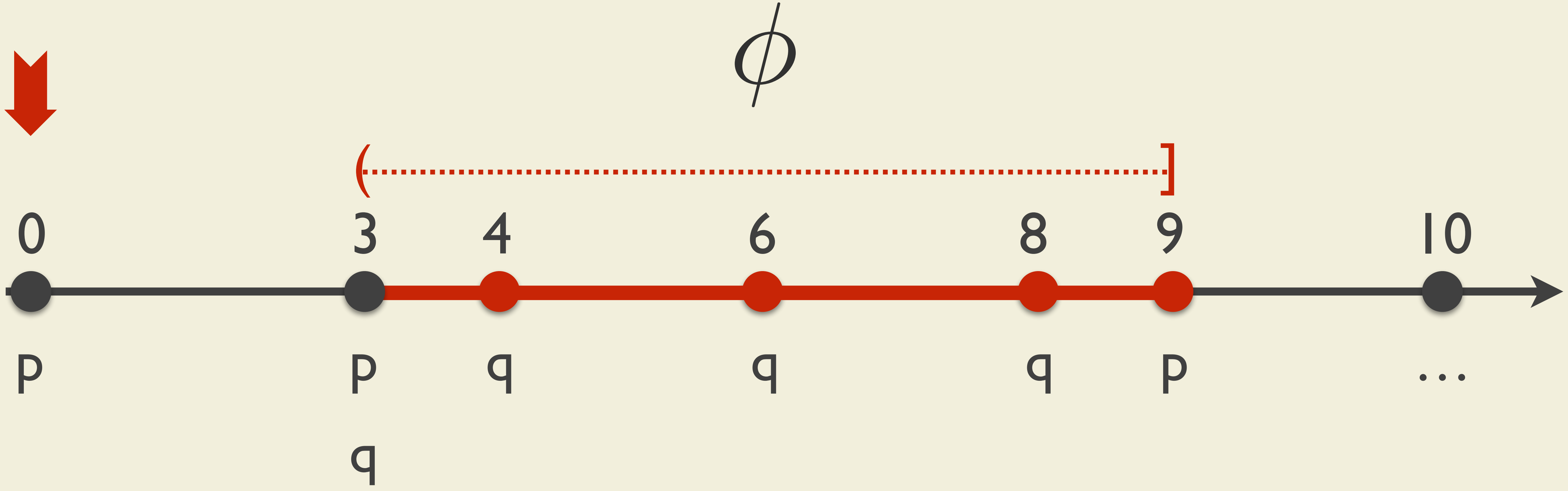
Lazy MTL Semantics

$$F_{(3,9]}(\phi)$$



Lazy MTL Semantics

$F_{(3,9]}(\phi)$



Lazy MTL Semantics

$$(\sigma, \tau, t) \models_L \mathsf{F}_I\phi \Leftrightarrow \exists t'.(t' \geq t \ \wedge t' - t \in I \wedge (\sigma, \tau, t') \models_L \phi)$$

Lazy semantics does not require the explicit existence of a position for temporal operators

Lazy MTL Semantics

$$(\sigma, \tau, t) \models_L \mathsf{F}_I \phi \Leftrightarrow \exists t'.(t' \geq t \; \wedge t' - t \in I \wedge (\sigma, \tau, t') \models_L \phi)$$

Lazy semantics does not require the explicit existence of a position for temporal operators

Lazy MTL Semantics

$$(\sigma, \tau, t) \models_L \mathsf{F}_I \phi \Leftrightarrow \exists t'.(t' \geq t \; \wedge t' - t \in I \wedge (\sigma, \tau, t') \models_L \phi)$$

Lazy semantics does not require the explicit existence of a position for temporal operators

(however, atomic propositions still require the explicit existence of a position)

Properties of Lazy MTL Semantics

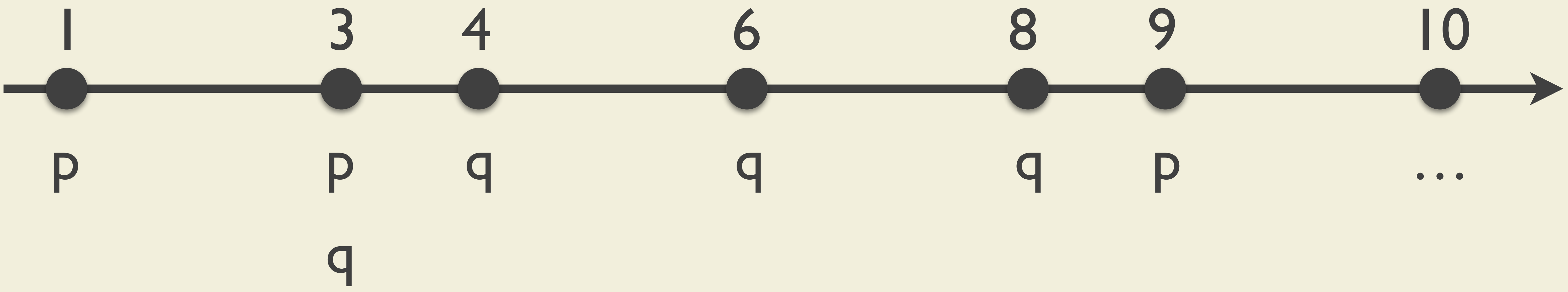
Properties of Lazy MTL Semantics

Nested intervals can be combined

Overlapping intervals can be combined

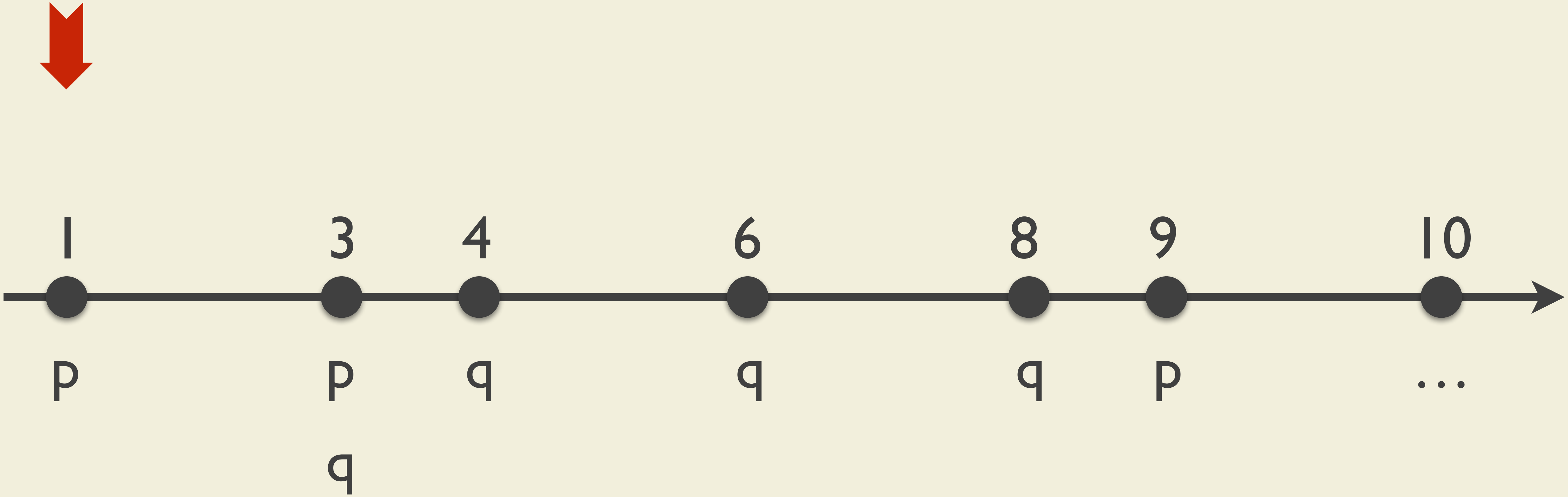
Strictly more expressive than point-based semantics

Combining the nested intervals



Combining the nested intervals

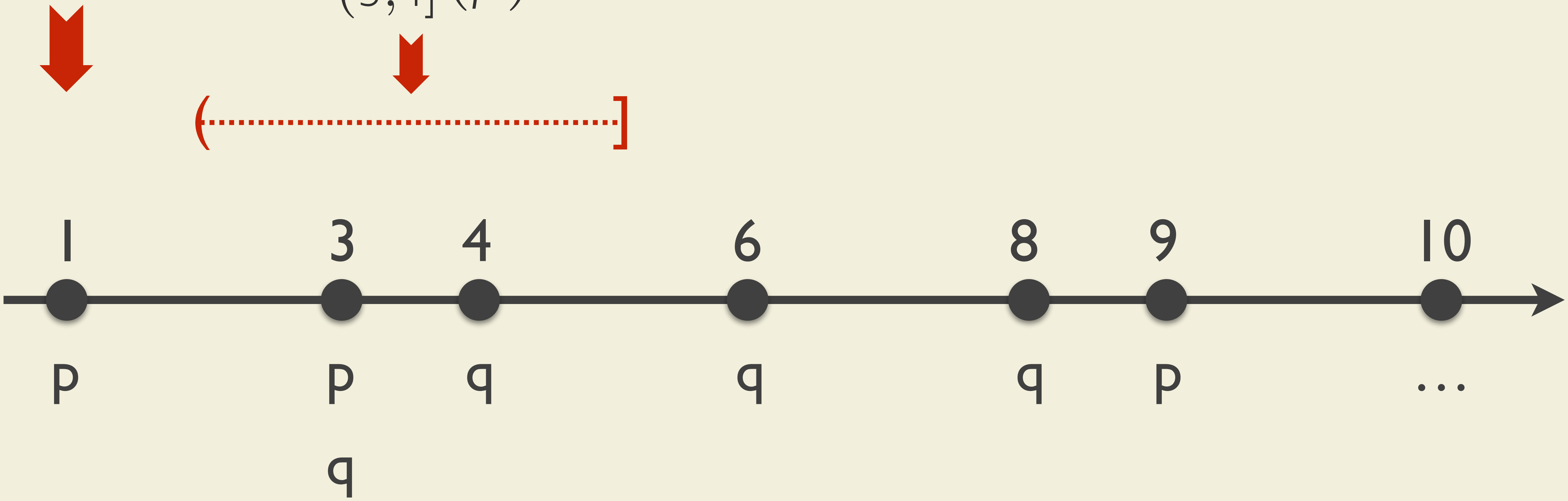
$\mathsf{F}_{(2,5]}\mathsf{F}_{(3,4]}(p)$



