УДК 004.415.52, 681.3

ДОВЕДЕННЯ ВЛАСТИВОСТІ КОРЕКТНОЇ РОБОТИ БАНКІВСЬКОЇ СИСТЕМИ ВИПЛАТИ ГРОШОВИХ ПЕРЕКАЗІВ

Ю.А. Остаповська, Т.В. Панченко, Н.В. Поліщук, М.О. Картавов

Застосовано метод доведення властивостей паралельних програм, що виконуються багатоекземплярно в режимі почергового покрокового переключення і взаємодіють через спільну пам'ять, для доведення властивості коректності банківської системи виплати грошових переказів. В роботі поставлено задачу, побудовано транзиційну систему для моделі зі спрощеним станом, сформульовано інваріант програми та проведено доведення істинності інваріанту над програмною системою у довільний момент часу. Зроблено висновки щодо зручності та адекватності застосування методу для доведення коректності паралельних систем.
Ключові слова: коректність програмного забезпечення, доведення часткової коректності, паралельна програма, interleaving, інваріант, IPCL, композиційно-номінативні мови, формальна верифікація

Применен метод доказательства свойств параллельных программ, которые выполняются многоэкземплярно в режиме поочередного пошагового переключения и взаимодействуют через общую память, для доказательства свойства корректности банковской системы выплаты денежных переводов. В работе поставлена задача, построена транзиционная система для модели с упрощенным состоянием, сформулирован инвариант программы и проведено доказательство истинности инварианта над программной системой в любой момент времени. Сделаны выводы о удобстве и адекватности применения метода для доказательства корректности параллельных систем.
Ключевые слова: корректность программного обеспечения, доказательство частичной корректности, параллельная программа, interleaving, инвариант, IPCL, композиционно-номинативные языки, формальная верификация

The method for properties proof for parallel programs running multiple-instance interleaving with shared memory was applied in order to prove the correctness property of the banking system for remittances payments. The task was stated, transitional system was built for the model with simplified state, and the program invariant was formulated and proved to keep true over the software system at any given time in this work. Conclusions about the convenience and adequacy of method application to prove the correctness of parallel systems were made.
Key words: software correctness, safety property proof, concurrent program, interleaving, invariant, IPCL, composition-nominative languages, formal verification

Постановка задачі та аналіз. Формулювання властивості коректності системи

Розглянемо застосування методики доведення властивостей паралельних програм у IPCL [1–3]. Задача стосується доведення критичних властивостей системи виплати міжнародних грошових переказів Vigo Remittance Corp., реалізованої в ВАТ "Державний ощадний банк України" (надалі – банк).

Сформулюємо задачу. Для виплати певного переказу одержувачу здійснюється авторизація – отримання дозволу на виплату цього переказу даному одержувачу (точніше, в даному випадку – претенденту на одержання). Така процедура виконується в середовищі SQL, модель роботи якого – паралелізм через спільну пам'ять в режимі почергового виконання, тобто кілька процедур виконуються "одночасно" в покроковому режимі, причому кожну дискрету часу виконується тільки одна команда тільки одного процесу (із усіх, запущених у паралель).

Процедура авторизації передбачає аутентифікацію та ідентифікацію претендента на одержання грошей шляхом перевірки наданих ним даних у запиті стосовно грошового переказу. Якщо інформація надана правильно, тобто співпадає з наявною у базі даних (база даних поповнюється паралельно з пересиланням грошей при надходженні нових переказів для виплати), переказ не заблоковано (переказ блокується після кількох невдалих авторизацій або з інших особливих обставин – анулювання, тощо – та може бути розблокований спеціальним оператором системи) та не виплачено, одержувач є дійсно тим, за кого себе видає, то авторизація вважається пройденою успішно і надається дозвіл на виплату переказаних грошей оператором системи (працівником банку) одержувачу. Ця процедура є необхідною (тобто попередній розподіл переказів по відділеннях банку є неможливим), оскільки повинна забезпечуватись можливість одержувачу отримати призначений йому грошовий переказ (тобто переказ на своє ім'я) у будь-якій точці (в будь-якому відділенні банку) країни.

Банку необхідно гарантувати, що система працює коректно, іншими словами, що він не втрачає кошти. Оскільки формалізувати ідентифікацію одержувача (людини) безглуздо (перевірка відповідності паспортних даних, порівняння по вклеєній у паспорт фотокартці, перевірка автентичності фотокартки, непідробленість паспорту взагалі тощо), зосередимось на ідентифікації переказу та його стану і відповідних кроках процедури авторизації. З цих позицій питання коректності, фактично, зводиться до питання неможливості виплати зайвих коштів. Якщо врахувати, що авторизація передбачає, зокрема, перевірку суми переказу, то зрозуміло, що в межах одного переказу банк не може втратити "зайвих" грошей. Звідси, банку потрібно гарантувати, фактично, що жодний переказ не може бути виплачений більше одного разу. (Певний переказ буде виплачений один раз лише за умови відповідності інформації в базі даних переказів з наданими претендентом на його одержання реквізитами.)

Звідси, коректність системи зводиться до необхідності довести, що один і той самий переказ не може бути виплачений двічі (і більше разів), тобто що дозвіл на його виплату (авторизація виплати) не може бути отриманий двічі (і більше).

Проектування

Тепер розглянемо детальніше процедуру авторизації. Легко побудувати більш-менш очевидну покрокову процедуру авторизації переказу в першому наближенні:

1) зафіксувати чергову спробу авторизації
2) якщо переказ не заблоковано то
якщо переказ не виплачено то
якщо надана коректна інформація по переказу то (*)
якщо ця інформація співпадає з даними у базі то (*)
авторизувати виплату переказу
інакше (в усіх інших випадках) відмовити в авторизації
3) якщо спроб авторизації накопичилось більше 2, то
заблокувати переказ.

Опустимо з аналізу перевірку коректності інформації (помічено *), оскільки реально це не впливає суттєво на результат авторизації (більш того, ці перевірки складають банківську таємницю) – так, це дуже важлива перевірка, але, як побачимо далі, вона не є критичною для доведення властивості, яка нас цікавить (іншими словами, цілком зрозуміло, як її робити). (Очевидно, що коли хоча б одна з умов не виконується, авторизація не пройде, оскільки вона відбувається лише у випадку, коли всі умови повертають True під час перевірки – принаймні логічно так має бути.) Як було відзначено, середовище SQL є середовищем з паралелізмом із взаємодією через спільну пам'ять. В зв'язку з цим слід також зауважити, що якщо деяка умова повернула True під час обчислення її значення, то пізніше (при виконанні наступних кроків) вона (той самий предикат умовного оператору) може потенційно прийняти значення False, отже коли виконання дійде до блоку "then" з "if-then-else", наприклад, то значення предиката умови може вже бути False (хоча до виконання блоку "then" можна було приступити лише в ситуації, коли при обчислення предикату умови було отримано результат True). А отже, ми можемо отримати суперечливу ситуацію і зокрема, наприклад, авторизувати виплату одного переказу двічі.

