

# Verslag Tinlab Advanced Algorithms

T. Ravensbergen  
G. Bartes  
K. G. Razmjou  
69

1 juni 2023



# Inhoudsopgave

<b>1</b>	<b>Inleiding</b>	<b>5</b>
1.1	Uppaal . . . . .	5
1.2	Statistical model checking . . . . .	5
1.3	Het vier variabelen model . . . . .	5
1.3.1	Monitored variabelen . . . . .	5
1.3.2	Controlled variabelen . . . . .	5
1.3.3	Input variabelen . . . . .	5
1.3.4	Output variabelen . . . . .	5
1.4	Literatuuronderzoek . . . . .	5
1.5	Conclusie . . . . .	6
<b>2</b>	<b>Requirements</b>	<b>6</b>
2.1	Requirements . . . . .	6
2.2	Sluisdeuren . . . . .	7
2.3	Stoplichten . . . . .	7
2.4	Waterpomp . . . . .	7
2.5	Boten . . . . .	7
2.6	Specificaties . . . . .	8
2.7	Notities die verwerkt moeten worden . . . . .	8
<b>3</b>	<b>Modellen</b>	<b>9</b>
3.1	De Kripke structuur . . . . .	9
3.2	Maincontroller . . . . .	9
3.3	Schip . . . . .	10
3.4	Sluis . . . . .	10
3.5	Stoplicht . . . . .	11
3.6	Deur . . . . .	11
3.7	Soorten modellen . . . . .	12
3.8	Tijd . . . . .	12
3.9	Guards en invarianten . . . . .	12
3.10	Deadlock . . . . .	12
3.11	Zeno gedrag . . . . .	12
<b>4</b>	<b>Logica</b>	<b>12</b>
4.1	Propositielogica . . . . .	12
4.2	Predicatenlogica . . . . .	12
4.3	Kwantoren . . . . .	12
4.4	Dualiteiten . . . . .	12
4.5	Proposities . . . . .	12
<b>5</b>	<b>Computation tree logic</b>	<b>16</b>
5.1	De computation tree . . . . .	16
5.2	Operator: AG . . . . .	16
5.3	Operator: EG . . . . .	16

5.4	Operator: EG	17
5.5	Operator: AF	17
5.6	Operator: EF	17
5.7	Operator: AX	17
5.8	Operator: EX	17
5.9	Operator: $p \cup q$	17
5.10	Operator: $p \cap q$	17
5.11	Operator: EX	17
5.12	Operator: $p \cup q$	17
5.13	Operator: $p \cap q$	17
5.14	Fairness	17
5.15	Liveness	17
<b>6</b>	<b>Testresultaten</b>	<b>18</b>
6.1	Inleiding	18
6.2	Resultaten	18
6.3	Conclusie	18
<b>7</b>	<b>Conclusie</b>	<b>19</b>
<b>8</b>	<b>Discussie</b>	<b>20</b>
8.1	Conclusie Galvin	20
8.2	Conclusie Tygo	20
8.3	Conclusie Koosha	20
<b>9</b>	<b>Eindverantwoording</b>	<b>21</b>

# 1 Inleiding

In deze case study wordt

## 1.1 Uppaal

## 1.2 Statistical model checking

Dit gaat in het algemeen over ...

## 1.3 Het vier variabelen model

Dit gaat in het specifiek over dit model

ware as an example.