Combining the nested intervals

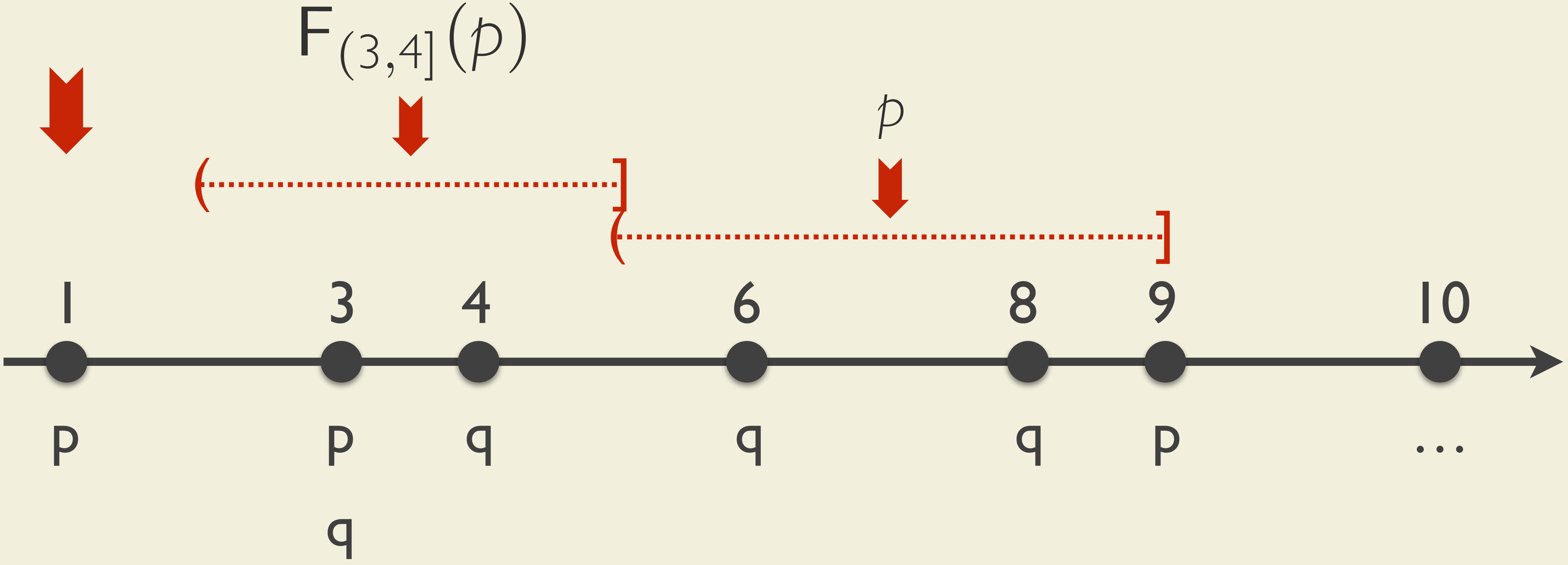
$F_{(2,5]} F_{(3,4]}(p)$

$F_{(3,4]}(p)$

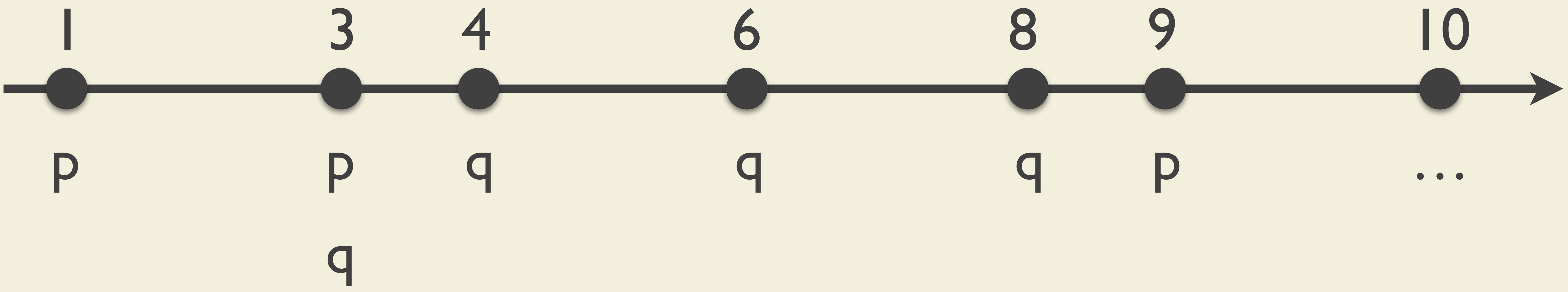


Combining the **nested** intervals

$$F_{(2,5]}F_{(3,4]}(p) \equiv F_{(5,9]}(p)$$

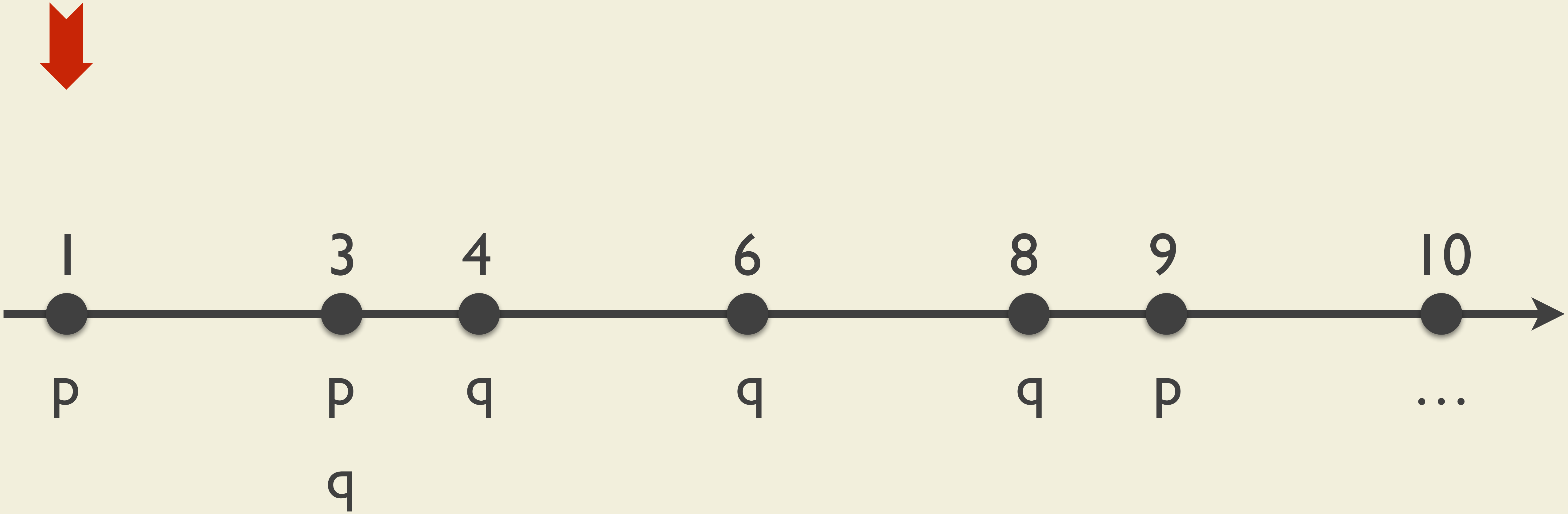


Combining overlapping intervals



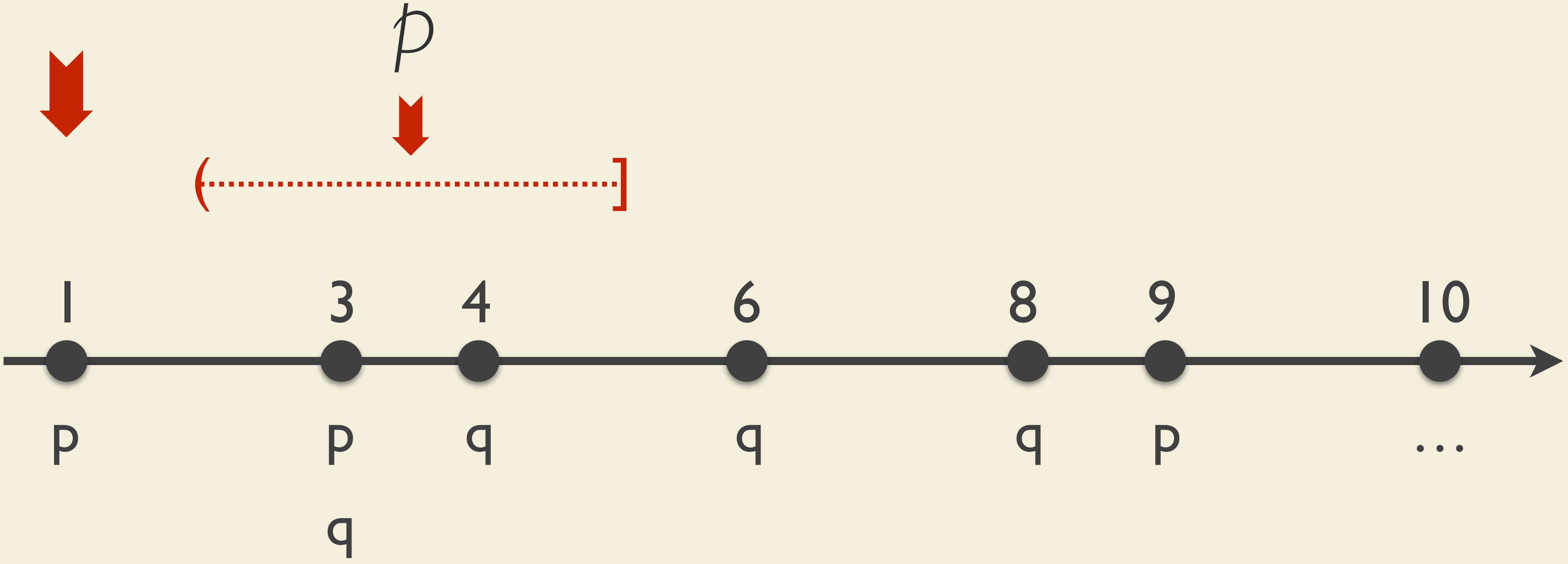
Combining overlapping intervals

$$F_{(2,5]}(p) \vee F_{(4,8]}(p)$$



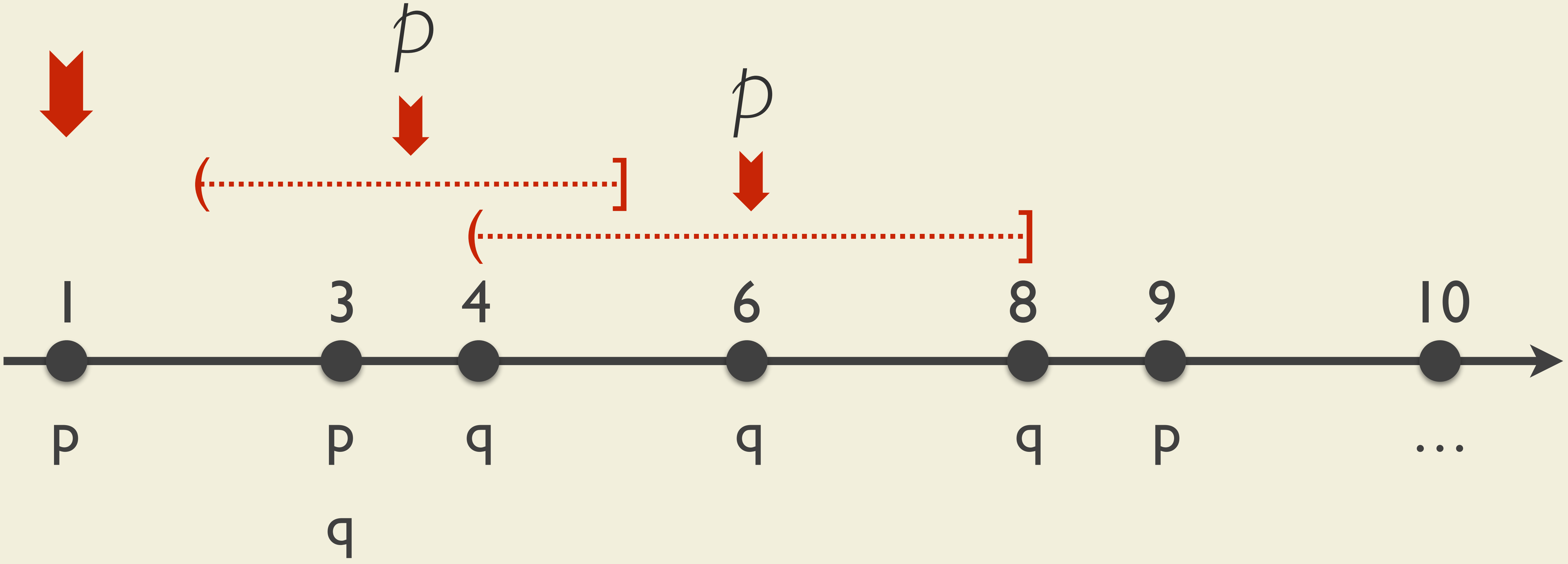
Combining overlapping intervals

$$F_{(2,5]}(p) \lor F_{(4,8]}(p)$$



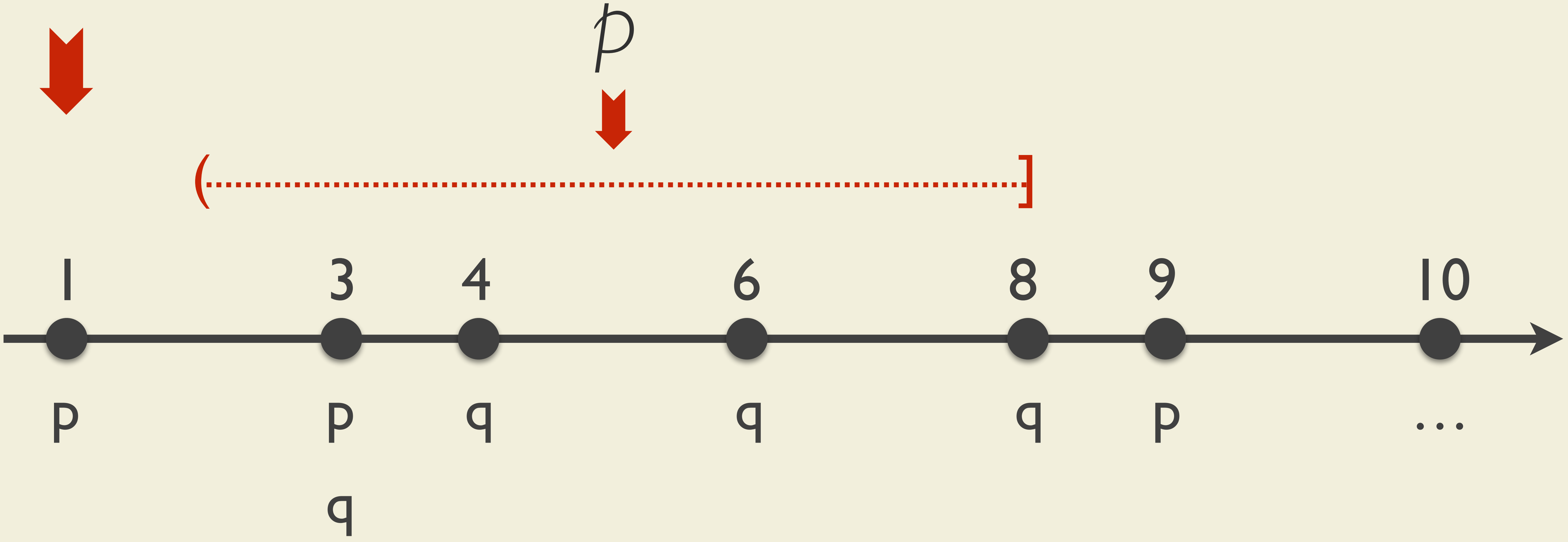
Combining overlapping intervals

$F_{(2,5]}(p) \vee F_{(4,8]}(p)$



Combining overlapping intervals

$$F_{(2,5]}(p) \vee F_{(4,8]}(p) \equiv F_{(2,8]}(p)$$



Decomposition of temporal formulae

