A constraint optimization model for analysis of telecommunication protocol logs

Olga Grinchtein — Ericsson AB

Mats Carlsson — SICS

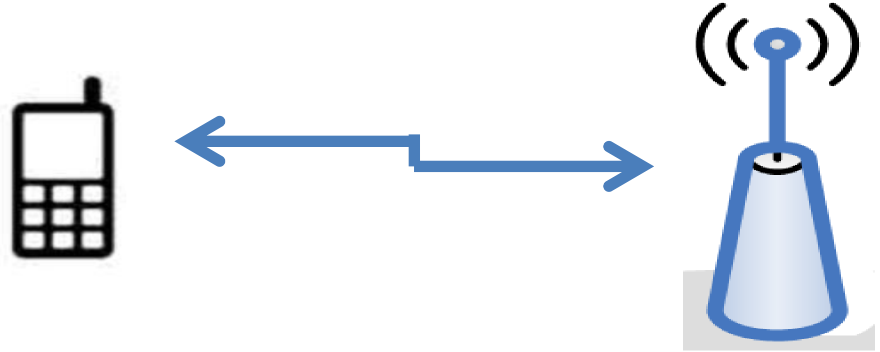
Justin Pearson — Uppsala University

Outline

- Overview of the approach
- Long Term Evolution (LTE) Radio Access Network
- A case study Public Warning System
- Constraint model
- Optimization
- Experimental results
- Conclusion

Overview

- The LTE Radio Base Station (eNodeB) is SUT at our department at Ericsson.
- The eNodeB communicates with the User Equipment (UE).



- The eNodeB has different functions. Our case study is Public Warning System.

Overview

- Test harness should analyze User Equipment protocol logs. Protocol logs are sequences of messages with timestamps.
- Protocol log analysis includes checks that protocol log contains correct messages with correct timing and content.
- We use MiniZinc constraint solving system to directly model the protocol and to implement a test harness to analyze protocol log.
- The approach requires a script that reads the protocol log, creates arrays of MiniZinc variables, and assigns values to the variables according to the information provided in the log.

Minizinc

- Constraint specification language
- Solver independent
- Separation data from model
- Model consists of variable declarations and constraint definitions
- Scalar types (Booleans, integers, floats) and compound types (sets, arrays)
- Comparisons, arithmetic operations(+,·,sum), logical operations ($\vee$,$\wedge$, forall, exists), set operations, array operations

Protocol log

Parameter of message M2

Parameters of message M3

10 : 00 : 00.000 M1{}
10 : 00 : 00.080 M2{y = 80}
10 : 00 : 00.400 M3{z1 = 1; z2 = aaba}
10 : 00 : 00.720 M3{z1 = 2; z2 = abab}
10 : 00 : 01.040 M3{z1 = 4; z2 = aaaa}
10 : 00 : 01.450 M1{}
10 : 00 : 01.580 M2{y = 320}
10 : 00 : 01:600 M3{z1 = 1; z2 = aaba}
10 : 00 : 01:920 M3{z1 = 2; z2 = abab}
10 : 00 : 02:900 M1{}
10 : 00 : 03:120 M3{z1 = 1; z2 = aaba}
10 : 00 : 03:440 M3{z1 = 2; z2 = abab}
10 : 00 : 04:350 M1{}

M1Time = [0; 1450; 2900; 4350]
M2Time = [80; 1580]
M3Time = [400; 720; 1040; 1600; 1920; 3120; 3440]

M2y = [1; 2]
M3z1 = [1; 2; 4; 1; 2; 1; 2]
M3z2 = [1; 2; 4; 1; 2; 1; 2]