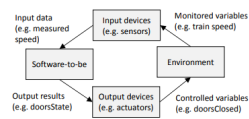


Figure 1: Four-variable model as illustrated in (van Lam-  
weerde, 2009)

### 1.3.1 Monitored variabelen

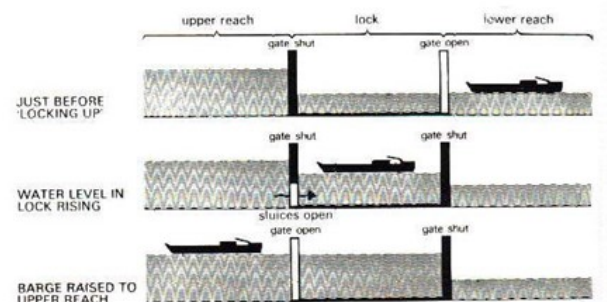
### 1.3.2 Controlled variabelen

### 1.3.3 Input variabelen

### 1.3.4 Output variabelen

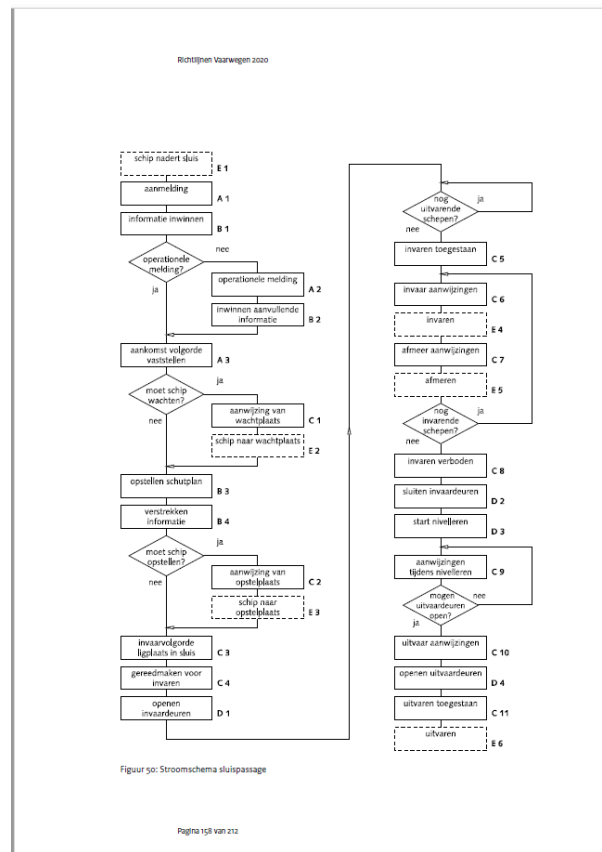
## 1.4 Literatuuronderzoek

Dit gaat over de literatuur die tijdens het project is verzameld en bestudeerd mbt de



opdracht. Hieronder een voorbeeld van een sluismodel

Hieronder een voorbeeld van de werking van en sluismodel volgens de richtlijn



vaarwegen.

## 1.5 Conclusie

## 2 Requirements

### 2.1 Requirements

Directe requirements van opdrachtgever:

Na grondige analyse van het Nederlandse sluisenpark is gebleken dat renovatie van een groot aantal sluisen noodzakelijk is. Een eerste verkenning heeft ontleend dat het gecombineerd renoveren en automatiseren van het Nederlandsesluisenpark een aanzienlijke verbetering kan opleveren t.a.v.:

- veiligheid
- efficiëntie
- capaciteit
- onderhoudskosten
- duurzaamheid

In het kader van het onlangs afgesloten klimaatakkoord heeft de Nederlandseoverheid daarom besloten over te gaan tot een ingrijpende renovatie van dediverse slui-

zen die ons land rijk is. Op het ministerie van infrastructuur en waterstaat is helaas onvoldoende kennis van ict en systemen aanwezig om eenen ander uit te voeren. Wij vragen u een model (of een onderling samenhangend aantal modellen)aan te leveren, opdat ontwerpen van verschillende, volledig geautomatiseerde sluizen in de toekomst gerealiseerd kunnen worden.

Eigen inbreng van deze requirements:

Wij gaan er van uit dat het volgende van ons verwacht wordt:

Maak een model dat als template dient gebruikt te worden voor het automatiseren van verschillende soorten sluizen. Verder moeten overwegingen gemaakt worden die goed onderbouwd zijn.