Не зосереджуючись докладно на архітектурних рішеннях та дизайні системи, відзначимо ключові думки та їх обґрунтування.

1. Вузьке місце. Інтуїтивно зрозуміло, що критичним місцем системи є паралельна спроба авторизації одного й того ж переказу кілька разів (наприклад, коли дехто перехопив коректну інформацію про переказ і намагається отримати авторизацію не будучи дійсним одержувачем переказу). Цей факт не впливає на подальшу побудову системи та доведення необхідної властивості, але дозволяє краще розуміти вузькі місця і приймати правильні рішення щодо дизайну та архітектури системи.

2. Рівень ізоляції транзакцій SQL (Transaction Isolation Level). Кожна авторизація складає транзакцію (насправді – дві транзакції: фіксація намірів авторизації та результат авторизації – власне авторизація). Рівень ізоляції обирається Read Committed. Зрозуміло, що нам необхідний рівень, який гарантував би неможливість одночасної роботи з одним переказом (фактично, не допускав би інтерференцію процесів), але, по-перше, найсуворіший рівень ізоляції – Serializable – не еквівалентний послідовному виконанню процесів (тоді була б гарантована свобода від інтерференції), а по-друге, рівень ізоляції Serializable рекомендований до застосування лише у випадках зі складною логікою обчислень (в даній системі логіка не є надто складною) і має великий відсоток відкатів транзакцій (transaction roll-back), що призводить до необхідності повторних спроб виконання транзакцій, ускладнення внутрішньої логіки виконання команд SQL-сервером (більше блокувань будуть утримуватись довший час) та зменшення загальної швидкодії системи (детальніше про це можна прочитати в документації по будь-якому SQL-серверу – Microsoft SQL Server, PostgreSQL, а також в [4]). Звичайно, останнє є неприпустимим у випадку критичних до часу банківських систем.

3. Уточнення алгоритму авторизації. Введемо перевірку на кількість паралельних (виконуваних одночасно) авторизацій. Це можливо, оскільки ми фіксуємо спробу авторизації в журналі, незалежно від її результату, який з'ясується пізніше.

Таким чином, уточнений алгоритм авторизації виглядатиме наступним чином:

1) записати чергову спробу авторизації
2) якщо кількість одночасних (паралельних) спроб = 1, то
якщо переказ не заблоковано то
якщо переказ не виплачено то
якщо надана коректна інформація по переказу і ця інформація співпадає з даними у базі то (*)
авторизувати виплату переказу
інакше (в усіх інших випадках) відмовити в авторизації
3) якщо спроб авторизації накопичилось більше 2, то
заблокувати переказ.

Паралельно з цим спеціальний оператор по вирішенню конфліктів може розблокувати спірні перекази, тобто паралельно виконується процес:

розблокувати переказ

Побудова програми IPCL та моделюючої транзиційної системи

Нам знадобиться модель системи (модель реальної системи на мові SQL в мові *IPCL*). Отже, одним з ключових моментів є визначення адекватного рівня деталізації системи (і, відповідно, доведення). Є різні можливості в даному аспекті – так, можна розглядати модель як в [5,6], тобто моделювати виконання операцій SQL-сервером (деталізувати до рівня операцій з таблицям), а можна абстрагуватись від представлення та роботи безпосередньо з даними – і розглядати функціонування системи на вищому рівні абстракції. Ми будемо покладатись на коректність реалізації SQL-сервера, а, отже, розглядати (точніше, використовувати) його операції без деталізації щодо їх реалізації – тобто розглядатимемо коректність програми відносно SQL.

Ще одне зауваження. В моделі ми зосередимось на роботі з одним переказом. Зрозуміло, що робота з різними переказами, хоч і виконувана в паралель, не інтерферує (немає взаємного впливу процесів авторизації різних переказів один на інший, адже такими процесами зачіпаються рядки таблиць, перетин яких є порожнім – оскільки вони мають принаймні різні коди переказів). Тому ми явно не будемо вказувати ні номер переказу, ні інформацію щодо нього.

Уточнення процедури авторизації. Нехай у нас є три таблиці: інформаційна (про перекази) *Inf*, авторизаційна (журнал авторизацій та спроб) *Auth* та таблиця з інформацією щодо блокувань *Block*. Авторизаційна таблиця містить кортежі зі станом спроб авторизації – “виплачено” (тобто авторизована виплата), “спроба авторизації” (зараз відбувається, in progress) та “відмова” (із зазначенням причини). Розглянемо ситуацію роботи з деяким переказом з кодом *Invoice*. Позначимо кількості рядків в цих таблицях наступним чином:

A – кількість виплат переказу *Invoice*, тобто кількість рядків таблиці *Auth*, які відносяться до *Invoice* та мають статус “виплачено”,

P – кількість паралельних (одночасних) спроб авторизації (виплати) переказу, тобто кількість рядків таблиці *Auth*, які відносяться до *Invoice* та мають статус “спроба авторизації”,

T – накопичувальна кількість всіх спроб виплати переказу, тобто загальна кількість рядків таблиці *Auth*, які відносяться до *Invoice*,

B – блокування переказу (0 = немає, 1 = є), тобто останній стан блокування *Invoice*, згідно таблиці *Block*, якщо такі записи є (1 – заблоковано або 0 – розблоковано), або 0, якщо немає відповідних записів в таблиці *Block*.

Зрозуміло, що значення всіх змінних завжди є невід’ємними цілими числами (це – кількості рядків в таблицях, окрім *B*, що приймає значення 0 або 1).

Тепер можна виписати процедуру авторизації у вигляді *IPCL*-програми над станом зі змінними *A, P, T, B*:

```
Auth_Prog =
    (T, P) := (T +1, P +1);// фіксація спроби авторизації
    if ( P = 1 ) then
        if ( B = 0 ) then
            if ( A = 0 ) then
                if ( data_provided_correct() ) then
                  (A, P) := (A +1, P –1)      // фіксація успішної авторизації
                else P := P –1                // завершення авторизації відмовою
            else P := P –1
        else P := P –1
    else P := P –1;
    if ( T > 2 ) then B := 1 else Id;
```

де *data_provided_correct*() – предикат, що повертає True, коли передані дані щодо переказу є коректними згідно бази даних переказів (таблиця *Inf*) та False у протилежному випадку.