$$\mathsf{F}_{(0,10000)}(\phi) \equiv \mathsf{F}_{(0,5000]}(\phi) \vee \mathsf{F}_{=5000}(\mathsf{F}_{(0,5000)}(\phi))$$

Decomposition of temporal formulae

$$\mathsf{F}_{(0,10000)}(\phi) \equiv \mathsf{F}_{(0,5000]}(\phi) \vee \overbrace{\mathsf{F}_{=5000}(\mathsf{F}_{(0,5000)}(\phi))}^{\text{Nested intervals}}$$

Decomposition of temporal formulae

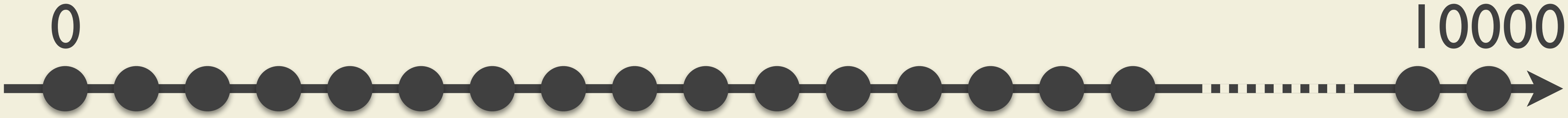
$$\mathsf{F}_{(0,10000)}(\phi) \equiv \underbrace{\mathsf{F}_{(0,5000]}(\phi) \vee \overbrace{\mathsf{F}_{=5000}(\mathsf{F}_{(0,5000)}(\phi))}^{\text{Nested intervals}}}_{\text{Overlapping intervals}}$$

Decomposition of temporal formulae

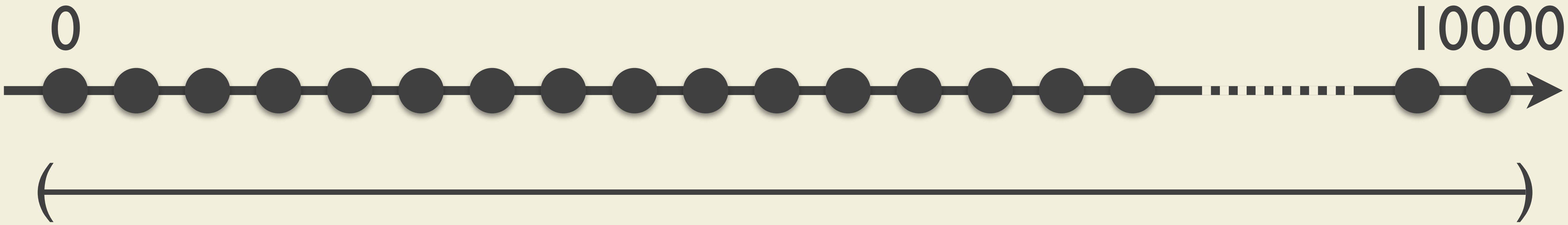
$$\mathsf{F}_{(0,10000)}(\phi) \equiv \underbrace{\mathsf{F}_{(0,5000]}(\phi) \vee \overbrace{\mathsf{F}_{=5000}(\mathsf{F}_{(0,5000)}(\phi))}^{\text{Nested intervals}}}_{\text{Overlapping intervals}}$$