Aangezien er van ons alleen een model verwacht wordt, zullen wij ons geheel focus-  
sen op de fundamentele werking van de sluis en hierbij zullen wij ons dus niet bezig  
houden met fysieke eisen zoals veiligheidshekjes en borden. Onze focus ligt geheel  
op de werking van de sluis; elke state waar de sluis zich in mag bevinden en welke be-  
slissingen de sluis moet maken op basis van bestaande protocols en benoemde eisen.

Deze requirements zullen hieronder uitgewerkt worden, per sluisonderdeel, deze be-  
staande uit de sluisdeuren, de stoplichten, de waterpomp en de boten.

## **2.2 Sluisdeuren**

De sluisdeuren.

## **2.3 Stoplichten**

De stoplichten

## **2.4 Waterpomp**

De waterpomp

## **2.5 Boten**

De meeste sluizen die zich in Nederland bevinden zijn schutsluizen; deze sluizen zijn bedoeld om boten, zowel vrachtschepen als pleziervaart afhankelijk van de locatie van de sluis, te verwerken. Om deze reden gaan wij deze dus ook verwerken in ons model. Mocht een sluis niet bedoeld zijn om boten te verwerken, dan zou dit model alsnog toegepast kunnen worden opp desbetreffende sluis. Boten worden toegevoed aan de queue. Hoe dit gebeurt, dat ligt aan de specifieke sluis. Sinds wij een template maken, hoeven wij geen rekening te houden met hoe de schepen in de queue komen. Het enige wat wij hoeven te doen, is de data verwerken.

Overige eisen op basis van eigen inbreng:

## 2.6 Specificaties

Vanuit deze requiremenst kunnen verdere specificaties opgesteld worden.

Even ter duidelijkheid: een requirement beschrijft wat een programma moet doen, en een specificatie beschrijft hoe men van plan is om deze requirements te realiseren.// Voorbeeld:// Requirement is dat de sluis meerdere boten moet kunnen verwerken; de specificatie zou hier zijn fdat de sluis minstens twee keer zo groot moet zijn dan de grootste boot die door de sluis kan.

## 2.7 Notities die verwerkt moeten worden

moet de intital state altijd in een loop zitten in uppaal? wat zijn urgent channels? rampen? er staat wel iets in de planning maar kan geen lessen of verdere documentatie of requirements terug vinden?

gesprek wessel: main controller slim dat direction een bool is. pomp is te slim, zoiu alleen maar aan of uit moeten gaan, of nog weg en in pompen maar meer niet. niets met waterlevel en aantal schepen. schip: niet doen. als een schip zich aanmeld, dan gebeuren er dingen, maar gaat hij naar binnen? je weet niet wat dat schip gaat doen want menselijk gedrag. beter niet het schip uitgebreid maken, maar eerder de sluis. te veel aannames.

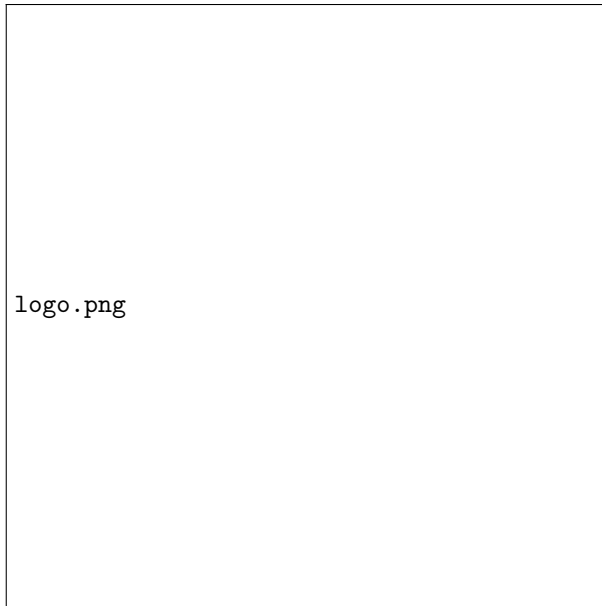
wessel model: alleen als wachtrij vol zit, doet de sluis iets. deur heeft een parameter zodat er meerdere deuren in de simulator neergezet kunnnen worden. ook bij wachtrij.