Паралельно може виконуватись процес “розблокувати переказ”:

```
Unblock =
  B := 0;
```

Семантика відповідних функцій алгебри $IPCL_A$ буде визначена природним чином:

$sem_{(T,\,P)\,:=\,(T+1,\,P+1)}\,(d) \equiv d \nabla [\, T \mapsto (T \Rightarrow (d) + 1),\, P \mapsto (P \Rightarrow (d) + 1)\,],$

$sem_{P\,=\,1}\,(d) \equiv \mathbf{IF}\ (\, P \Rightarrow (d) = 1,\, True,\, False\,),$

$sem_{B=0}(d) \equiv$ **IF** ($B{\Rightarrow}(d) = 0$, *True*, *False*),

$sem_{A=0}(d) \equiv$ **IF** ($A{\Rightarrow}(d) = 0$, *True*, *False*),

$sem_{data\_provided\_correct()}(d) \equiv$ *choice* (True, False),

$sem_{(A,P):=(A+1,P-1)}(d) \equiv d \nabla [\, A \mapsto (A{\Rightarrow}(d) + 1),\ P \mapsto (P{\Rightarrow}(d) - 1)\,]$,

$sem_{P:=P-1}(d) \equiv d \nabla [\, P \mapsto (P{\Rightarrow}(d) - 1)\,]$,

$sem_{T>2}(d) \equiv$ **IF** ($T{\Rightarrow}(d) > 2$, *True*, *False*),

$sem_{B:=1}(d) \equiv d \nabla [\, B \mapsto 1\,]$,

$sem_{\mathrm{Id}}(d) \equiv d$ (identity, дане не змінюється),

$sem_{B:=0}(d) \equiv d \nabla [\, B \mapsto 0\,]$,

де *choice* (*A*, *B*) – недетермінований вибір між значеннями *A* та *B*. Дане *d* має загальний вигляд

$$d = [\, P \mapsto \mathrm{p},\ T \mapsto \mathrm{t},\ A \mapsto \mathrm{a},\ B \mapsto \mathrm{b}\,].$$

Таким чином, маємо програму в *IPCL* (точніше, в підкласі – *SeqILProgs*):

$Program = Auth\_Prog^n \parallel Unblock^m$.

Оскільки у нас немає локальних даних (точніше, для спрощення ми від них відмовились – зокрема, в *data_provided_correct*()), побудуємо спрощену модель (зі спрощеними станами) [7]. Після виконання алгоритму розстановки міток отримаємо:

Auth_Prog =
[*M*1:] (*T*, *P*) := (*T* +1, *P* +1);
if **[*M*2:]** (*P* = 1) **then**
if **[*M*3:]** (*B* = 0) **then**
if **[*M*4:]** (*A* = 0) **then**
if **[*M*5:]** (*data_provided_correct*()) **then**
[*M*6:] (*A*, *P*) := (*A* +1, *P* –1)
else **[*M*7:]** *P* := *P* –1
else **[*M*8:]** *P* := *P* –1
else **[*M*9:]** *P* := *P* –1
else **[*M*10:]** *P* := *P* –1;
if **[*M*11:]** (*T* > 2) **then** **[*M*12:]** *B* := 1 **else** **[*M*13:]** Id;
[*M*14:]
Unblock =
[*M*15:] *B* := 0;
[*M*16:]

Множина *SStates* буде представляти собою підмножину $N^{16}{\times}D$, згідно спрощеного варіанту методу, де $D = ND(V, W)$ – спільні глобальні дані для всіх процесів, причому $\{A, P, T, B\}{\subseteq}V$ та $N{\cup}\{0\}{\subseteq}W$.

Функція *SStep* (виконання програми *Program*) буде визначена за цим алгоритмом наступним чином:

$\{ (\, (s_1, s_2, s_3, s_4, s_5, s_6, s_7, s_8, s_9, s_{10}, s_{11}, s_{12}, s_{13}, s_{14}, s_{15}, s_{16}, d), (s_1{-}1, s_2{+}1, s_3, s_4, s_5, s_6, s_7, s_8, s_9, s_{10}, s_{11}, s_{12}, s_{13}, s_{14}, s_{15}, s_{16}, sem_{(T,P):=(T+1,P+1)}(d))\,) \mid s_1{>}0 \ \&\ \forall i{\in}N_{16} \bullet s_i{\in}N \} \cup$

$\{ (\, (s_1, s_2, s_3, s_4, s_5, s_6, s_7, s_8, s_9, s_{10}, s_{11}, s_{12}, s_{13}, s_{14}, s_{15}, s_{16}, d), (s_1, s_2{-}1, s_3{+}1, s_4, s_5, s_6, s_7, s_8, s_9, s_{10}, s_{11}, s_{12}, s_{13}, s_{14}, s_{15}, s_{16}, d)\,) \mid s_2{>}0 \ \&\ \forall i{\in}N_{16} \bullet s_i{\in}N \ \&\ sem_{P=1}(d){=}True \} \cup$

$\{ (\, (s_1, s_2, s_3, s_4, s_5, s_6, s_7, s_8, s_9, s_{10}, s_{11}, s_{12}, s_{13}, s_{14}, s_{15}, s_{16}, d), (s_1, s_2{-}1, s_3, s_4, s_5, s_6, s_7, s_8, s_9, s_{10}{+}1, s_{11}, s_{12}, s_{13}, s_{14}, s_{15}, s_{16}, d)\,) \mid s_2{>}0 \ \&\ \forall i{\in}N_{16} \bullet s_i{\in}N \ \&\ sem_{P=1}(d){=}False \} \cup$

$\{ (\, (s_1, s_2, s_3, s_4, s_5, s_6, s_7, s_8, s_9, s_{10}, s_{11}, s_{12}, s_{13}, s_{14}, s_{15}, s_{16}, d), (s_1, s_2, s_3{-}1, s_4{+}1, s_5, s_6, s_7, s_8, s_9, s_{10}, s_{11}, s_{12}, s_{13}, s_{14}, s_{15}, s_{16}, d)\,) \mid s_3{>}0 \ \&\ \forall i{\in}N_{16} \bullet s_i{\in}N \ \&\ sem_{B=0}(d){=}True \} \cup$

$\{ (\, (s_1, s_2, s_3, s_4, s_5, s_6, s_7, s_8, s_9, s_{10}, s_{11}, s_{12}, s_{13}, s_{14}, s_{15}, s_{16}, d), (s_1, s_2, s_3{-}1, s_4, s_5, s_6, s_7, s_8, s_9{+}1, s_{10}, s_{11}, s_{12}, s_{13}, s_{14}, s_{15}, s_{16}, d)\,) \mid s_3{>}0 \ \&\ \forall i{\in}N_{16} \bullet s_i{\in}N \ \&\ sem_{B=0}(d){=}False \} \cup$