Example of constraint

10 : 00 : 00.000	M1{}
10 : 00 : 00.080	M2{y = 80}
10 : 00 : 00.400	M3{z1 = 1; z2 = aaba}
10 : 00 : 00.720	M3{z1 = 2; z2 = abab}
10 : 00 : 01.040	M3{z1 = 4; z2 = aaaa}
10 : 00 : 01.450	M1{}
10 : 00 : 01.580	M2{y = 320}
10 : 00 : 01:600	M3{z1 = 1; z2 = aaba}
10 : 00 : 01:920	M3{z1 = 2; z2 = abab}
10 : 00 : 02:900	M1{}
10 : 00 : 03:120	M3{z1 = 1; z2 = aaba}
10 : 00 : 03:440	M3{z1 = 2; z2 = abab}
10 : 00 : 04:350	M1{}

The first message M1 the mobile phone reads with some delay, and we introduce decision variable **delay**, which is between 0 and 100 milliseconds.

After a M1 message, which is transmitted within 1500 milliseconds, the M3 messages follow with z1 equal to 1,2, 3 and 4. After messages M1, which are transmitted after 1500 milliseconds, the messages M3 follow with z1 equal to 1 and 2.

Boolean decision variable **M3contentinc** equal to 1 indicates an error in the log

$$
\begin{aligned}
&(\forall 1 \le i \le 3) \\
&\quad ((\mathtt{M1Time}_i < 1500 - delay \wedge \\
&\quad (\forall 1 \le j \le 4)(\exists 1 \le k \le 7)\mathtt{M1Time}_i < \mathtt{M3Time}_k < \mathtt{M1Time}_{i+1} \wedge \mathtt{M3z1}_k = j) \\
&\quad \vee \\
&\quad (\mathtt{M1Time}_i \ge 1500 - delay \wedge \\
&\quad (\forall 1 \le j \le 2)(\exists 1 \le k \le 7)\mathtt{M1Time}_i < \mathtt{M3Time}_k < \mathtt{M1Time}_{i+1} \wedge \mathtt{M3z1}_k = j)) \\
&\leftrightarrow M3contentinc_i = 0
\end{aligned}
$$

Example of constraint

10 : 00 : 00.000	M1{}
10 : 00 : 00.080	M2{y = 80}
10 : 00 : 00.400	M3{z1 = 1; z2 = aaba}
10 : 00 : 00.720	M3{z1 = 2; z2 = abab}
10 : 00 : 01.040	M3{z1 = 4; z2 = aaaa}
10 : 00 : 01.450	M1{}
10 : 00 : 01.580	M2{y = 320}
10 : 00 : 01:600	M3{z1 = 1; z2 = aaba}
10 : 00 : 01:920	M3{z1 = 2; z2 = abab}
10 : 00 : 02:900	M1{}
10 : 00 : 03:120	M3{z1 = 1; z2 = aaba}
10 : 00 : 03:440	M3{z1 = 2; z2 = abab}
10 : 00 : 04:350	M1{}

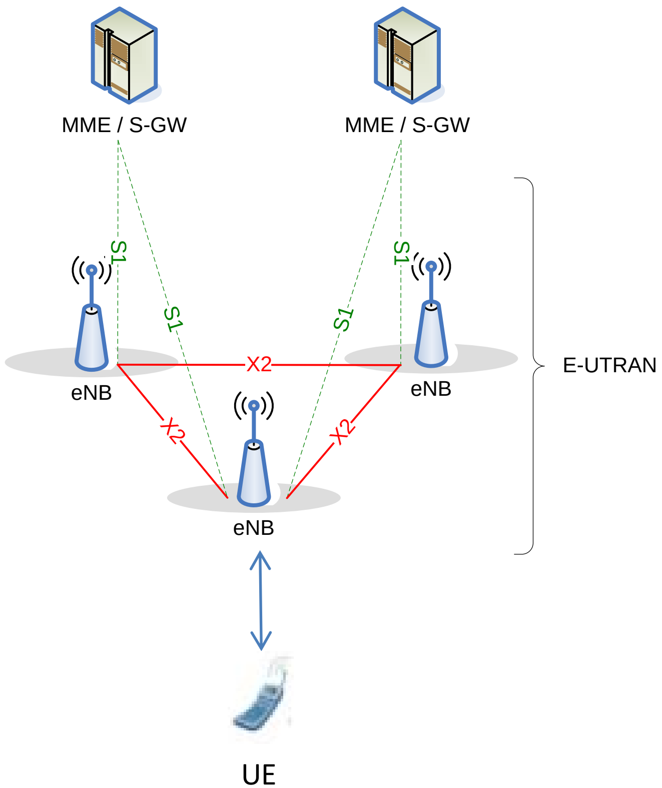
$M3contentinc_1$ =1, since there is no message M3 with the parameter z1 = 3 between first and second messages M1.