✓

Parametric Decomposition



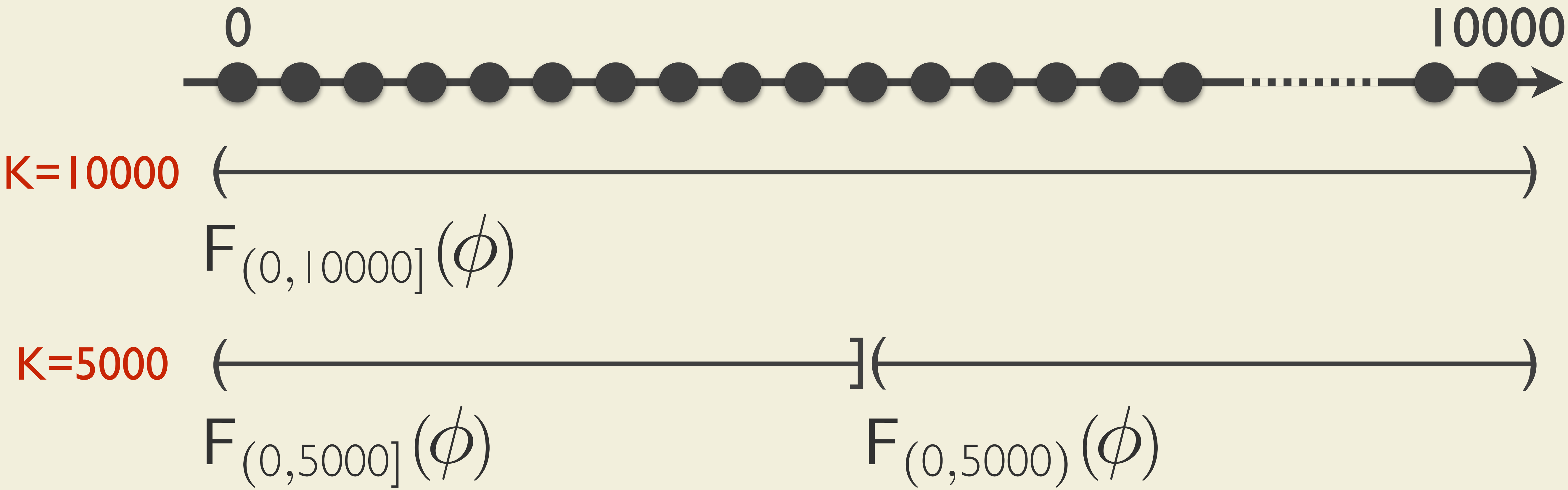
Parametric Decomposition



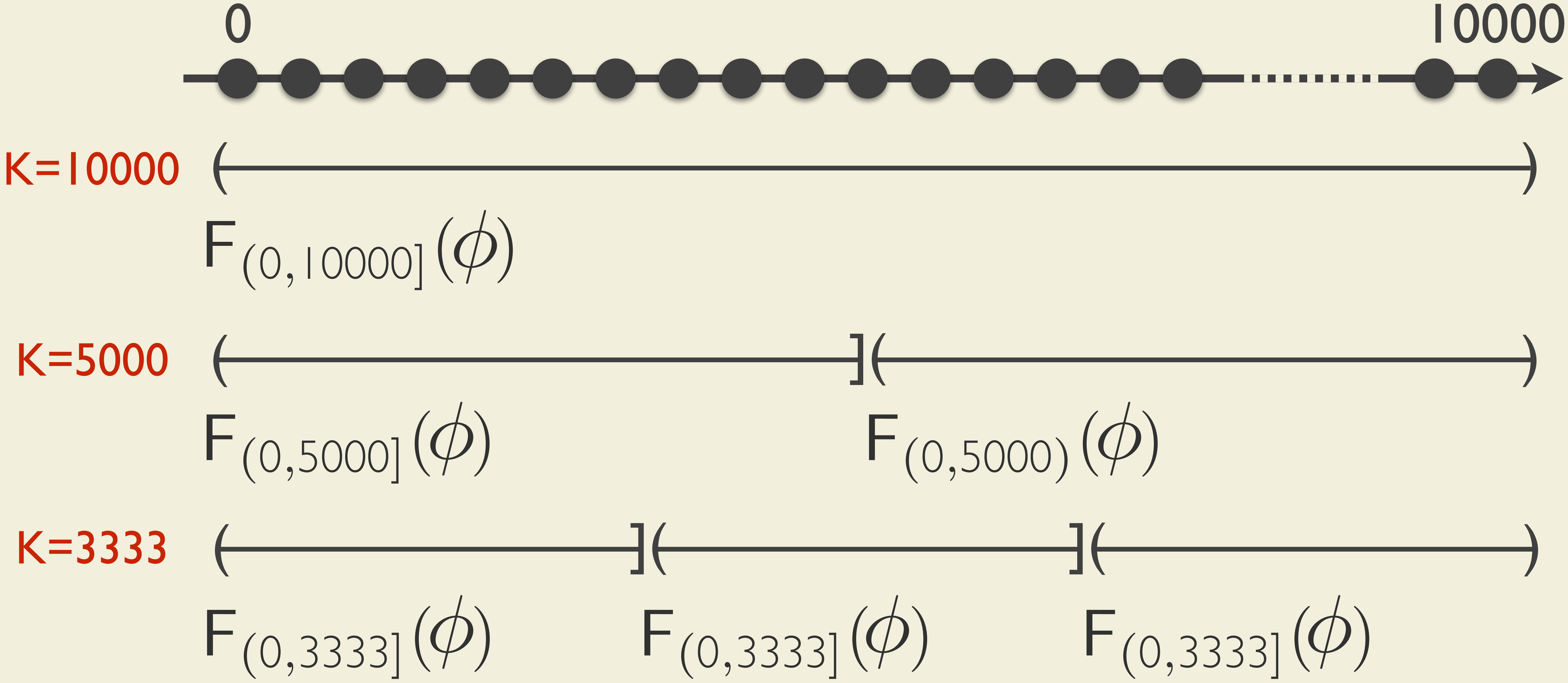
K=10000

$$F_{(0,10000]}(\phi)$$

Parametric Decomposition



Parametric Decomposition



K=10000

$F_{(0,10000]}(\phi)$

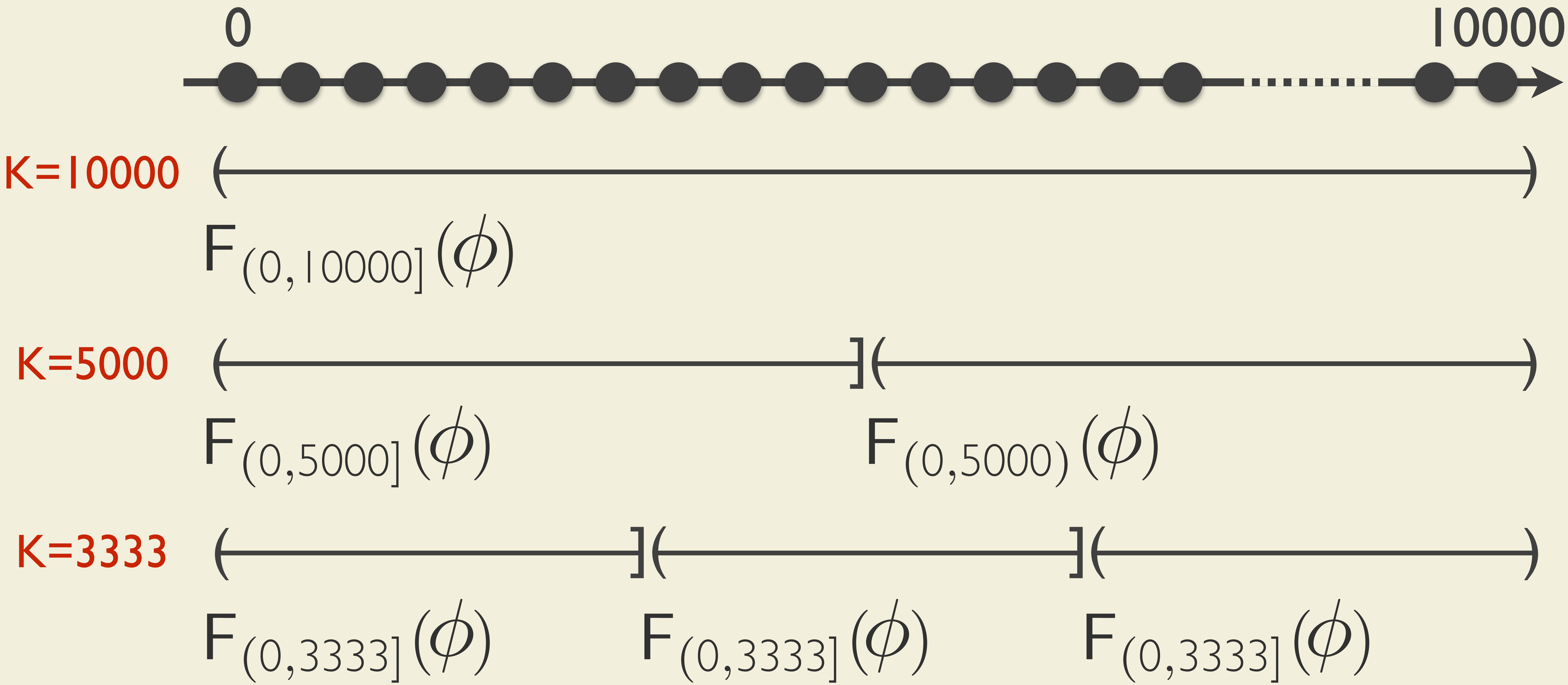
K=5000

$F_{(0,5000]}(\phi)$ $F_{(0,5000)}(\phi)$

K=3333

$F_{(0,3333]}(\phi)$ $F_{(0,3333]}(\phi)$ $F_{(0,3333]}(\phi)$

Parametric Decomposition



0

10000

K=10000

$F_{(0,10000]}(\phi)$

K=5000

$F_{(0,5000]}(\phi)$ $F_{(0,5000)}(\phi)$

K=3333

$F_{(0,3333]}(\phi)$ $F_{(0,3333]}(\phi)$ $F_{(0,3333]}(\phi)$

… or using any parameter K

How to pick an appropriate parameter K?

How to pick an appropriate parameter K?