$\{ ((s_1, s_2, s_3, s_4, s_5, s_6, s_7, s_8, s_9, s_{10}, s_{11}, s_{12}, s_{13}, s_{14}, s_{15}, s_{16}, d), (s_1, s_2, s_3, s_4-1, s_5+1, s_6, s_7, s_8, s_9, s_{10}, s_{11}, s_{12}, s_{13}, s_{14}, s_{15}, s_{16}, d)) \mid s_4>0 \ \& \ \forall i \in N_{16} \bullet s_i \in N \ \& \ sem_{A=0}(d)=True \} \cup$

$\{ ((s_1, s_2, s_3, s_4, s_5, s_6, s_7, s_8, s_9, s_{10}, s_{11}, s_{12}, s_{13}, s_{14}, s_{15}, s_{16}, d), (s_1, s_2, s_3, s_4-1, s_5, s_6, s_7, s_8+1, s_9, s_{10}, s_{11}, s_{12}, s_{13}, s_{14}, s_{15}, s_{16}, d)) \mid s_4>0 \ \& \ \forall i \in N_{16} \bullet s_i \in N \ \& \ sem_{A=0}(d)=False \} \cup$

$\{ ((s_1, s_2, s_3, s_4, s_5, s_6, s_7, s_8, s_9, s_{10}, s_{11}, s_{12}, s_{13}, s_{14}, s_{15}, s_{16}, d), (s_1, s_2, s_3, s_4, s_5-1, s_6+1, s_7, s_8, s_9, s_{10}, s_{11}, s_{12}, s_{13}, s_{14}, s_{15}, s_{16}, d)) \mid s_5>0 \ \& \ \forall i \in N_{16} \bullet s_i \in N \ \& \ sem_{data\_provided\_correct()}(d)=True \} \cup$

$\{ ((s_1, s_2, s_3, s_4, s_5, s_6, s_7, s_8, s_9, s_{10}, s_{11}, s_{12}, s_{13}, s_{14}, s_{15}, s_{16}, d), (s_1, s_2, s_3, s_4, s_5-1, s_6, s_7+1, s_8, s_9, s_{10}, s_{11}, s_{12}, s_{13}, s_{14}, s_{15}, s_{16}, d)) \mid s_5>0 \ \& \ \forall i \in N_{16} \bullet s_i \in N \ \& \ sem_{data\_provided\_correct()}(d)=False \} \cup$

$\{ ((s_1, s_2, s_3, s_4, s_5, s_6, s_7, s_8, s_9, s_{10}, s_{11}, s_{12}, s_{13}, s_{14}, s_{15}, s_{16}, d), (s_1, s_2, s_3, s_4, s_5, s_6-1, s_7, s_8, s_9, s_{10}, s_{11}+1, s_{12}, s_{13}, s_{14}, s_{15}, s_{16}, sem_{(A,P):=(A+1,P-1)}(d))) \mid s_6>0 \ \& \ \forall i \in N_{16} \bullet s_i \in N \} \cup$

$\{ ((s_1, s_2, s_3, s_4, s_5, s_6, s_7, s_8, s_9, s_{10}, s_{11}, s_{12}, s_{13}, s_{14}, s_{15}, s_{16}, d), (s_1, s_2, s_3, s_4, s_5, s_6, s_7-1, s_8, s_9, s_{10}, s_{11}+1, s_{12}, s_{13}, s_{14}, s_{15}, s_{16}, sem_{P:=P-1}(d))) \mid s_7>0 \ \& \ \forall i \in N_{16} \bullet s_i \in N \} \cup$

$\{ ((s_1, s_2, s_3, s_4, s_5, s_6, s_7, s_8, s_9, s_{10}, s_{11}, s_{12}, s_{13}, s_{14}, s_{15}, s_{16}, d), (s_1, s_2, s_3, s_4, s_5, s_6, s_7, s_8-1, s_9, s_{10}, s_{11}+1, s_{12}, s_{13}, s_{14}, s_{15}, s_{16}, sem_{P:=P-1}(d))) \mid s_8>0 \ \& \ \forall i \in N_{16} \bullet s_i \in N \} \cup$

$\{ ((s_1, s_2, s_3, s_4, s_5, s_6, s_7, s_8, s_9, s_{10}, s_{11}, s_{12}, s_{13}, s_{14}, s_{15}, s_{16}, d), (s_1, s_2, s_3, s_4, s_5, s_6, s_7, s_8, s_9-1, s_{10}, s_{11}+1, s_{12}, s_{13}, s_{14}, s_{15}, s_{16}, sem_{P:=P-1}(d))) \mid s_9>0 \ \& \ \forall i \in N_{16} \bullet s_i \in N \} \cup$

$\{ ((s_1, s_2, s_3, s_4, s_5, s_6, s_7, s_8, s_9, s_{10}, s_{11}, s_{12}, s_{13}, s_{14}, s_{15}, s_{16}, d), (s_1, s_2, s_3, s_4, s_5, s_6, s_7, s_8, s_9, s_{10}-1, s_{11}+1, s_{12}, s_{13}, s_{14}, s_{15}, s_{16}, sem_{P:=P-1}(d))) \mid s_{10}>0 \ \& \ \forall i \in N_{16} \bullet s_i \in N \} \cup$

$\{ ((s_1, s_2, s_3, s_4, s_5, s_6, s_7, s_8, s_9, s_{10}, s_{11}, s_{12}, s_{13}, s_{14}, s_{15}, s_{16}, d), (s_1, s_2, s_3, s_4, s_5, s_6, s_7, s_8, s_9, s_{10}, s_{11}-1, s_{12}+1, s_{13}, s_{14}, s_{15}, s_{16}, d)) \mid s_{11}>0 \ \& \ \forall i \in N_{16} \bullet s_i \in N \ \& \ sem_{T>2}(d)=True \} \cup$

$\{ ((s_1, s_2, s_3, s_4, s_5, s_6, s_7, s_8, s_9, s_{10}, s_{11}, s_{12}, s_{13}, s_{14}, s_{15}, s_{16}, d), (s_1, s_2, s_3, s_4, s_5, s_6, s_7, s_8, s_9, s_{10}, s_{11}-1, s_{12}, s_{13}+1, s_{14}, s_{15}, s_{16}, d)) \mid s_{11}>0 \ \& \ \forall i \in N_{16} \bullet s_i \in N \ \& \ sem_{T>2}(d)=False \} \cup$