stoplichen kunnen er wel in maar als je simpeler wilt, gaan die als eerste weg. zes variabelen model is voorgesteld maar niet goed op gereageerd. alleen er van af weten is genoeg. rampen alleen voor persoonlijk verslag

## 3 Modellen

### 3.1 De Kripke structuur

### 3.2 Maincontroller

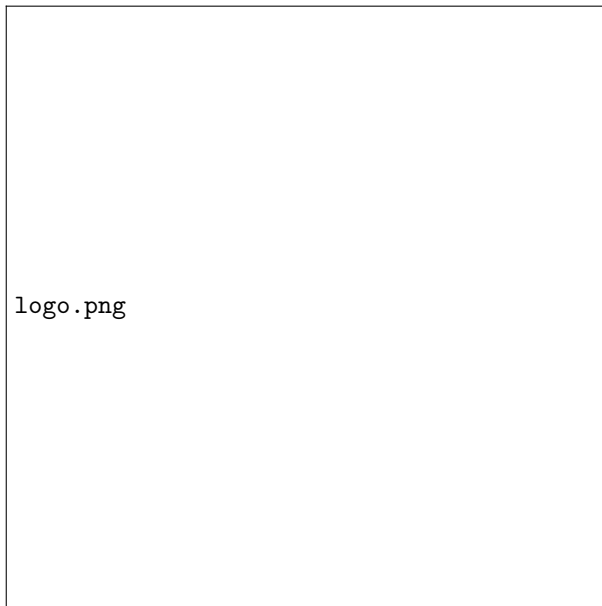




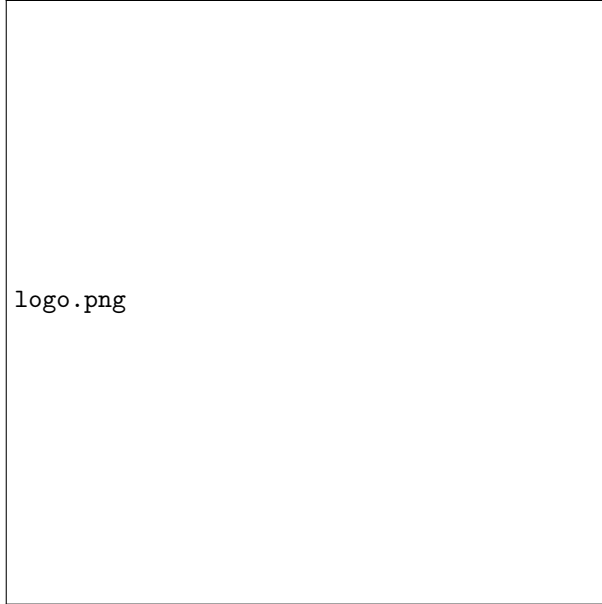
### 3.3 Schip



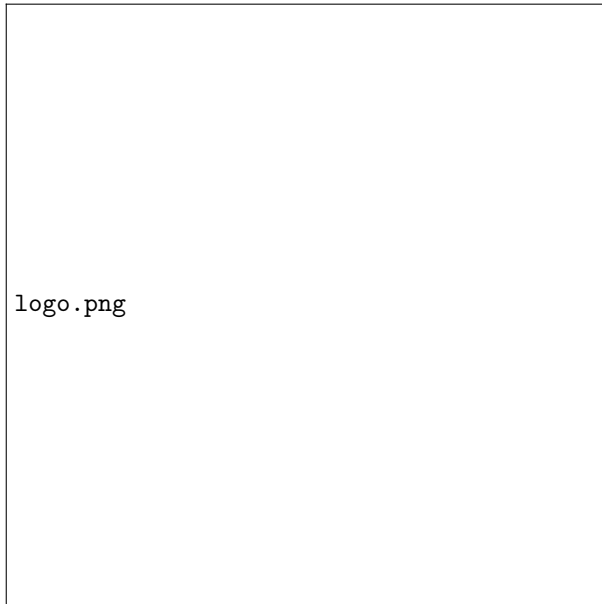
### 3.4 Sluis



### 3.5 Stoplicht



### 3.6 Deur



### 3.7 Soorten modellen

### 3.8 Tijd

### 3.9 Guards en invarianten

### 3.10 Deadlock

### 3.11 Zeno gedrag

## 4 Logica

### 4.1 Propositielogica

### 4.2 Predicatenlogica

### 4.3 Kwantoren

### 4.4 Dualiteiten

### 4.5 Propositions

- P1 Het is mogelijk dat de sluis van richting verandert.  $\exists j \neg \text{Main.Direction}$
- P2 Het is mogelijk dat de sluispomp in een cyclus teveel water heeft gepompt en dat er daardoor water weggepompt dan wel bijgekompt dient te worden  $\exists j \neg \text{main.waterlevel}$
- P3 Het is al binnen 100 ms mogelijk omte achterhalen aan welke kant de sluisdeuren open moeten.
- P4 Als de richting van een schip gelijk is aan N, dan is het waterlevel niet gelijk aan 1-5 of R
- P5 De sluispomp is nooit in positie AAN, wanneer de sluisdeuren open zijn.
- P6 In het geval dat er geen errors zijn ( in de stoplichten, sluisdeuren) and ideal (wachtrij) scenario,
  - a) dan is een cyclus gegarandeerd binnen 100 ms (including 100 ms) (undefined)
  - a') dan is een cyclus niet gegarandeerd binnen 100 ms
  - b) dan is het onmogelijk om van beneden naar boven te varen, of andersom binnen 150 ms
  - b') dan is het mogelijk om van beneden naar boven te varen, of andersom binnen 150 ms
  - c) het is onmogelijk om van richting te veranderen in minder dan 400 ms als de pomp al op niveau x is

- c') het is mogelijk om van richting te veranderen in minder dan 400 ms als de pomp al op niveau x is