Rule of thumb: largest K supported by the infrastructure

Trace Checking using MapReduce and Lazy Semantics

1. Infer the value of K

2. Analyze the input formula Φ

3. If all intervals are bounded by K, apply the point-based semantics

4. Otherwise, decompose the formula according to K and then apply the lazy semantics

Evaluation Highlights

- RQ1: Scalability with respect to the size of the time interval

- RQ2: Time/memory tradeoff with respect to the decomposition parameter K

- RQ3: Size and the height of the decomposed formula

Evaluation Highlights

- RQ1: Scalability with respect to the size of the time interval

Memory scalability obtained

- RQ2: Time/memory tradeoff with respect to the decomposition parameter K

- RQ3: Size and the height of the decomposed formula

Evaluation Highlights

- RQ1: Scalability with respect to the size of the time interval

Memory scalability obtained

- RQ2: Time/memory tradeoff with respect to the decomposition parameter K

Smaller K: less memory, more time

- RQ3: Size and the height of the decomposed formula

Evaluation Highlights

- RQ1: Scalability with respect to the size of the time interval

Memory scalability obtained

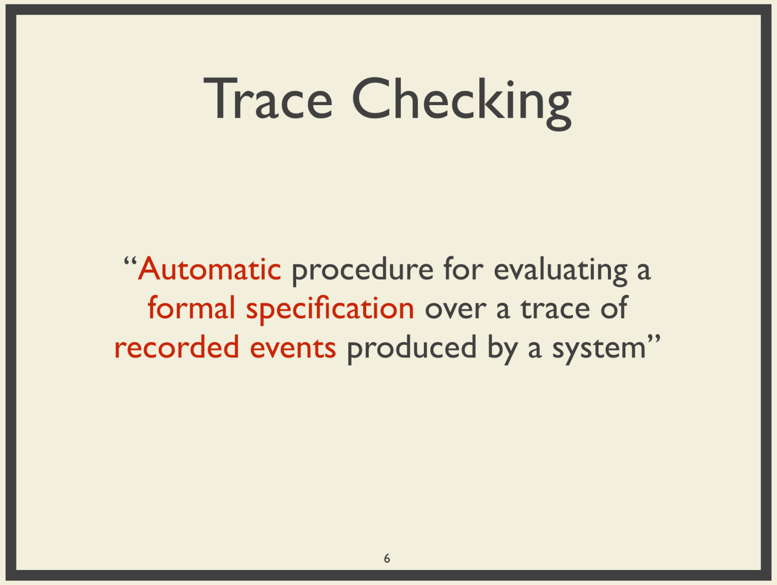
- RQ2: Time/memory tradeoff with respect to the decomposition parameter K

Smaller K: less memory, more time

- RQ3: Size and the height of the decomposed formula

Smaller K: larger formula

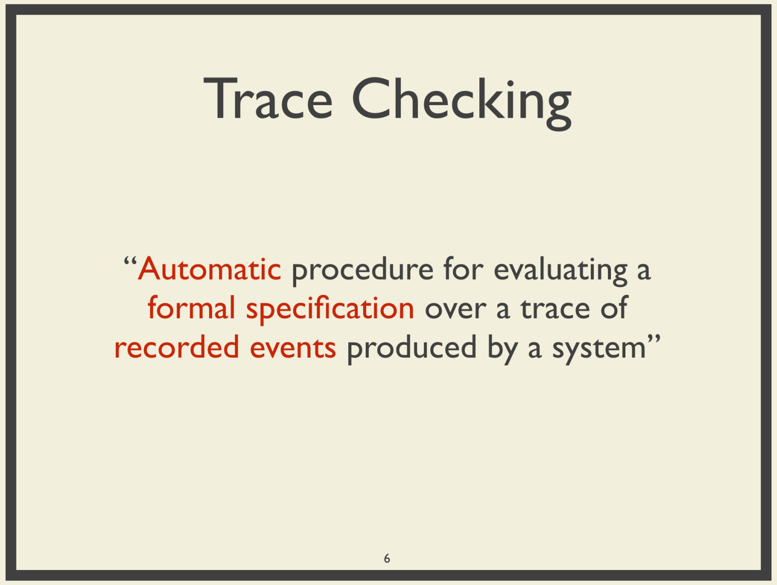
Summary



Trace Checking

"Automatic procedure for evaluating a formal specification over a trace of recorded events produced by a system"

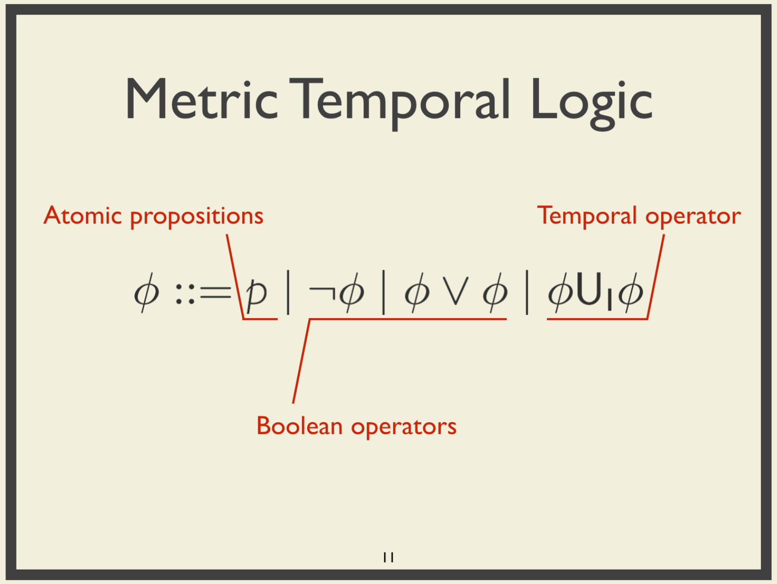
6



Trace Checking

"Automatic procedure for evaluating a formal specification over a trace of recorded events produced by a system"

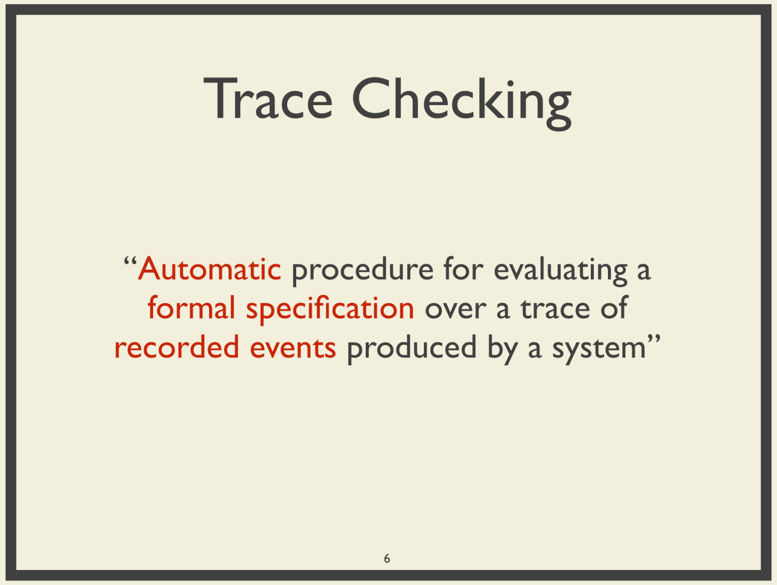
6



Metric Temporal Logic

$$\phi ::= \overbrace{p}^{\text{Atomic propositions}} \mid \underbrace{\neg\phi \mid \phi \vee \phi}_{\text{Boolean operators}} \mid \overbrace{\phi \mathsf{U}_I \phi}^{\text{Temporal operator}}$$

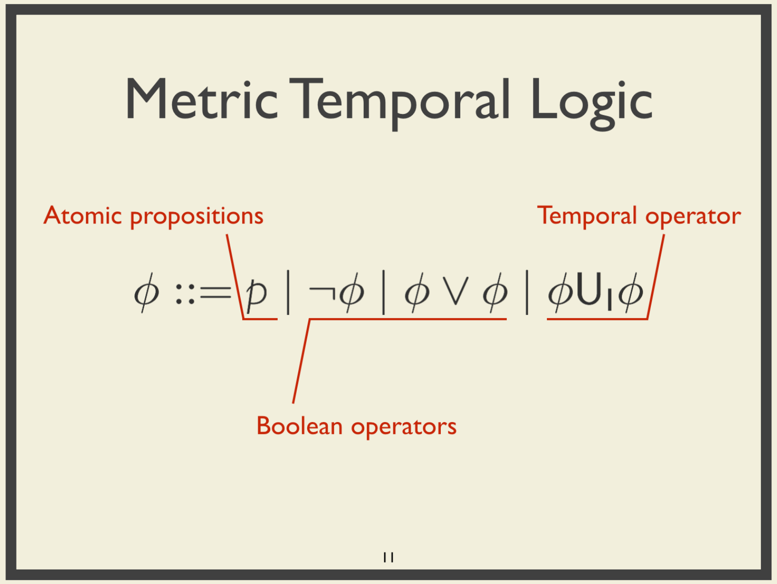
11



Trace Checking

"Automatic procedure for evaluating a formal specification over a trace of recorded events produced by a system"

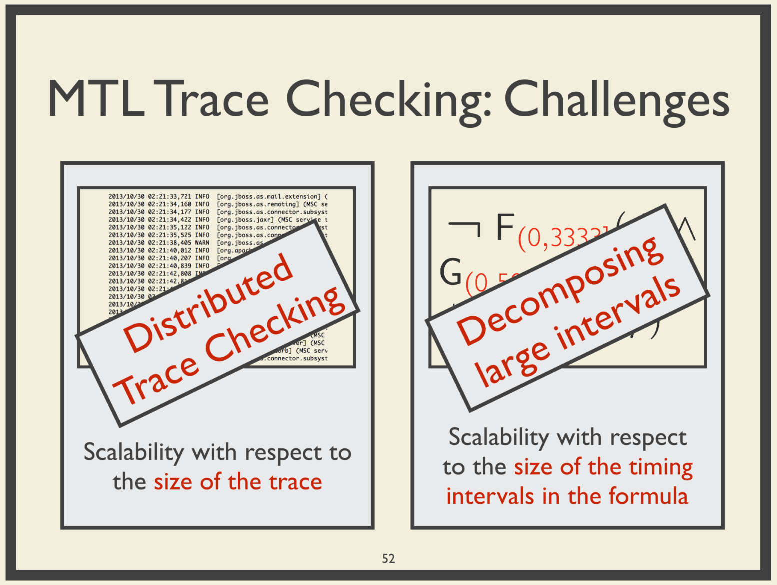
6



Metric Temporal Logic

$$\phi ::= \overbrace{p}^{\text{Atomic propositions}} \mid \underbrace{\neg\phi \mid \phi \vee \phi}_{\text{Boolean operators}} \mid \overbrace{\phi \mathsf{U}_I \phi}^{\text{Temporal operator}}$$