If **delay** = 20, then $M1Time_2$ = 1450 < 1500-**delay** and **$M3contentinc_2$** = 1, since there are no messages M3 with z1 equal to 3 and 4 between second and third messages M1.

$$
\begin{aligned}
&(\forall 1 \le i \le 3)\\
&\quad ((\mathtt{M1Time}_i < 1500 - delay \wedge\\
&\quad (\forall 1 \le j \le 4)(\exists 1 \le k \le 7)\mathtt{M1Time}_i < \mathtt{M3Time}_k < \mathtt{M1Time}_{i+1} \wedge \mathtt{M3z1}_k = j)\\
&\quad \vee\\
&\quad (\mathtt{M1Time}_i \ge 1500 - delay \wedge\\
&\quad (\forall 1 \le j \le 2)(\exists 1 \le k \le 7)\mathtt{M1Time}_i < \mathtt{M3Time}_k < \mathtt{M1Time}_{i+1} \wedge \mathtt{M3z1}_k = j))\\
&\leftrightarrow M3contentinc_i = 0
\end{aligned}
$$

LTE Radio Base Station (eNodeB)

A LTE Radio Access Network consists of LTE Radio Base Stations (RBS), which support the LTE air interface and performs radio resource management.
An eNB connects to UE via the air interface.
LTE RBSes are interconnected with each other by means of the X2 interfaces.
LTE RBSes are also connected by means of the S1 interface to the Evolved Packet Core



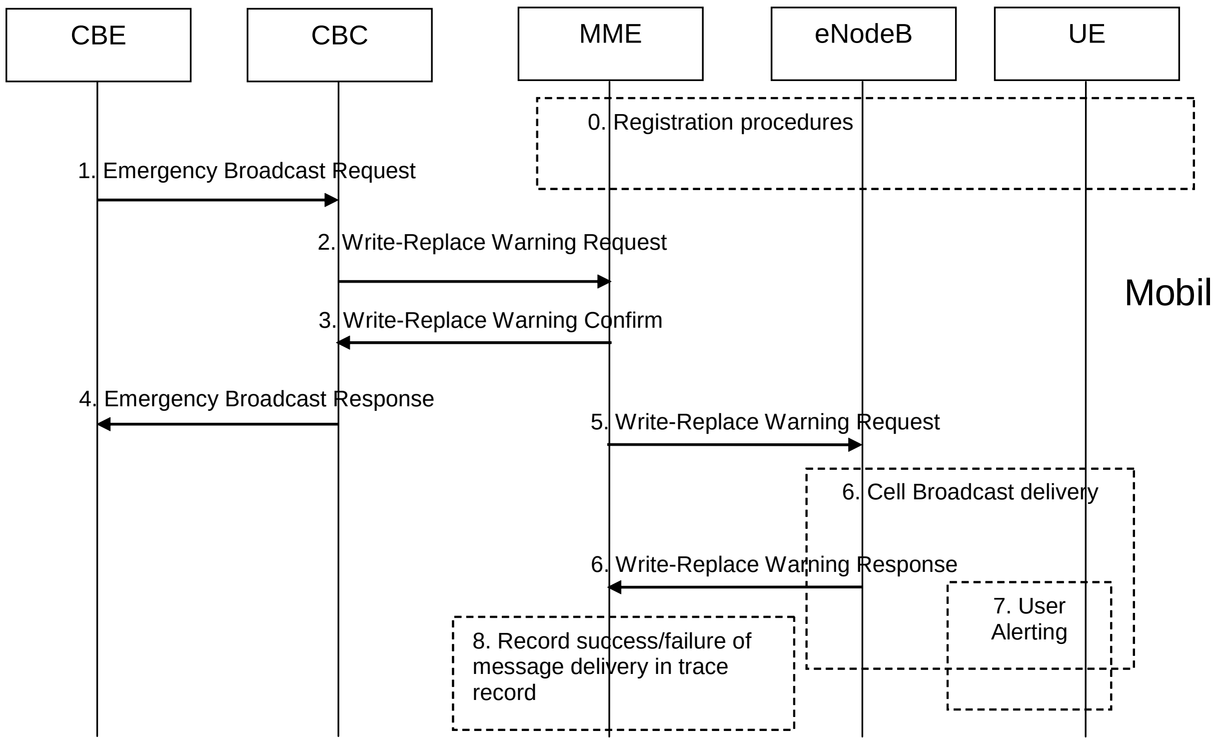
Mobility Management Entity (MME)