- P7 Als zich geen errors voordoen bij stoplicht en deur, maar de waterpomp uitvalt:
  - a) a gear switch is gearanteerd after 1055 ms ( not including 1055) (deleted)
  - a') it is impossible to switch gear in 1055 ms (deleted)
  - b) it is impossible to switch gear in less than 550 ms (deleted)
  - b') it is possible to switch gear at 550 ms (deleted)
  - c) it is impossible to switch gear in less than 700 ms if the switch is not from/to gear N (deleted)
  - c') it is possible to switch gear at 700 ms if the switch is not from/to gear N (deleted)
- p8 When no error occurs, but engine fails to find synchronous speed
  - a) a gear switch is guaranteed in 1205 ms (including 1205)
  - a') a gear switch is not gearanteerd at less than 1205 ms
  - b) it is impossible to switch gear in less than 450 ms
  - b') it is possible to switch gear at 450 ms
  - c) it is impossible to switch gear in less than 750 ms if the switch is not from/to gear N
  - c') it is not possible to switch gear at 750 ms if the switch is not from/to gear N
- p9 Clutch errors
  - a) If the clutch is not closed properly (i.e. a timeout occurs) the gearbox controller will enter the location CCCloseError with 200 ms (undefined)
  - b) When the gearbox controller enters location CCloseError, there is always a problem in the clutch with closing the clutch. (undefined)
  - a) If the clutch is not closed properly (i.e. a timeout occurs) the gearbox controller will enter the location CCloseError within 200 ms (undefined)
  - b) When the gearbox controller enters location CCloseError, there is always a problem in the clutch with closing the clutch. (undefined)
- p10 Gearbox errors
  - a) If the gearbox can not enter a requested gear ( i.e. a timeout occurs) the gearbox controller will enter the location GsetError within 350 ms (undefined)