11

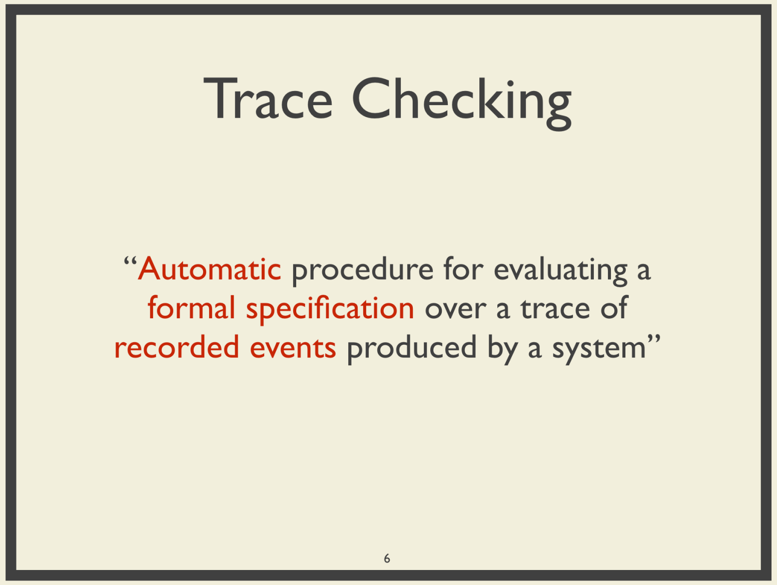


MTL Trace Checking: Challenges

Scalability with respect to the size of the trace

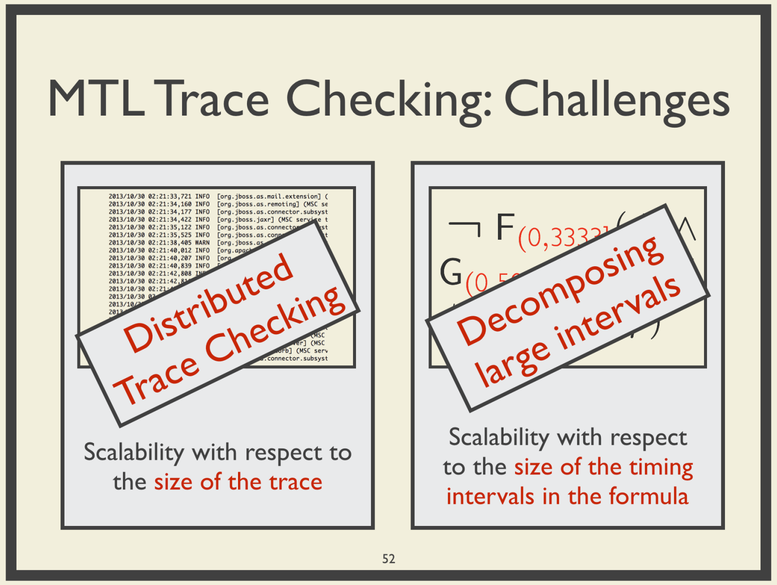
Scalability with respect to the size of the timing intervals in the formula

52



Trace Checking

"Automatic procedure for evaluating a formal specification over a trace of recorded events produced by a system"



MTL Trace Checking: Challenges

```
2013/10/30 02:21:33,721 INFO  [org.jboss.as.mail.extension] (
2013/10/30 02:21:34,160 INFO  [org.jboss.as.remoting] (MSC se
2013/10/30 02:21:34,177 INFO  [org.jboss.as.connector.subsyst
2013/10/30 02:21:34,422 INFO  [org.jboss.jaxr] (MSC service t
2013/10/30 02:21:35,122 INFO  [org.jboss.as.connecto
2013/10/30 02:21:35,525 INFO  [org.jboss.as.conn
2013/10/30 02:21:38,405 WARN  [org.jboss.as
2013/10/30 02:21:40,012 INFO  [org.apac
2013/10/30 02:21:40,207 INFO  [org
2013/10/30 02:21:40,839 INFO
2013/10/30 02:21:42,808 IN
2013/10/30 02:21:42,81
2013/10/30 02:21:
2013/10/30 02
2013/10/
2013
```

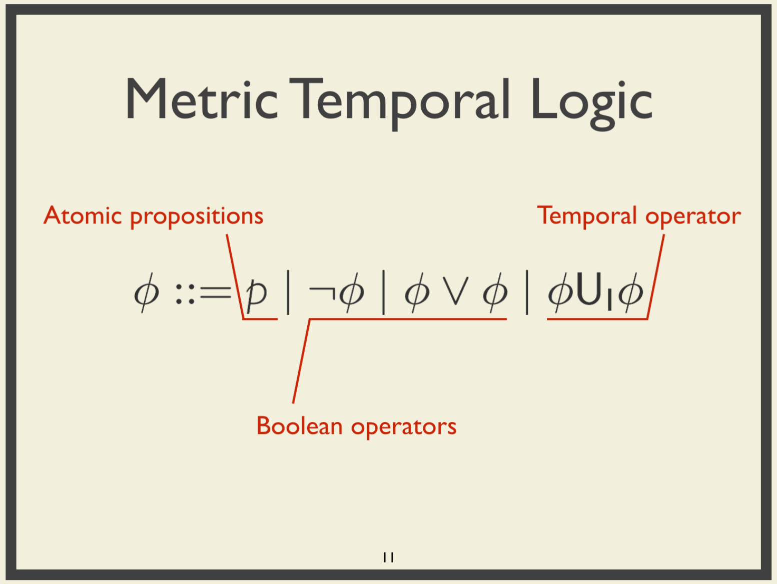
Distributed Trace Checking

Scalability with respect to the size of the trace

$\neg\ \mathsf{F}_{(0,3333]}$ … $\mathsf{G}_{(0,5}$ …

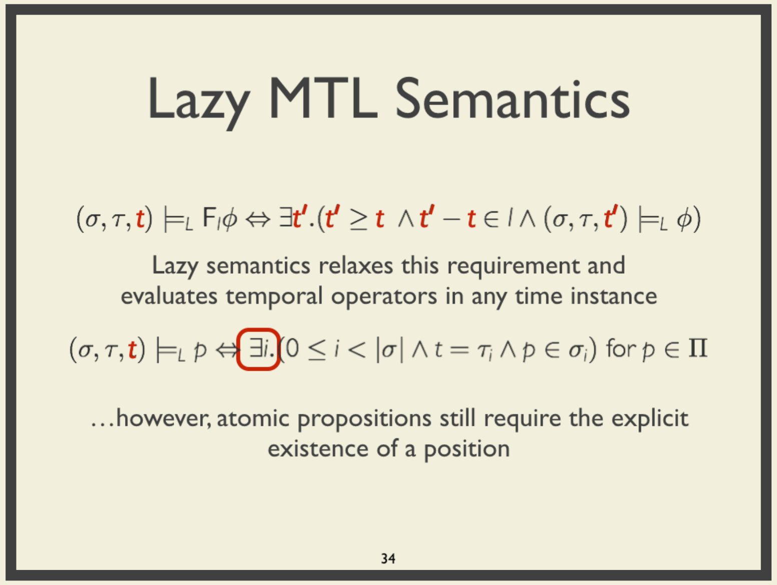
Decomposing large intervals

Scalability with respect to the size of the timing intervals in the formula



Metric Temporal Logic

$$\phi ::= \underbrace{p}_{\text{Atomic propositions}} \mid \underbrace{\neg\phi \mid \phi \vee \phi}_{\text{Boolean operators}} \mid \underbrace{\phi \mathsf{U}_I \phi}_{\text{Temporal operator}}$$



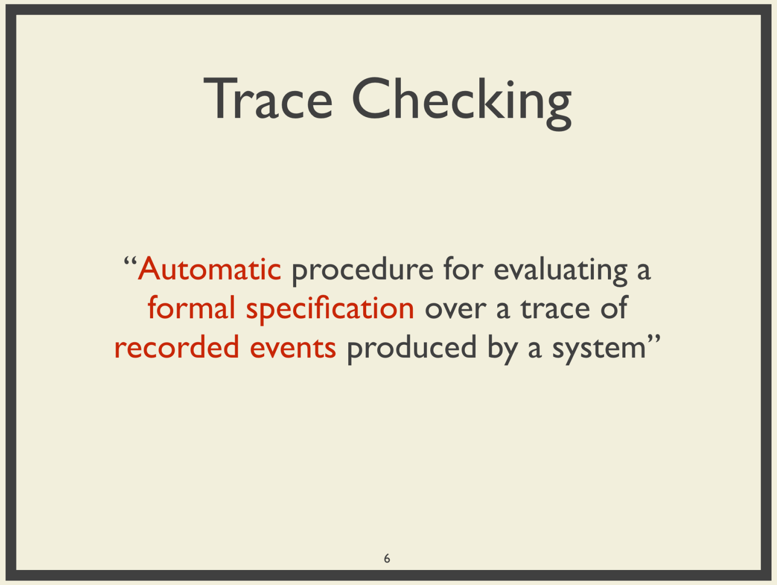
Lazy MTL Semantics

$$(\sigma,\tau,t) \models_L \mathsf{F}_I\phi \Leftrightarrow \exists t'.(t' \geq t \ \wedge t' - t \in I \wedge (\sigma,\tau,t') \models_L \phi)$$

Lazy semantics relaxes this requirement and evaluates temporal operators in any time instance