Serving Gateway (S-GW)

User Equipment (UE)

3GPP TS 36.300

Public Warning System



3GPP TS 23.041

Cell Broadcast Entity (CBE)

Cell Broadcast Center (CBC)

Mobility Management Entity (MME)

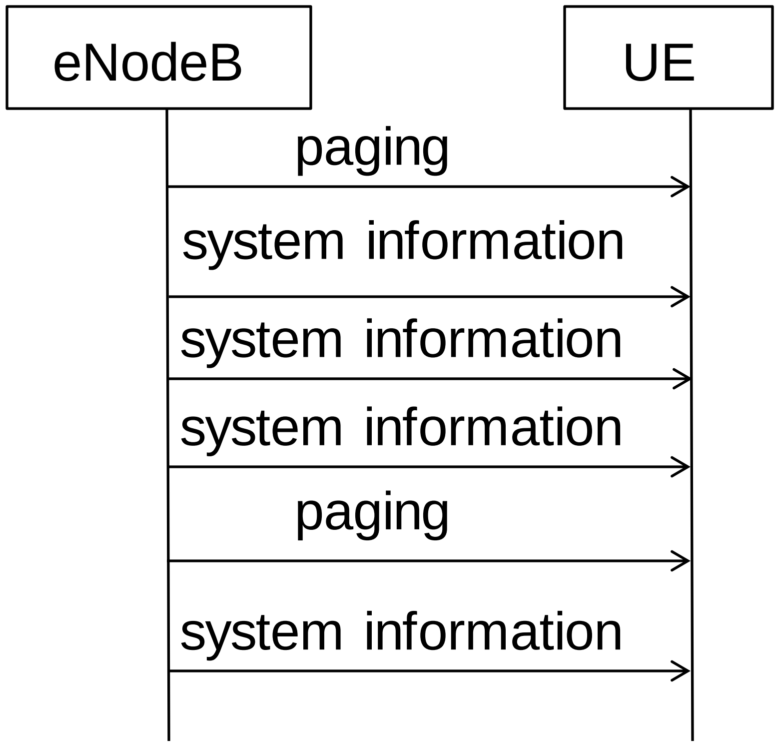
User Equipment (UE)

The eNodeB sends different types of system information messages to UE

- SIB1 (schedulingInfoList)
- SIB10 (primary notification)
- SIB11 (secondary notification)

