2. $A = X \to P(Y)$ $B = Pow(X \times Y)$

Let $X = \{x_1, x_2, \cdots x_n\}$, $Y = \{y_1, y_2, \cdots y_m\}$ $(n, m \geq 0)$

Define $f: A \to B$: for any $a: X \to P(Y) \in A$, a is a binary relation where $a = \{(x_1, \{y_{11}, y_{12}, \cdots y_{1a_1}\}), (x_2, \{y_{21}, y_{22}, \cdots y_{2a_2}\}), \cdots, (x_n, \{y_{n1}, y_{n2}, \cdots y_{na_n}\})\}$

where $a_1, \cdots a_n \geq 0$, it means ϕ when $a_i = 0$

let $f(a) = \{(x_1, y_{11}), (x_1, y_{12}) \cdots (x_1, y_{1a_1}), (x_2, y_{21}), (x_2, y_{22}) \cdots, (x_2, y_{2a_2}), \cdots (x_n, y_{na_n})\}$

We need to show f is bijective.

Injective: Consider two diffrent element $a_1, a_2 \in A$

$a_1 \neq a_2$ means at least one element from both two binary relation sets are different, which means an element for X. let's assum x, is mapped to different elements in $P(Y)$ by a_1, a_2.

Assme $a_1(x) = \{y_{a_{11}}, y_{a_{12}}, \cdots, y_{a_{1j}}\}$, $a_2(x) = \{y_{a_{21}}, y_{a_{22}}, \cdots y_{a_{2k}}\}$, $j, k \geq 0$, $a_1(x) \neq a_2(x)$

then $f(a_1) = \{\cdots, (x, y_{a_{11}}), (x, y_{a_{12}}), \cdots (x, y_{a_{1j}}), \cdots\}$

$f(a_2) = \{\cdots, (x, y_{a_{21}}), (x, y_{a_{22}}), \cdots (x, y_{a_{2k}}), \cdots\}$

so, $f(a_1) \neq f(a_2)$ since the binary pairs of x are different.

so f is injective.

Surjective: $|A| = (2^m)^n = 2^{mn}$ (each element in X has 2^m possible maps.)

$|B| = 2^{m \times n} = 2^{mn}$; $|A| = |B|$

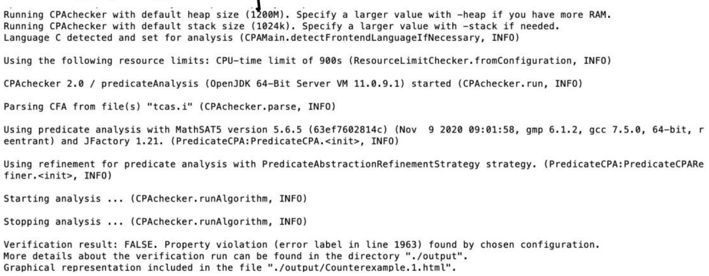
sice f is injective, then every element in A is mapped to different element in B, so at least 2^{mn} elements are in the image of f. Sice B only has 2^{mn} elements, then every element in B should be in the image of f.

so the codomain of f is equal to the image.

so f is surject.

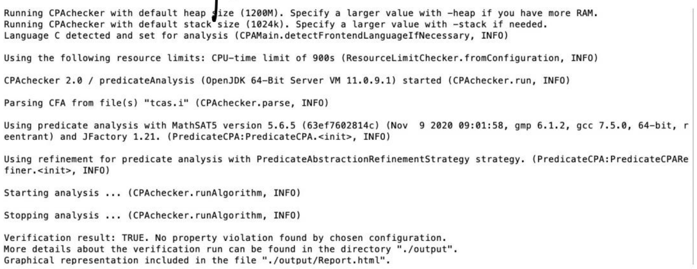
so f is bijective $\Longleftrightarrow$ there is 1 to 1 correspondace between sets A and B.