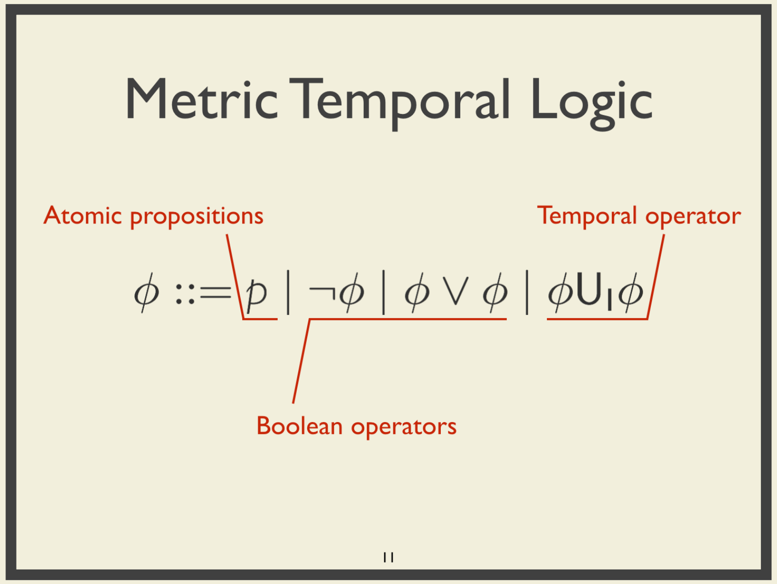
$$(\sigma,\tau,t) \models_L p \Leftrightarrow \boxed{\exists i.}(0 \leq i < |\sigma| \wedge t = \tau_i \wedge p \in \sigma_i) \text{ for } p \in \Pi$$

…however, atomic propositions still require the explicit existence of a position



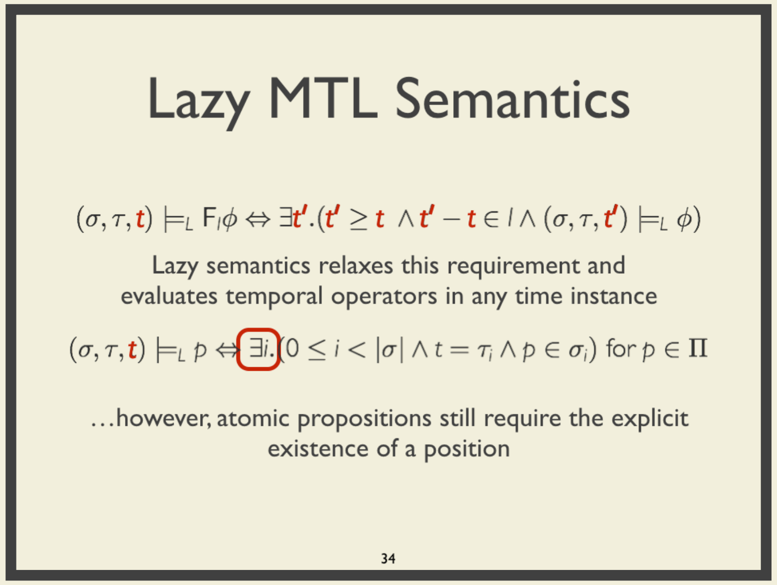
Trace Checking

"Automatic procedure for evaluating a formal specification over a trace of recorded events produced by a system"



Metric Temporal Logic

$$\phi ::= \underbrace{p}_{\text{Atomic propositions}} \mid \underbrace{\neg\phi \mid \phi \vee \phi}_{\text{Boolean operators}} \mid \underbrace{\phi \mathsf{U}_I \phi}_{\text{Temporal operator}}$$



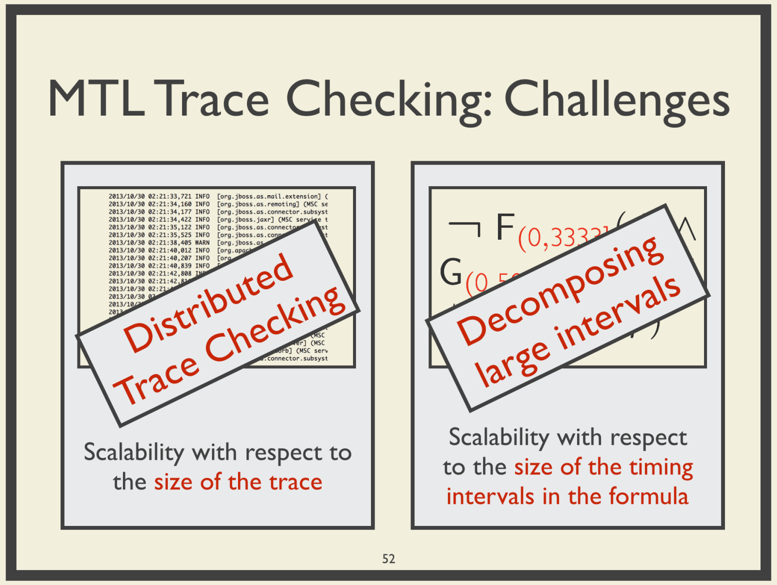
Lazy MTL Semantics

$$(\sigma, \tau, t) \models_L \mathsf{F}_I\phi \Leftrightarrow \exists t'.(t' \geq t \wedge t' - t \in I \wedge (\sigma, \tau, t') \models_L \phi)$$

Lazy semantics relaxes this requirement and evaluates temporal operators in any time instance

$$(\sigma, \tau, t) \models_L p \Leftrightarrow \boxed{\exists i.}(0 \leq i < |\sigma| \wedge t = \tau_i \wedge p \in \sigma_i) \text{ for } p \in \Pi$$

…however, atomic propositions still require the explicit existence of a position



MTL Trace Checking: Challenges

```
2013/10/30 02:21:33,721 INFO  [org.jboss.as.mail.extension] (
2013/10/30 02:21:34,160 INFO  [org.jboss.as.remoting] (MSC se
2013/10/30 02:21:34,177 INFO  [org.jboss.as.connector.subsyst
2013/10/30 02:21:34,422 INFO  [org.jboss.jaxr] (MSC service t
2013/10/30 02:21:35,122 INFO  [org.jboss.as.connector
2013/10/30 02:21:35,525 INFO  [org.jboss.as.conn
2013/10/30 02:21:38,405 WARN  [org.jboss.as
2013/10/30 02:21:40,012 INFO  [org.apach
2013/10/30 02:21:40,207 INFO  [org
2013/10/30 02:21:40,839 INFO
2013/10/30 02:21:42,808 IN
2013/10/30 02:21:42,8
2013/10/30 02:21:
2013/10/30 0
2013/10/
2013
```

Distributed Trace Checking

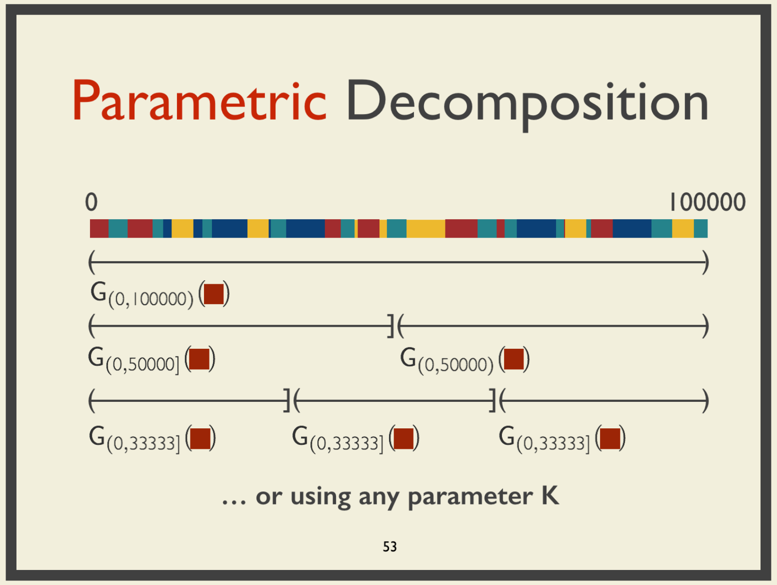
Scalability with respect to the size of the trace

$\neg\ \mathsf{F}_{(0,3333}$ … $\wedge$

$\mathsf{G}_{(0,5}$

Decomposing large intervals

Scalability with respect to the size of the timing intervals in the formula



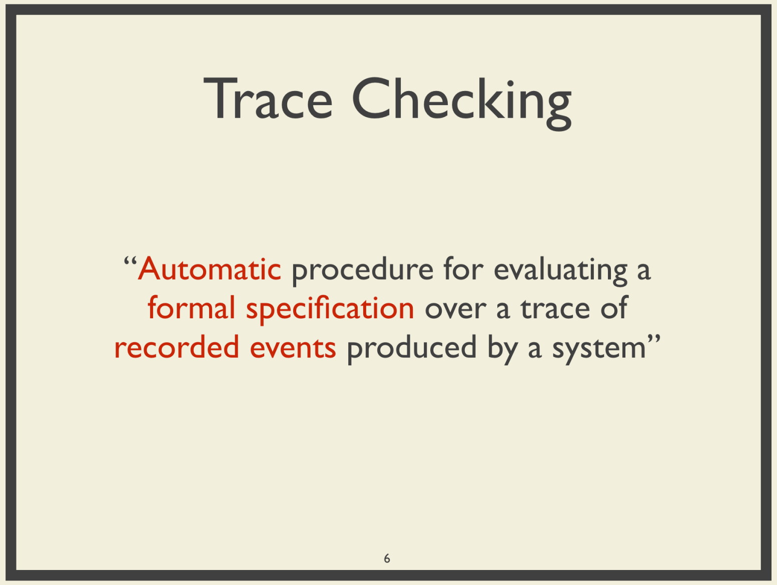
Parametric Decomposition

$\mathsf{G}_{(0,100000)}(\blacksquare)$

$\mathsf{G}_{(0,50000]}(\blacksquare)$ $\mathsf{G}_{(0,50000)}(\blacksquare)$

$\mathsf{G}_{(0,33333]}(\blacksquare)$ $\mathsf{G}_{(0,33333]}(\blacksquare)$ $\mathsf{G}_{(0,33333]}(\blacksquare)$