- b) When the gearbox controller enters location GSetError, there is always a problem in the gearbox with setting the gear. (undefined)
- p11 IF no error occurs in the engine, it is guaranteed to find synchronous speed (undefined)
- p12 Wanneer beide sluisdeuren in state gesloten zijn, dan is de pomp in zijn initiale state of 100 ms verwijderd van zijn initiele state
- A[]
- p13 When the gear controller has a greater set, torque regulaton is always indicated in the engine (undefined)
- A[]
- p14
- a) Als de deur open is(ongeacht boven of beneden, dan bevind de sluispomp zich in een predefined state (undefined) A[] (gate(0).open——gate(1).open) -  
 $\neg$  (main.pomp<sub>i</sub>dle||main.pomp2<sub>i</sub>dle)b) Als de deur is gesloten dan bevind de maincontroller zich in een predefined state (undefined) A[] (gate(0).closed——gate(1).closed) -  
 $\neg$  (main.pomp<sub>i</sub>dle||main.pomp2<sub>i</sub>dle) main.idle
- p15
- p16 If engine regulation is on torque, then the clutch is closed (undefined) A[] (Engine.Torque == torque && Clutch.closed == false)
- p17 Voor invaren geldt altijd: waterlevel, pomp uit, sluisdeuren open en stoplicht op groen A[] main.s5 - $\neg$  main.waterlevel<sub>1</sub>aagidle<sub>pomp1</sub>gate(0).open<sub>gate(1).open</sub>(stoplight(0).green<sub>stoplight(1).green</sub>)  
 $\neg$  forall(i : id<sub>a</sub>)forall(j : id<sub>s</sub>)gate(i).closed<sub>stoplight(j).rood</sub>main.rd<sub>1</sub>p19uitvarendenhebbenvoorangopinvarenden
- p20 Voor invarenden geldt pomp uit, sleusdeur open en stoplicht op groen A[] main.s6 - $\neg$  gate(0).open gate(1).open stoplight(0).groen stoplight(1).groen
- p21 voor nivelleren geldt pomp is aan, sluisduren zijn doicht en het stoplicht is op rood A[] (main.rn1 —— main.rn2) - $\neg$  forall (i:id<sub>a</sub>)forall(j : id<sub>s</sub>)gate(i).closed<sub>stoplight(j).rood</sub>p22Alseenschipver
- p23 urgent locations; het is niet mogelijk om hier te wachten
- p24 urgent syn; een synchronisatie moet direct worden uitgevoerd als de guards geldig zijn
- p25 als een schip binnen is, en er zijn wachtende schepen, dan moet het stoplicht via oranje naar rood A[]
- p26 committed; als deze staat actief is dan wordt de eerst volgende transitie uitaande van deze state