$\{ ((s_1, s_2, s_3, s_4, s_5, s_6, s_7, s_8, s_9, s_{10}, s_{11}, s_{12}, s_{13}, s_{14}, s_{15}, s_{16}, d), (s_1, s_2, s_3, s_4, s_5, s_6, s_7, s_8, s_9, s_{10}, s_{11}, s_{12}-1, s_{13}, s_{14}+1, s_{15}, s_{16}, sem_{B:=1}(d))) \mid s_{12}>0 \ \& \ \forall i \in N_{16} \bullet s_i \in N \} \cup$

$\{ ((s_1, s_2, s_3, s_4, s_5, s_6, s_7, s_8, s_9, s_{10}, s_{11}, s_{12}, s_{13}, s_{14}, s_{15}, s_{16}, d), (s_1, s_2, s_3, s_4, s_5, s_6, s_7, s_8, s_9, s_{10}, s_{11}, s_{12}, s_{13}-1, s_{14}+1, s_{15}, s_{16}, sem_{Id}(d))) \mid s_{13}>0 \ \& \ \forall i \in N_{16} \bullet s_i \in N \} \cup$

$\{ ((s_1, s_2, s_3, s_4, s_5, s_6, s_7, s_8, s_9, s_{10}, s_{11}, s_{12}, s_{13}, s_{14}, s_{15}, s_{16}, d), (s_1, s_2, s_3, s_4, s_5, s_6, s_7, s_8, s_9, s_{10}, s_{11}, s_{12}, s_{13}, s_{14}, s_{15}-1, s_{16}+1, sem_{B:=0}(d))) \mid s_{15}>0 \ \& \ \forall i \in N_{16} \bullet s_i \in N \},$

де s_i – кількість процесів, що знаходяться в позиції (на мітці) $\mathbf{[}M_i\mathbf{:]}$ в даний момент часу.

Візьмемо

$$SStartStates = \{ (n, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, m, 0, d) \mid n \in N, m \in N, [P \mapsto 0, T \mapsto 0, A \mapsto 0, B \mapsto 0] \subseteq d \}$$

та

$$SStopStates = \{ s \mid s=(0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, n, 0, m, d') \ \& \ n \in N \ \& \ m \in N \ \& \ \exists s_1, s_2, \ldots, s_l \bullet$$

$$(s_1 \in SStartStates \ \& \ s_l=s \ \& \ (\forall i \in N_{l-1} \bullet (s_i, s_{i+1}) \in SStep)) \}.$$

В даній постановці модель можна розглядати як таку, що описує повний шлях існування (“життєвий цикл”) певного переказу (його спроби авторизації – вдалі та невдалі, блокування та розблокування спеціальним оператором). Оскільки *n* та *m* (ступені підпрограм) можуть бути довільними цілими невід’ємними числами, то, дійсно, модель задає довільний можливий шлях роботи з будь-яким одним переказом (іншими словами, сімейство програм). Обробка різних переказів не перетинається – такі процеси обробки переказів не інтерферують. До речі, програма *P*, зокрема, може перетворитись на послідовне виконання її підпрограм (в довільному ступені), згідно заданої семантики мови *IPCL* та алгоритму побудови моделі – така ситуація, мабуть, найчастіше виникає на практиці. Але разом з тим програма (і модель) передбачає можливість перетинатись довільним чином операторам (окремим діям) паралельних процесів (згідно заданої семантики – та принципів паралелізму з почерговим виконанням, interleaving concurrency). Отже, модель передбачає і описує всі можливі траси виконання програми *Program* та системи (авторизації виплати переказів), що моделюється.

Доведення коректності

Таким чином, необхідно довести властивість $A \le 1$, якщо на початку роботи $A \le 1$, тобто $\{A \le 1\}$ *Program* $\{A \le 1\}$.

Якщо

$$PreCond(S) = (A \Rightarrow (d) \le 1),\ PostCond(S) = (A \Rightarrow (d) \le 1),$$

де

$$S = (s_1, s_2, s_3, s_4, s_5, s_6, s_7, s_8, s_9, s_{10}, s_{11}, s_{12}, s_{13}, s_{14}, s_{15}, s_{16}, d),$$

то необхідно показати, що

$$\forall S \in SStartStates \bullet \forall S' \in SStopStates \bullet (PreCond(S) \ \&\ (S, S') \in SStep \rightarrow PostCond(S)).$$

Розглянемо

$$Inv(S) \equiv (s_3+s_4+s_5+s_6 \le 1)\ \&\ (s_2+s_3+s_4+s_5+s_6+s_7+s_8+s_9+s_{10} = P \Rightarrow (d))\ \&\ (\ (A \Rightarrow (d) = 0) \vee (\ (A \Rightarrow (d) = 1)\ \&\ (s_5 = 0)\ \&\ (s_6 = 0)\ )\ ),$$

де

$$S = (s_1, s_2, s_3, s_4, s_5, s_6, s_7, s_8, s_9, s_{10}, s_{11}, s_{12}, s_{13}, s_{14}, s_{15}, s_{16}, d).$$

Цей інваріант, насправді, досить легко утворити, розуміючи логіку програми та описані вище критичні місця. Так, він вказує, що кількість паралельних (конкуруючих) процесів, які намагаються безпосередньо авторизувати даний переказ, не може бути більше одного ($s_3+s_4+s_5+s_6 \le 1$) – умова ($P=1$) відсікає інші можливості, при цьому значення A може бути лише 0 або 1. Якщо ж значення є вже 1 (переказ авторизовано і виплачено), то жоден процес не потрапить в "зону безпосередньої авторизації" ($(A \Rightarrow (d) = 1)\ \&\ (s_5 = 0)\ \&\ (s_6 = 0)$). Ще додається одна умова "цілісності" – оператори виконуються у відповідній послідовності, "фантомні" (нові по ходу виконання) програми (підпрограми авторизації) не виникають, а кількість програм в точках перевірки (s_2, s_3, s_4, s_5, s_6, s_7, s_8, s_9, s_{10}) дорівнює кількості паралельних спроб авторизації даного переказу (P). Більш того, в інваріанті не фігурує змінна B, тобто блокування несуттєво впливають на хід авторизації (з точки зору інтерференції та паралелізму), що зрозуміло з логіки програми. (Отже, оператори, пов'язані з блокуванням, можна було б взагалі опустити з моделі та аналізу.)

Згідно методики доведемо, що

$$InvCond(Inv, PreCond, PostCond) = \text{True},$$

де

$$InvCond(Inv, PreCond, PostCond) = \forall S \in SStartStates \bullet (\ PreCond(S) \rightarrow Inv(S)\ )\ \&$$

$$\forall S \in SStopStates \bullet (\ Inv(S) \rightarrow PostCond(S)\ )\ \&\ \forall (S, S') \in SStep \bullet (\ Inv(S) \rightarrow Inv(S')\ ),$$

а також перевіримо, що $\forall S \in SStartStates \bullet PreCond(S)$. Тоді будемо мати $\forall S \in SStopStates \bullet PostCond(S)$.