Secondary notification can come in one or several segments

Public Warning System

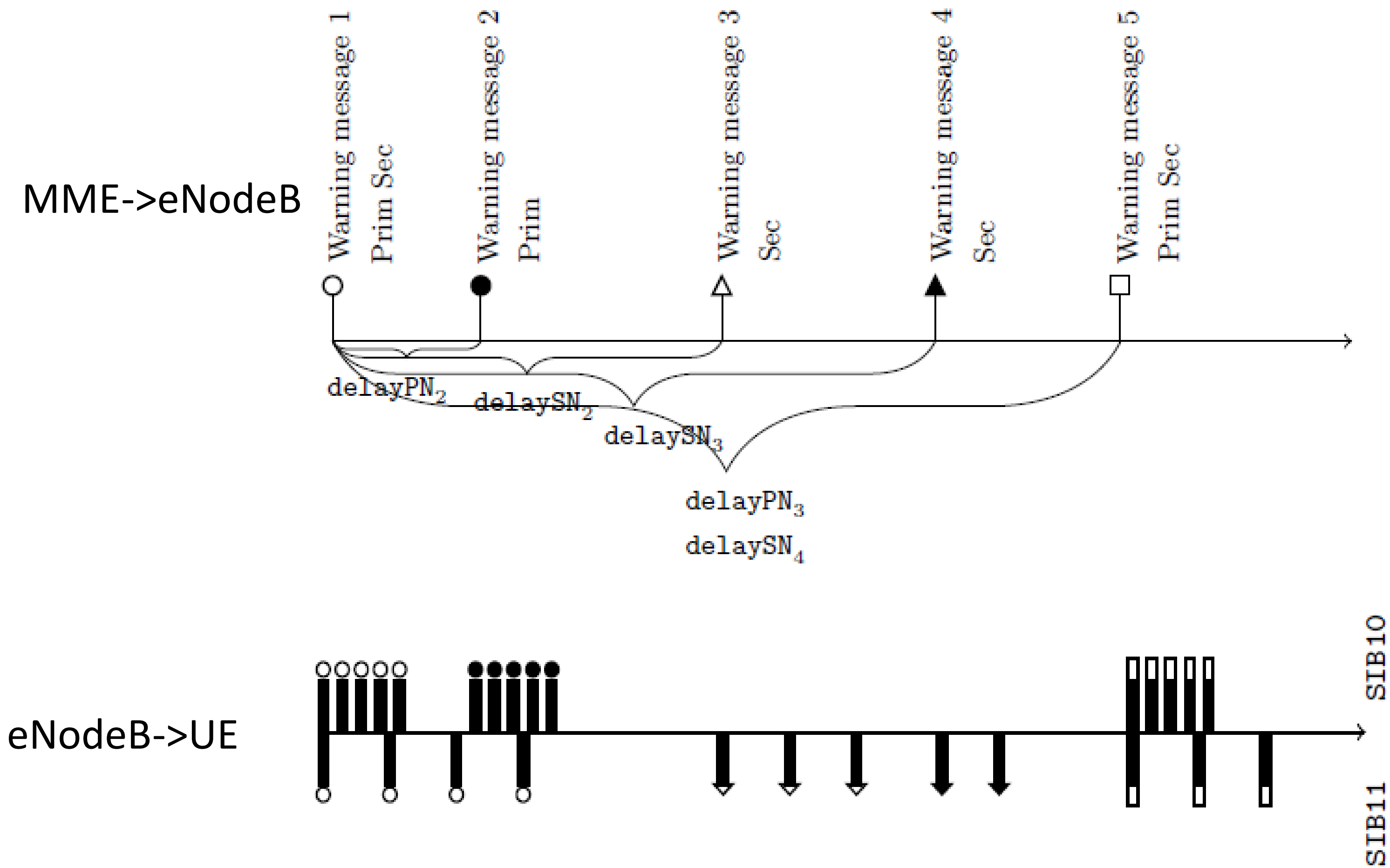


The eNodeB sends different types of system information messages

- SIB1 (schedulingInfoList)
- SIB10 (primary notification)
- SIB11 (secondary notification)