- p27 als een schip binnen vaart moet hij ook oft binnen zijn en niet binnenvaren, dit geldt ook voor p28 sluisdeuren en pompen dus deze zijn committed.  $A[]$
- p28 Een schip komt aanvaren en geeft een signaal aan de sluis.  $A[]$
- p29 Indien er meer dan twee schepen in de sluis zitten dan wordt het ship geplaatst in de wachrij.  $A[] \text{ Queue.list}[N-1] == 2 \rightarrow ( \text{Sluiskolk.list}[N] == 1 \rightarrow \text{Sluiskolk.list}[N] == 2 )$
- p30 Een schip kan pas naar binnenrijden als de sluisdeuren open zijn, het stoplicht is op groen er er zijn minder dan 2 schepen in de sluis.  $A[] \text{ main.s6 schip.varen } \rightarrow \text{Queue.list}[N-1] \leq 2$
- p32 Eenmaal in de sluis zal het schip moeten wachten op de sluis en de pomp.  $A[] \text{ Queue.list}[N-1] == 2$
- p33 Een schip mag alleen uitvaren als de pomp klaar is, de sluisdeuren open.  $A[] \text{ schip.varen main.s12 } \rightarrow \text{main.s13 } \rightarrow (!\text{main.rn1 } \wedge \text{main.rn2})$
- p34 Een sluis ontvangt een aankomst signaal van een schip en bestuurt de sluisdeuren en de pomp.  $A[]$
- p35 De sensor is een onderdeel van de sluis en ontvangt signalen van naderende schepen.  $A[]$
- p36 De sluisdeur voor boven en beneden kunnen beiden open en dicht. De sluisdeur wordt aangestuurd door de sluis.  $A[]$
- p37 Een pomp begint met pompen bij een signaal van de sluis. Een sluis op zijn beurt geeft alleen een signaal aan de pomp als de sluisdeuren dicht zijn  $A[] \text{ pomp.pomp\_active} \rightarrow \text{main.s6 forall}(i : \text{id}_a) \text{gate}(i).closed \rightarrow \text{p38 Geendeadlock}$
- p39 Voor geen enkel pad geldt dat als de deuren gesloten zijn volgens de sluis dat er een deur openstaat om een schip naar buiten te laten.  $A[] \text{ not forall}(i : \text{id}_a) \text{gate}(i).closed \rightarrow ( \text{main.s12} \vee \text{main.s13} ) \rightarrow \text{p40 Voor alle paden geldt dat als een sluis aan het voorbereiden is, dan zijn alle deuren dicht. } A[] \text{ not forall}( \text{gate}(0).closed$
- p41 Voor alle paden geldt dat als een deur dicht is het aantal schepen in de kade gelijk is aan nul  $A[]$  p42 Voor geen enkel pad geldt dat als het binnenstoplicht op groen staat dat het niet toegestaan is naar binnen te varen  $E \exists i \text{ stoplight}(2).groen \rightarrow \text{stoploght}(3).groen } \rightarrow i \text{ main.s6}$
- p43 Voor alle paden geldt dat de globale tijd langer is dan 30 tijdseenheden  $A[] \text{ main.s13 } \rightarrow i \text{ main.processtime } \leq 30$
- p44 Er is een pad waarvoor geldt dat als een schip wilt stoppen dat er meer dan 5 schepen in de sluis zitten.  $E \exists i$
- p45 Voor alle paden geldt als schip vertrekt is sluisdeur dicht  $A[]$