Оскільки для кожного

$$S \in SStartStates, \text{ де } S = (s_1, s_2, s_3, s_4, s_5, s_6, s_7, s_8, s_9, s_{10}, s_{11}, s_{12}, s_{13}, s_{14}, s_{15}, s_{16}, d),$$

маємо

$$s_2 = s_3 = s_4 = s_5 = s_6 = s_7 = s_8 = s_9 = s_{10} = 0,\ P \Rightarrow (d) = 0 \text{ та } A \Rightarrow (d) = 0 \vee A \Rightarrow (d) = 1,$$

що випливає з $(A \Rightarrow (d) \le 1)$, тобто $PreCond(S)$, то $\forall S \in SStartStates \bullet (\ PreCond(S) \rightarrow Inv(S)\ )$.

Покажемо, що

$$\forall S \in SStates \bullet (\ Inv(S) \rightarrow PostCond(S)\ ).$$

Очевидно,

$$\{\ (A \Rightarrow (d) = 0) \rightarrow A \Rightarrow (d) \le 1,\ (A \Rightarrow (d) = 1) \rightarrow A \Rightarrow (d) \le 1 \Rightarrow (\ (A \Rightarrow (d) = 1)\ \&\ (s_5 = 0)\ \&\ (s_6 = 0)\ ) \rightarrow A \Rightarrow (d) \le 1\ \} \Rightarrow$$

$$(\ (A \Rightarrow (d) = 0) \vee (\ (A \Rightarrow (d) = 1)\ \&\ (s_5 = 0)\ \&\ (s_6 = 0)\ )\ ) \rightarrow A \Rightarrow (d) \le 1 \Rightarrow (s_3+s_4+s_5+s_6 \le 1)\ \&$$

$$(s_2+s_3+s_4+s_5+s_6+s_7+s_8+s_9+s_{10} = P \Rightarrow (d))\ \&\ (\ (A \Rightarrow (d) = 0) \vee (\ (A \Rightarrow (d) = 1)\ \&\ (s_5 = 0)\ \&\ (s_6 = 0)\ )\ )$$

$$\rightarrow A \Rightarrow (d) \le 1 = Inv(S) \rightarrow PostCond(S).$$

Отже, зокрема,

$$\forall S \in SStopStates \bullet (\ Inv(S) \rightarrow PostCond(S)\ ).$$

Залишається довести, що

$$\forall (S, S') \in SStep \bullet (\ Inv(S) \rightarrow Inv(S')\ ).$$

Розглянемо всі пари $(S, S') \in SStep$ і покажемо, що для кожної такої пари спрощених станів якщо $Inv(S)$, то і $Inv(S')$. Дослідимо множини пар станів $(S, S') \in SStep$ "по частинах" (множини "однотипних" перетворень).

1. Розглянемо $(S, S') \in SStep$:

$$\{ ((s_1, s_2, s_3, s_4, s_5, s_6, s_7, s_8, s_9, s_{10}, s_{11}, s_{12}, s_{13}, s_{14}, s_{15}, s_{16}, d), (s_1-1, s_2+1, s_3, s_4, s_5, s_6, s_7, s_8, s_9, s_{10}, s_{11}, s_{12}, s_{13}, s_{14}, s_{15}, s_{16}, sem_{(T, P) := (T+1, P+1)}(d))) \mid s_1>0 \ \& \ \forall i \in N_{16} \bullet s_i \in N \}$$

якщо

$$Inv(S) = (s_3+s_4+s_5+s_6 \le 1) \ \& \ (s_2+s_3+s_4+s_5+s_6+s_7+s_8+s_9+s_{10}=P\Rightarrow(d)) \ \& \ ((A\Rightarrow(d)=0) \vee ((A\Rightarrow(d)=1) \ \& \ (s_5=0) \ \& \ (s_6=0))) = \text{True},$$

де

$$S=(s_1, s_2, s_3, s_4, s_5, s_6, s_7, s_8, s_9, s_{10}, s_{11}, s_{12}, s_{13}, s_{14}, s_{15}, s_{16}, d),$$

то

$$Inv(S') = (s_3+s_4+s_5+s_6 \le 1) \ \& \ ((s_2+1)+s_3+s_4+s_5+s_6+s_7+s_8+s_9+s_{10}=P\Rightarrow(d'))$$

$$\& \ ((A\Rightarrow(d')=0) \vee ((A\Rightarrow(d')=1) \ \& \ (s_5=0) \ \& \ (s_6=0))),$$

де

$$d' = sem_{(T, P) := (T+1, P+1)}(d) = d \nabla [T \mapsto (T\Rightarrow(d) + 1), P \mapsto (P\Rightarrow(d) + 1)].$$

Таким чином,

$$P\Rightarrow(d') = P\Rightarrow(d) + 1, \text{ а } A\Rightarrow(d') = A\Rightarrow(d).$$

Звідси,

$$Inv(S') = (s_3+s_4+s_5+s_6 \le 1) \ \& \ ((s_2+1)+s_3+s_4+s_5+s_6+s_7+s_8+s_9+s_{10}=P\Rightarrow(d)+1) \ \& \ ((A\Rightarrow(d)=0) \vee$$

$$((A\Rightarrow(d)=1) \ \& \ (s_5=0) \ \& \ (s_6=0))) = Inv(S) = \text{True}.$$

2. Розглянемо $(S, S') \in SStep$:

$$\{ ((s_1, s_2, s_3, s_4, s_5, s_6, s_7, s_8, s_9, s_{10}, s_{11}, s_{12}, s_{13}, s_{14}, s_{15}, s_{16}, d), (s_1, s_2-1, s_3+1, s_4, s_5, s_6, s_7, s_8, s_9, s_{10}, s_{11}, s_{12}, s_{13}, s_{14}, s_{15}, s_{16}, d)) \mid s_2>0 \ \& \ \forall i \in N_{16} \bullet s_i \in N \ \& \ sem_{P=1}(d)=\text{True} \}$$

якщо

$$Inv(S) = (s_3+s_4+s_5+s_6 \le 1) \ \& \ (s_2+s_3+s_4+s_5+s_6+s_7+s_8+s_9+s_{10}=P\Rightarrow(d)) \ \&$$

$$((A\Rightarrow(d)=0) \vee ((A\Rightarrow(d)=1) \ \& \ (s_5=0) \ \& \ (s_6=0))) = \text{True},$$