Secondary notification can come in one or several segments

Replacement of Warning messages



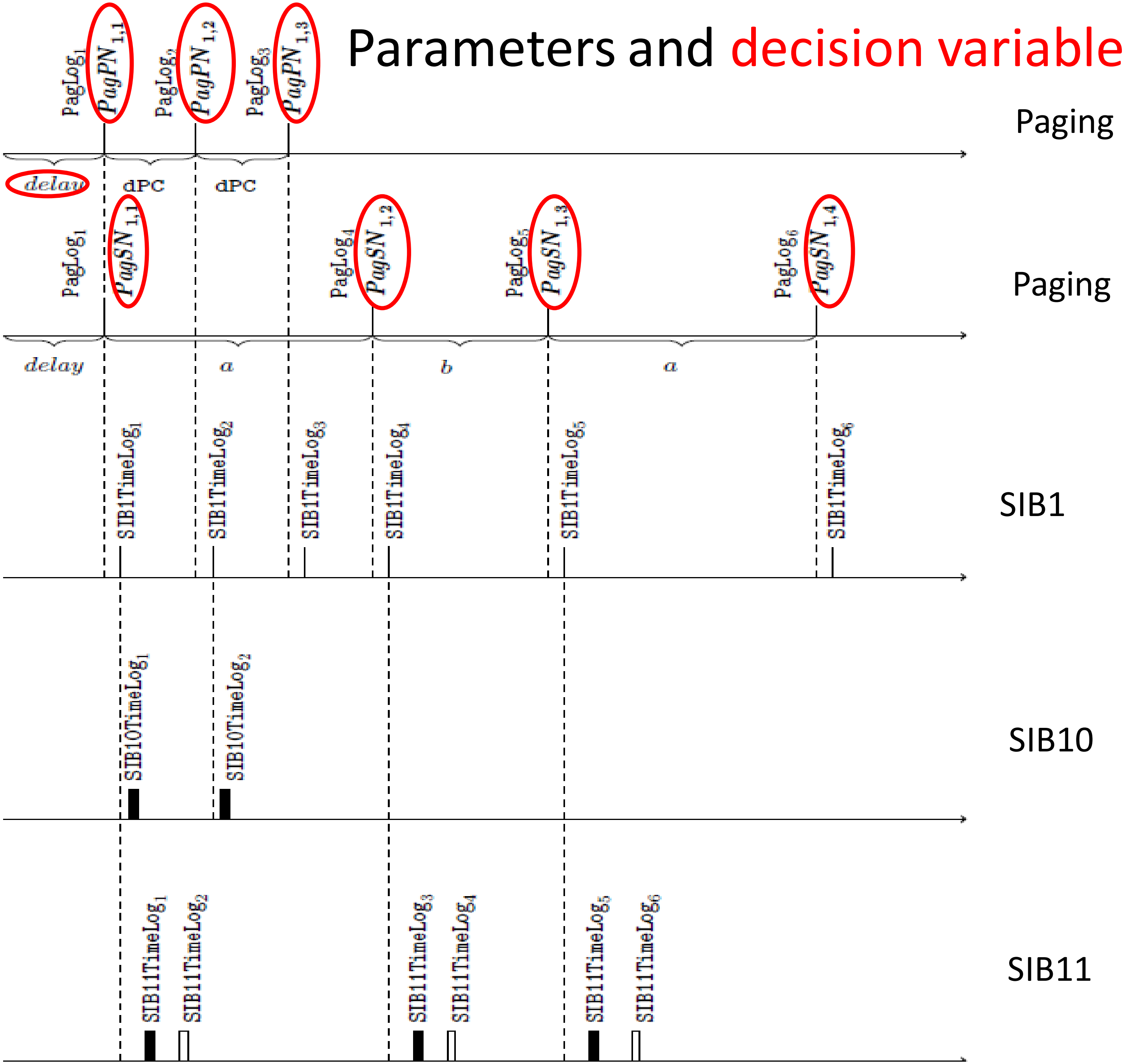
Constraint model

- Model consists of constraints on timestamps and the content of paging, SIB1, SIB10 and SIB11 messages.
- We divided the Model into three submodels
  - PagingModel checks that the log contains all required paging messages, and that the number and timestamps of paging messages are correct.
  - SIB1Model checks that the log contains all required SIB1 messages, and that the scheduling information is correct.
  - PrimSecModel checks that the log contains SIB10 messages with correct timestamps, content and identity numbers. It also checks that SIB11 messages have correct timestamps, content, segments and identity numbers.