- p46 Voor alle paden geldt als stoplicht op rood sluisdeuren dicht en schip vertrokken dan is de nivelleermachine uit  $A[]$
- p47 Er is geen pad waarop een schip vertrekt vanuit de rechtersluisdeur en de linkersluisdeur is open en linkeruitvaartstoplicht en linkeruitvaartsoplicht opgroen en nivelleermachine is aan  $E_i \dot{z}$
- p48 Er is een pad waarvoor geldt dat linkersluisdeuren dicht zijn, rechtersluisdeuren dicht zijn rechteruitvaartstoplicht is rood en rechteruitvaartsoplicht is rood terwijl er geen schip in de sluis licht  $E_i \dot{z}$
- p49 Een stoplicht staat altijd op groen als de deuren open staan en de pomp niet bezig is.  $A[] \text{ forall } (i:id_s) \text{ stoplight.groen} \rightarrow \text{gate}(0).open \text{gate}(1).open (main.pomp1_i dle || main.pomp2_i dle) p50 Inge$
- p51 Voor alle paden in een pomp geldt dat als water level lager is dan waterlaag pompwaterweg is altijd false  $A[] (main.waterlevel_i waterlaag) \rightarrow (!pompwaterweg \rightarrow \text{pompwaterweg} == \text{false})$
- p52 Voor alle paden geldt dat als water level hoger is dan waterhoog dan is pompwater altijd false  $A[]$
- p53 Het zal nooit gebeuren dat een pomp water toevoegt als deuren open zijn, geen schip in sluis en stoplicht op groen  $A[] \text{ not } main.rn1 \rightarrow \text{main.rn2} \rightarrow \text{gate}(0).open \text{gate}(1).open \text{ Queue.list[N-1]} == 0 ((\text{stoplight}(0).groen \rightarrow \text{stoplight}(1).groen) \rightarrow (\text{stoplight}(3).groen \text{ stoplight}(4).groen))$
- p54 Het kan gebeuren dat bij pompr het stoplicht op rood staat, het schip in de sluis en deur is dicht, en waterstand gelijk aan waterlaag  $E_i \dot{z} (main.blocked1 \rightarrow \text{main.blocked2}) \rightarrow \text{Queue.list[N-1]} \dot{z} 0 \text{ gate}(0).closed \text{ gate}(1).closed \text{ main.waterlevel} == \text{main.waterlevel}_i \text{ aagp55 Eris}$   
 $main.rn1 || main.rn2 \rightarrow \text{gate}(0).closed \text{ main.waterlevel} == \text{waterlaag}$
- p56 Het kan voorkomen dat bij state pompaan het waterniveau gelijk is aan waterlaag  $E_i \dot{z} \text{ main.rn1} \rightarrow \text{main.rn2} \rightarrow \text{main.waterlevel} == \text{main.waterlaag}$
- p57 Voor alle paden geldt dat er een mogelijkheid is dat deur is open/dicht en sluis nivelleert omhoog/omlaag  $A[] \text{ gate}(0).open () \text{ main.direction} == 0 \rightarrow \text{main.direction} == 1)$
- p58  $A[] (1 \dot{z} 0)$

## 5 Computation tree logic

### 5.1 De computation tree

### 5.2 Operator: AG

### 5.3 Operator: EG

Voor alle paden geldt dat het waterlevel lager is dan het niveau van de kant. (???)  
 Voor alle paden geldt dat een pomp alleen werkzaam is als alle sluisdeuren dicht

zijn. Voor alle paden geldt dat het aantal schepen in de sluis maximaal 2 is. Voor alle paden geldt dat een schip nooit langer dan 30 seconden in een sluiskolk zit zonder dat het waterpeil is aangepast.

#### **5.4 Operator: EG**

Er bestaat op elk pad een

#### **5.5 Operator: AF**

#### **5.6 Operator: EF**

Er is een mogelijkheid dat twee schepen in de sluis een verschillende uitvaarrichting hebben. (Hoe?)

#### **5.7 Operator: AX**

#### **5.8 Operator: EX**

#### **5.9 Operator: $p \cup q$**

#### **5.10 Operator: $p \cap q$**

Voor alle paden geldt dat een schip alleen kan invaren als de sluisdeur aan de andere zijde is gesloten.

#### **5.11 Operator: EX**

Er bestaat geen situatie waar een pomp actief is terwijl er een sluisdeur open staat

#### **5.12 Operator: $p \cup q$**

Vanaf aankomst tot uitvaren is de clocktijd lager dan 30 tijdseenheden

#### **5.13 Operator: $p \cap q$**

Vanaf het invaren tot en met het uitvaren van een schip en geldig is  $x$  lager dan 15 tijdseenheden???? vanaf aanvaren staat een schip maximaal 40 tijdseenheden in de wachtrij.

#### **5.14 Fairness**

Definitie

#### **5.15 Liveness**

Definities



## **6 Testresultaten**

### **6.1 Inleiding**

Inleiding

### **6.2 Resultaten**

### **6.3 Conclusie**

data.txt

## 7 Conclusie

## 8 Discussie

### 8.1 Conclusie Galvin

### 8.2 Conclusie Tygo

### 8.3 Conclusie Koosha

## 9 Eindverantwoording

## Referenties

[1] Inside the cunning, unprecedented hack of ukraine's power grid.

[2] title.