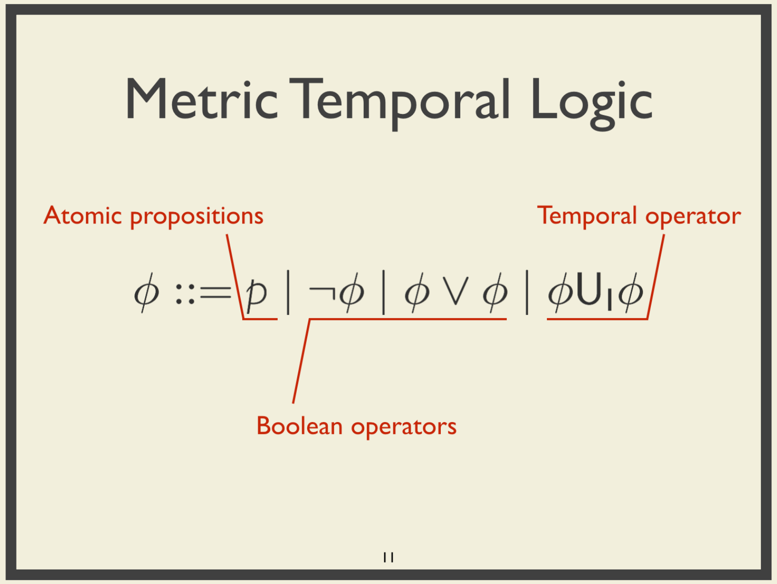
… or using any parameter K



Trace Checking

"Automatic procedure for evaluating a
formal specification over a trace of
recorded events produced by a system"

6



Metric Temporal Logic

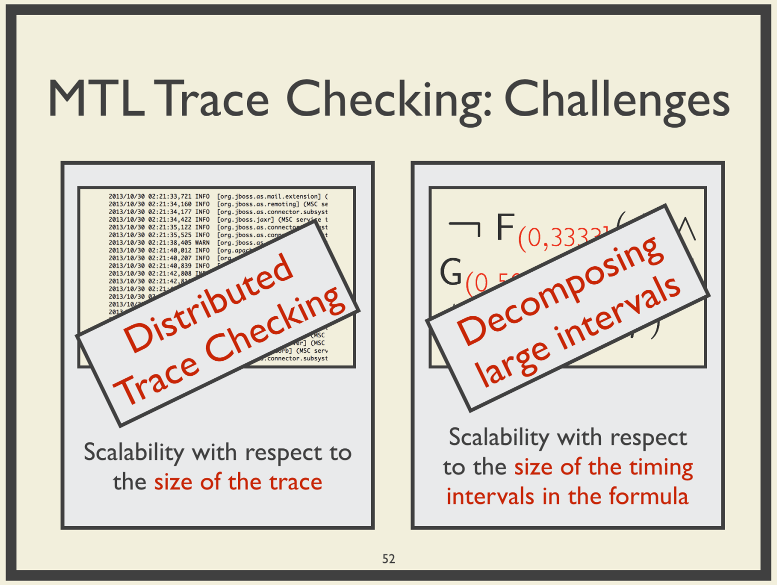
$\phi ::= p \mid \neg\phi \mid \phi \vee \phi \mid \phi \mathsf{U}_I \phi$

Atomic propositions

Temporal operator

Boolean operators

11



MTL Trace Checking: Challenges

```
2013/10/30 02:21:33,721 INFO  [org.jboss.as.mail.extension] (
2013/10/30 02:21:34,160 INFO  [org.jboss.as.remoting] (MSC se
2013/10/30 02:21:34,177 INFO  [org.jboss.as.connector.subsyst
2013/10/30 02:21:34,422 INFO  [org.jboss.jaxr] (MSC service t
2013/10/30 02:21:35,122 INFO  [org.jboss.as.connector      yst
2013/10/30 02:21:35,525 INFO  [org.jboss.as.conn           t
2013/10/30 02:21:38,405 WARN  [org.jboss.as
2013/10/30 02:21:40,012 INFO  [org.apac
2013/10/30 02:21:40,207 INFO  [org
2013/10/30 02:21:40,839 INFO
2013/10/30 02:21:42,808 IN
2013/10/30 02:21:42,81
2013/10/30 02:21:
2013/10/30 0
2013/10/
2013
                                          (MSC
                                      ver] (MSC
                                      orb] (MSC serv
                                      s.connector.subsyst
```

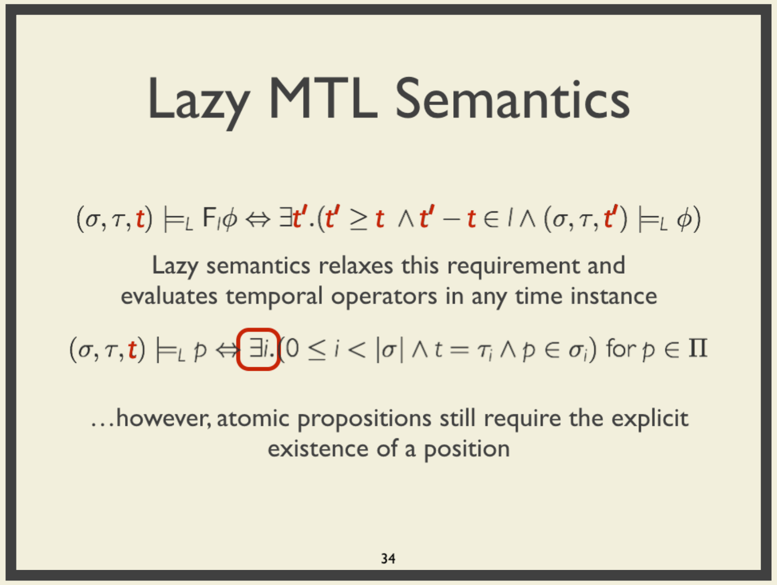
Distributed Trace Checking

Scalability with respect to the size of the trace

Decomposing large intervals

Scalability with respect to the size of the timing intervals in the formula

52



Lazy MTL Semantics

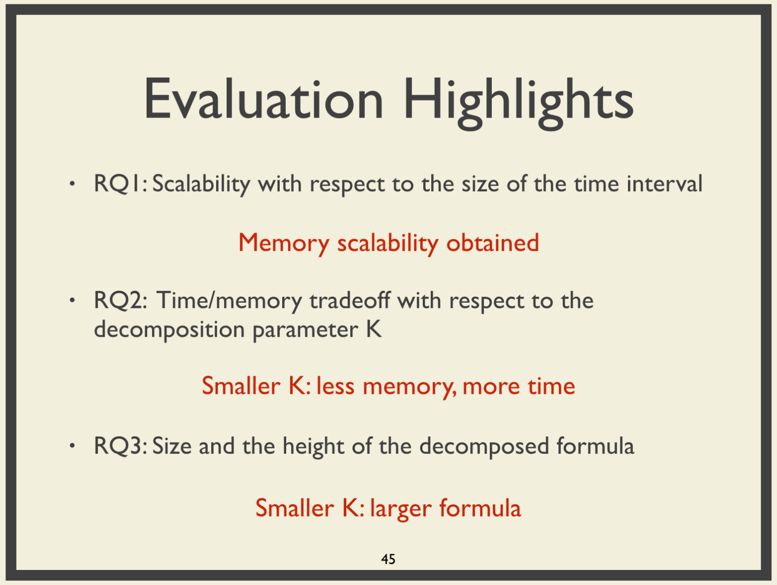
$(\sigma, \tau, t) \models_L \mathsf{F}_I\phi \Leftrightarrow \exists t'.(t' \geq t \wedge t' - t \in I \wedge (\sigma, \tau, t') \models_L \phi)$

Lazy semantics relaxes this requirement and
evaluates temporal operators in any time instance

$(\sigma, \tau, t) \models_L p \Leftrightarrow \boxed{\exists i.}(0 \leq i < |\sigma| \wedge t = \tau_i \wedge p \in \sigma_i)$ for $p \in \Pi$

…however, atomic propositions still require the explicit
existence of a position

34



Evaluation Highlights

- RQ1: Scalability with respect to the size of the time interval

Memory scalability obtained

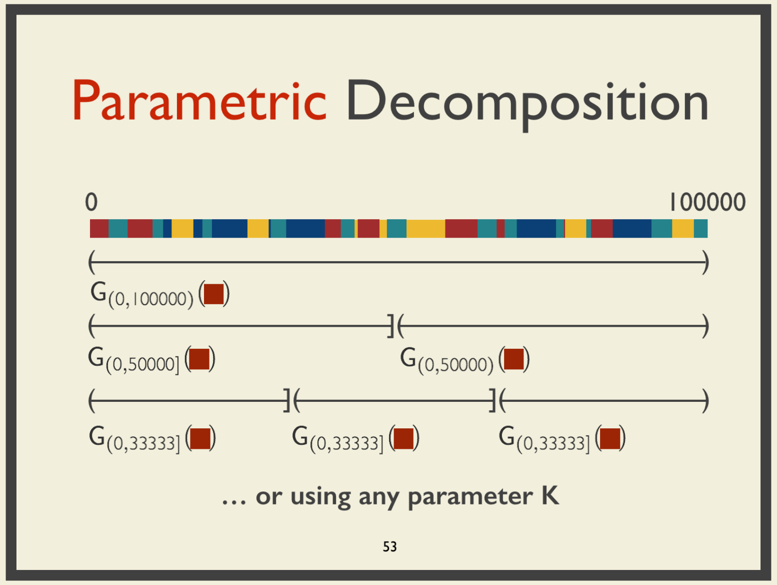
- RQ2: Time/memory tradeoff with respect to the decomposition parameter K

Smaller K: less memory, more time

- RQ3: Size and the height of the decomposed formula

Smaller K: larger formula

45



Parametric Decomposition

… or using any parameter K

53

Efficient Large-scale Trace Checking using MapReduce

Srđan Krstić

with

Marcello M. Bersani, Domenico Bianculli, Carlo Ghezzi and Pierluigi San Pietro

POLITECNICO MILANO

uni.lu UNIVERSITÉ DU LUXEMBOURG

Future Research Directions

- Lazy semantics vs signal-based semantics
- Automatic and heterogeneous decomposition
- Decomposition of other operators