Constraint model

- The model consists of constraints on arrays of timestamps and message content
  - Decision variable that represents delay of first paging message.
  - Array of decision variables of extra possible delay of warning requests
  - Arrays of decision variables of correct timestamps of paging messages.
  - Arrays of parameters which contain timestamps and content of messages from the log.
  - Boolean decision variables indicate errors in the log
- Protocol log analysis is an optimization problem. We minimize sum of Boolean decision variables.

Parameters and decision variables



Delays of Warning Messages

- Variable delay can occur in acquiring of Warning Messages by eNodeB
- We introduce arrays of decision variables delayPN50 and delaySN50 which represent extra delay of Warning Messages. We assume that delays are less than 50 milliseconds.
- We set values to extra delay $delayPN50_i$ by calculating distances between timestamps of SIB1, SIB10 and SIB11 messages and timestamp of ith Warning Message.

Constraint in the model

$$
\begin{aligned}
&(\forall 1 \le k \le \mathtt{nSIB1Log}) \\
&\quad ((((\exists 1 \le i \le \mathtt{nPrim} - 1) \\
&\qquad (\mathtt{SIB1TimeLog}_k \ge \mathtt{delayPN}_i + \mathit{delayPN50}_i - \mathit{delay} \wedge \\
&\qquad ((\mathit{PagPN}_{i,\mathtt{ndPC}} = -1 \wedge \mathtt{SIB1TimeLog}_k < \\
&\qquad \mathtt{delayPN}_{i+1} + \mathit{delayPN50}_{i+1} - \mathit{delay}) \vee \\
&\qquad (\mathit{PagPN}_{i,\mathtt{ndPC}} \ne -1 \wedge \mathtt{SIB1TimeLog}_k \le \mathit{PagPN}_{i,\mathtt{ndPC}}))) \\
&\quad \bigvee \\
&\quad (\mathtt{SIB1TimeLog}_k \ge \mathtt{delayPN}_{\mathtt{nPrim}} + \mathit{delayPN50}_{\mathtt{nPrim}} - \mathit{delay} \wedge \\
&\qquad \mathtt{SIB1TimeLog}_k \le \mathit{PagPN}_{\mathtt{nPrim},\mathtt{ndPC}})) \\
&\leftrightarrow (\mathtt{SIB1TypeLog}_k = 1 \vee \mathtt{SIB1TypeLog}_k = 3)) \\
&\leftrightarrow \mathit{SIB1PrimTypeinc}_k = 0
\end{aligned}
$$

We check that content of SIB1 message is correct.

nSIB1log, number of SIB1 messages
nPrim, number of primary notifications
SIB1TimeLog, timestamp of SIB1 message in the log
delayPN, delay of warning request
SIB1TypeLog, content of SIB1 message in the log
PagPN, decision variable, correct timestamp of paging message of primary notification
delayPN50, decision variable, extra possible delay of warning request
SIB1PrimeTypeinc, boolean decision variable

Partitioning of timestamps

- $(\forall\ 1 \le i \le nWR)(\exists\ 1 \le j \le nSIB10).\phi$

  nWR is number of Warning messages

  nSIB10 is number of SIB10 messages in log
- timestamp $SIB10TimeLog_j$ occurs in ϕ
- $F_i^{min} = min_{\{1 \le j \le nSIB10\}}\{j \mid WRTime_i - 1000 <= SIB10TimeLog_j <= WRTime_i + 1000\}$

  $WRTime_i$ is timestamp of ith Warning message
- $(\forall\ 1 \le i \le nWR)(\exists\ F_i^{min} \le j \le F_i^{max}).\phi$

Analysis of real logs

- We used model to analyze small real logs, found in internal archive.
- The size of logs is between 138KB and 578KB.
- The running time for the Model was a few seconds for each log.
- The objective was between 0 and 70.