3. Screenshot for 1a



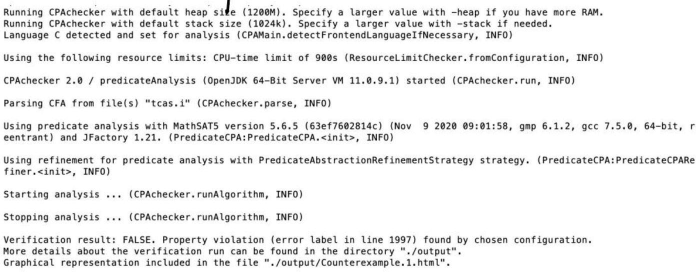
```
Running CPAchecker with default heap size (1200M). Specify a larger value with -heap if you have more RAM.
Running CPAchecker with default stack size (1024k). Specify a larger value with -stack if needed.
Language C detected and set for analysis (CPAMain.detectFrontendLanguageIfNecessary, INFO)

Using the following resource limits: CPU-time limit of 900s (ResourceLimitChecker.fromConfiguration, INFO)

CPAchecker 2.0 / predicateAnalysis (OpenJDK 64-Bit Server VM 11.0.9.1) started (CPAchecker.run, INFO)

Parsing CFA from file(s) "tcas.i" (CPAchecker.parse, INFO)

Using predicate analysis with MathSAT5 version 5.6.5 (63ef7602814c) (Nov  9 2020 09:01:58, gmp 6.1.2, gcc 7.5.0, 64-bit, r
eentrant) and JFactory 1.21. (PredicateCPA:PredicateCPA.<init>, INFO)

Using refinement for predicate analysis with PredicateAbstractionRefinementStrategy strategy. (PredicateCPA:PredicateCPARe
finer.<init>, INFO)

Starting analysis ... (CPAchecker.runAlgorithm, INFO)

Stopping analysis ... (CPAchecker.runAlgorithm, INFO)

Verification result: FALSE. Property violation (error label in line 1963) found by chosen configuration.
More details about the verification run can be found in the directory "./output".
Graphical representation included in the file "./output/Counterexample.1.html".
```

Screenshot for 1b



```
Running CPAchecker with default heap size (1200M). Specify a larger value with -heap if you have more RAM.
Running CPAchecker with default stack size (1024k). Specify a larger value with -stack if needed.
Language C detected and set for analysis (CPAMain.detectFrontendLanguageIfNecessary, INFO)

Using the following resource limits: CPU-time limit of 900s (ResourceLimitChecker.fromConfiguration, INFO)

CPAchecker 2.0 / predicateAnalysis (OpenJDK 64-Bit Server VM 11.0.9.1) started (CPAchecker.run, INFO)

Parsing CFA from file(s) "tcas.i" (CPAchecker.parse, INFO)

Using predicate analysis with MathSAT5 version 5.6.5 (63ef7602814c) (Nov  9 2020 09:01:58, gmp 6.1.2, gcc 7.5.0, 64-bit, r
eentrant) and JFactory 1.21. (PredicateCPA:PredicateCPA.<init>, INFO)

Using refinement for predicate analysis with PredicateAbstractionRefinementStrategy strategy. (PredicateCPA:PredicateCPARe
finer.<init>, INFO)

Starting analysis ... (CPAchecker.runAlgorithm, INFO)

Stopping analysis ... (CPAchecker.runAlgorithm, INFO)

Verification result: TRUE. No property violation found by chosen configuration.
More details about the verification run can be found in the directory "./output".
Graphical representation included in the file "./output/Report.html".
```

Screenshot for 2b



```
Running CPAchecker with default heap size (1200M). Specify a larger value with -heap if you have more RAM.
Running CPAchecker with default stack size (1024k). Specify a larger value with -stack if needed.
Language C detected and set for analysis (CPAMain.detectFrontendLanguageIfNecessary, INFO)

Using the following resource limits: CPU-time limit of 900s (ResourceLimitChecker.fromConfiguration, INFO)

CPAchecker 2.0 / predicateAnalysis (OpenJDK 64-Bit Server VM 11.0.9.1) started (CPAchecker.run, INFO)

Parsing CFA from file(s) "tcas.i" (CPAchecker.parse, INFO)

Using predicate analysis with MathSAT5 version 5.6.5 (63ef7602814c) (Nov  9 2020 09:01:58, gmp 6.1.2, gcc 7.5.0, 64-bit, r
eentrant) and JFactory 1.21. (PredicateCPA:PredicateCPA.<init>, INFO)

Using refinement for predicate analysis with PredicateAbstractionRefinementStrategy strategy. (PredicateCPA:PredicateCPARe
finer.<init>, INFO)

Starting analysis ... (CPAchecker.runAlgorithm, INFO)

Stopping analysis ... (CPAchecker.runAlgorithm, INFO)

Verification result: FALSE. Property violation (error label in line 1997) found by chosen configuration.
More details about the verification run can be found in the directory "./output".
Graphical representation included in the file "./output/Counterexample.1.html".
```

Property 1a means reporting an error when the variable 'thresh' is no greater then 'Up_Separation' and greater than 'Dawn_Separation'. I think tsca.i is a good test case. It gives both violations or correct results to let the designer to evaluate the CPA checker. If tsca.i will never violates the properties, then it shows the CPA checker gives false positive. If it wonld, then it means CPAchecker successfully catch the problem of the model. Overall, CPAchecker is useful at this moment, it checks the propertes in very few seconds even this could be a very time-consuming problem.

1 HW0

- **0 pts** Correct

Peer Review ID: 61992969 — enter this when you fill out your peer evaluation via gradescope