де

$$S=(s_1, s_2, s_3, s_4, s_5, s_6, s_7, s_8, s_9, s_{10}, s_{11}, s_{12}, s_{13}, s_{14}, s_{15}, s_{16}, d),$$

а також $sem_{P=1}(d) = (P\Rightarrow(d) = 1) = \text{True}$, тобто $P\Rightarrow(d) = 1$,
то з

$$(s_2+s_3+s_4+s_5+s_6+s_7+s_8+s_9+s_{10}=P\Rightarrow(d)), \ P\Rightarrow(d) = 1 \text{ та } s_2>0$$

випливає, що

$$s_2=1 \text{ та } s_3=s_4=s_5=s_6=s_7=s_8=s_9=s_{10}=0,$$

тоді, враховуючи

$$((A\Rightarrow(d)=0) \vee ((A\Rightarrow(d)=1) \ \& \ (s_5=0) \ \& \ (s_6=0))) = \text{True}, \ Inv(S') = ((s_3+1)+s_4+s_5+s_6 \le 1) \ \&$$

$$((s_2-1)+(s_3+1)+s_4+s_5+s_6+s_7+s_8+s_9+s_{10}=P\Rightarrow(d)) \ \& \ ((A\Rightarrow(d)=0) \vee ((A\Rightarrow(d)=1) \ \& \ (s_5=0) \ \& \ (s_6=0))) = \text{True}.$$

3. Розглянемо $(S, S') \in SStep$:

$$\{ ((s_1, s_2, s_3, s_4, s_5, s_6, s_7, s_8, s_9, s_{10}, s_{11}, s_{12}, s_{13}, s_{14}, s_{15}, s_{16}, d), (s_1, s_2-1, s_3, s_4, s_5, s_6, s_7, s_8, s_9, s_{10}+1, s_{11}, s_{12}, s_{13}, s_{14}, s_{15}, s_{16}, d)) \mid s_2>0 \ \& \ \forall i \in N_{16} \bullet s_i \in N \ \& \ sem_{P=1}(d)=\text{False} \} \cup$$

якщо

$$Inv(S) = (s_3+s_4+s_5+s_6 \le 1) \ \& \ (s_2+s_3+s_4+s_5+s_6+s_7+s_8+s_9+s_{10}=P\Rightarrow(d)) \ \& \ ((A\Rightarrow(d)=0) \vee$$

$$((A\Rightarrow(d)=1) \ \& \ (s_5=0) \ \& \ (s_6=0))) = \text{True},$$

де

$$S=(s_1, s_2, s_3, s_4, s_5, s_6, s_7, s_8, s_9, s_{10}, s_{11}, s_{12}, s_{13}, s_{14}, s_{15}, s_{16}, d),$$

а також $sem_{P=1}(d) = (\text{P}\Rightarrow(d) = 1) = \text{False}$,
то

$$Inv(S') = (s_3+s_4+s_5+s_6 \le 1) \ \& \ ((s_2-1)+s_3+s_4+s_5+s_6+s_7+s_8+s_9+(s_{10}+1)=P\Rightarrow(d)) \ \&$$

$$((A\Rightarrow(d)=0) \vee ((A\Rightarrow(d)=1) \ \& \ (s_5=0) \ \& \ (s_6=0))) = Inv(S) = \text{True}.$$

4. Розглянемо $(S, S') \in SStep$:

$$\{ ((s_1, s_2, s_3, s_4, s_5, s_6, s_7, s_8, s_9, s_{10}, s_{11}, s_{12}, s_{13}, s_{14}, s_{15}, s_{16}, d), (s_1, s_2, s_3-1, s_4+1, s_5, s_6, s_7, s_8, s_9, s_{10}, s_{11}, s_{12}, s_{13}, s_{14}, s_{15}, s_{16}, d)) \mid s_3>0 \ \& \ \forall i \in N_{16} \bullet s_i \in N \ \& \ sem_{B=0}(d)=\text{True} \}$$

якщо

$$Inv(S) = (s_3+s_4+s_5+s_6 \le 1) \ \& \ (s_2+s_3+s_4+s_5+s_6+s_7+s_8+s_9+s_{10}=P \Rightarrow (d)) \ \& \ ((A \Rightarrow (d)=0) \vee ((A \Rightarrow (d)=1) \ \& \ (s_5=0) \ \& \ (s_6=0))) = \text{True}, \text{ де } S=(s_1, s_2, s_3, s_4, s_5, s_6, s_7, s_8, s_9, s_{10}, s_{11}, s_{12}, s_{13}, s_{14}, s_{15}, s_{16}, d),$$

а також $sem_{B=0}(d) = (B \Rightarrow (d) = 0) = \text{True}$,
то

$$Inv(S') = ((s_3-1)+(s_4+1)+s_5+s_6 \le 1) \ \& \ (s_2+(s_3-1)+(s_4+1)+s_5+s_6+s_7+s_8+s_9+s_{10}=P \Rightarrow (d)) \ \& \ ((A \Rightarrow (d)=0) \vee ((A \Rightarrow (d)=1) \ \& \ (s_5=0) \ \& \ (s_6=0))) = Inv(S) = \text{True}.$$

5. Розглянемо $(S, S') \in SStep$:

$$\{ ((s_1, s_2, s_3, s_4, s_5, s_6, s_7, s_8, s_9, s_{10}, s_{11}, s_{12}, s_{13}, s_{14}, s_{15}, s_{16}, d), (s_1, s_2, s_3-1, s_4, s_5, s_6, s_7, s_8, s_9+1, s_{10}, s_{11}, s_{12}, s_{13}, s_{14}, s_{15}, s_{16}, d)) \mid s_3>0 \ \& \ \forall i \in N_{16} \bullet s_i \in N \ \& \ sem_{B=0}(d)=\text{False} \}$$

якщо

$$Inv(S) = (s_3+s_4+s_5+s_6 \le 1) \ \& \ (s_2+s_3+s_4+s_5+s_6+s_7+s_8+s_9+s_{10}=P \Rightarrow (d)) \ \& \ ((A \Rightarrow (d)=0) \vee ((A \Rightarrow (d)=1) \ \& \ (s_5=0) \ \& \ (s_6=0))) = \text{True},$$

де

$$S=(s_1, s_2, s_3, s_4, s_5, s_6, s_7, s_8, s_9, s_{10}, s_{11}, s_{12}, s_{13}, s_{14}, s_{15}, s_{16}, d),$$

а також $sem_{B=0}(d) = (B \Rightarrow (d) = 0) = \text{False}$,
то легко побачити, що

$$Inv(S') = ((s_3-1)+s_4+s_5+s_6 \le 1) \ \& \ (s_2+(s_3-1)+s_4+s_5+s_6+s_7+s_8+(s_9+1)+s_{10}=P \Rightarrow (d)) \ \& \ ((A \Rightarrow (d)=0) \vee ((A \Rightarrow (d)=1) \ \& \ (s_5=0) \ \& \ (s_6=0))) = \text{True},$$

адже з $Inv(S)$ логічно випливає $Inv(S')$.