Analysis of correct generated logs

		log1	log2	log3	log4	log5
	nPrim	30	25	20	15	10
	nSec	30	25	20	15	10
PagingModel	nPagLog	625	510	419	307	218
	time	0:03:59	0:02:39	0:00:22	0:00:11	0:00:07
	time,gecode	0:00:04	0:00:02	0:00:02	0:00:01	0:00:01
	objective	0	0	0	0	0
SIB1Model	nSIB1Log	625	510	419	307	218
	time	0:04:21	0:02:56	0:00:42	0:00:21	0:00:11
	time,gecode	0:00:04	0:00:03	0:00:02	0:00:01	0:00:01
	objective	0	0	0	0	0
PrimSecModel	nSIB10Log	3062	2628	2091	1435	1057
	nSIB11Log	2753	2209	1500	1052	867
	time	0:16:47	0:12:46	0:03:49	0:02:13	0:01:27
	time,gecode	0:00:05	0:00:03	0:00:02	0:00:02	0:00:01
	objective	0	0	0	0	0
Model	time	0:20:59	0:14:31	0:04:46	0:02:45	0:01:43
	time,gecode	0:00:05	0:00:04	0:00:03	0:00:02	0:00:02
	objective	0	0	0	0	0

Analysis of generated logs with injected errors

		log6	log7	log8	log9	log10	log11	log12	log13	log14	log15
	nPrim	20									
	nSec	20									
	errors in messages		r,a paging	t paging	r,c SIB1	t,c SIB1	r SIB10	t,c SIB10	r,t,c SIB11	r,a,t,c all	r,a,t,c all
PagingModel	nPagLog	374	174	374	374	374	374	374	374	374	398
	time	0:00:23	0:00:27	0:01:23						0:00:58	0:23:15
	time,gecode	0:00:02	0:00:08	0:01:02						0:00:39	0:22:56
	objective	0	413	155						192	789
SIB1Model	nSIB1Log	374	374	374	75	374	374	374	374	383	473
	time	0:00:39	0:00:41	0:01:30	0:01:07	0:04:23				0:01:22	0:32:16
	time,gecode	0:00:02	0:00:06	0:00:50	0:00:38	0:03:44				0:00:45	0:31:33
	objective	0	413	155	684	395				304	1386
PrimSecModel	nSIB10Log	1875	1916	1994	1871	1858	406	1922	1888	1683	2034
	nSIB11Log	1430	1429	1429	1429	1429	1430	1429	1143	1295	1242
	time	0:03:38	0:03:45	0:04:49			0:03:28	0:07:54	0:04:11	0:06:24	1:25:55
	time,gecode	0:00:03	0:00:08	0:00:58			0:01:00	0:04:11	0:00:59	0:02:57	1:22:21
	objective	0	413	155			220	3611	2061	4267	6789
Model	time	0:05:27	0:04:48	0:06:04	0:04:42	0:08:38	0:04:22	0:08:34	0:05:14	0:07:12	1:40:13
	time,gecode	0:00:03	0:00:09	0:01:04	0:00:45	0:03:40	0:01:00	0:03:46	0:01:01	0:02:46	1:35:18
	objective	0	413	155	684	395	220	3611	2061	4379	7396

Conclusion

- The real log analysis took a few seconds.
- Analysis of large logs depended on number and type of errors.
- Constraint solver was able to handle big domains of parameters.
- Constraint solver can easily handle complex requirements on timestamps.
- Constraint programming can be used to analyze small logs if some parameter is unknown.
- Protocol log analysis with constraint programming can be a part of test automation and can be useful for functional testing as well as in regression testing.