6. Розглянемо $(S, S') \in SStep$:

$$\{ ((s_1, s_2, s_3, s_4, s_5, s_6, s_7, s_8, s_9, s_{10}, s_{11}, s_{12}, s_{13}, s_{14}, s_{15}, s_{16}, d), (s_1, s_2, s_3, s_4-1, s_5+1, s_6, s_7, s_8, s_9, s_{10}, s_{11}, s_{12}, s_{13}, s_{14}, s_{15}, s_{16}, d)) \mid s_4>0 \ \& \ \forall i \in N_{16} \bullet s_i \in N \ \& \ sem_{A=0}(d)=\text{True} \}$$

якщо

$$Inv(S) = (s_3+s_4+s_5+s_6 \le 1) \ \& \ (s_2+s_3+s_4+s_5+s_6+s_7+s_8+s_9+s_{10}=P \Rightarrow (d)) \ \& \ ((A \Rightarrow (d)=0) \vee ((A \Rightarrow (d)=1) \ \& \ (s_5=0) \ \& \ (s_6=0))) = \text{True},$$

де

$$S=(s_1, s_2, s_3, s_4, s_5, s_6, s_7, s_8, s_9, s_{10}, s_{11}, s_{12}, s_{13}, s_{14}, s_{15}, s_{16}, d),$$

а також $sem_{A=0}(d) = (A \Rightarrow (d) = 0) = \text{True}$, тобто $A \Rightarrow (d) = 0$, то

$$Inv(S') = (s_3+(s_4-1)+(s_5+1)+s_6 \le 1) \ \& \ (s_2+s_3+(s_4-1)+(s_5+1)+s_6+s_7+s_8+s_9+s_{10}=P \Rightarrow (d)) \ \& \ ((A \Rightarrow (d)=0) \vee ((A \Rightarrow (d)=1) \ \& \ ((s_5+1)=0) \ \& \ (s_6=0))) = \text{True},$$

адже перші два кон'юнкти – істинні з передумови ($Inv(S) = \text{True}$), а третій – оскільки $A \Rightarrow (d) = 0$.

7. Розглянемо $(S, S') \in SStep$:

$$\{ ((s_1, s_2, s_3, s_4, s_5, s_6, s_7, s_8, s_9, s_{10}, s_{11}, s_{12}, s_{13}, s_{14}, s_{15}, s_{16}, d), (s_1, s_2, s_3, s_4-1, s_5, s_6, s_7, s_8+1, s_9, s_{10}, s_{11}, s_{12}, s_{13}, s_{14}, s_{15}, s_{16}, d)) \mid s_4>0 \ \& \ \forall i \in N_{16} \bullet s_i \in N \ \& \ sem_{A=0}(d)=\text{False} \}$$

якщо

$$Inv(S) = (s_3+s_4+s_5+s_6 \le 1) \ \& \ (s_2+s_3+s_4+s_5+s_6+s_7+s_8+s_9+s_{10}=P \Rightarrow (d)) \ \& \ ((A \Rightarrow (d)=0) \vee ((A \Rightarrow (d)=1) \ \& \ (s_5=0) \ \& \ (s_6=0))) = \text{True},$$

де

$$S=(s_1, s_2, s_3, s_4, s_5, s_6, s_7, s_8, s_9, s_{10}, s_{11}, s_{12}, s_{13}, s_{14}, s_{15}, s_{16}, d),$$

а також $sem_{A=0}(d) = (A \Rightarrow (d) = 0) = \text{False}$,
то

$$Inv(S') = (s_3+(s_4-1)+s_5+s_6\le 1)\ \&\ (s_2+s_3+(s_4-1)+s_5+s_6+s_7+(s_8+1)+s_9+s_{10}=P\Rightarrow(d))\ \&$$

$$(\ (A\Rightarrow(d)=0)\ \vee\ (\ (A\Rightarrow(d)=1)\ \&\ (s_5=0)\ \&\ (s_6=0)\ )\ ) = \text{True}\ (\text{з } Inv(S)$$

логічно випливає $Inv(S')$).

8. Розглянемо $(S, S')\in SStep$:

$$\{\ (\ (s_1, s_2, s_3, s_4, s_5, s_6, s_7, s_8, s_9, s_{10}, s_{11}, s_{12}, s_{13}, s_{14}, s_{15}, s_{16}, d),\ (s_1, s_2, s_3, s_4, s_5-1, s_6+1, s_7, s_8, s_9, s_{10}, s_{11}, s_{12},$$

$$s_{13}, s_{14}, s_{15}, s_{16}, d)\ )\ |\ s_5>0\ \&\ \forall i\in N_{16}\bullet s_i\in N\ \&\ sem_{\ data\_provided\_correct()}\ (d)=\text{True}\ \}$$

якщо

$$Inv(S) = (s_3+s_4+s_5+s_6\le 1)\ \&\ (s_2+s_3+s_4+s_5+s_6+s_7+s_8+s_9+s_{10}=P\Rightarrow(d))\ \&\ (\ (A\Rightarrow(d)=0)\ \vee\ (\ (A\Rightarrow(d)=1)\ \&$$

$$(s_5=0)\ \&\ (s_6=0)\ )\ ) = \text{True},$$

де

$$S=(s_1, s_2, s_3, s_4, s_5, s_6, s_7, s_8, s_9, s_{10}, s_{11}, s_{12}, s_{13}, s_{14}, s_{15}, s_{16}, d),$$

а також $sem_{\ data\_provided\_correct()}\ (d) = (choice\ (\text{True, False})) = \text{True}$,
звідси, враховуючи $s_5>0$, маємо $s_5=1$ та $s_3=s_4=s_6=0$ (випливає з істинності першого кон'юнкта $Inv(S)$), звідси

$$(\ (A\Rightarrow(d)=1)\ \&\ (s_5=0)\ \&\ (s_6=0)\ ) = \text{False, тому } (A\Rightarrow(d)=0) = \text{True},$$

щоб

$$(\ (A\Rightarrow(d)=0)\ \vee\ (\ (A\Rightarrow(d)=1)\ \&\ (s_5=0)\ \&(s_6=0)\ )\ ) = \text{True},$$

тоді, очевидно,

$$Inv(S') = (s_3+s_4+(s_5-1)+(s_6+1)\le 1)\ \&\ (s_2+s_3+s_4+(s_5-1)+(s_6+1)+s_7+s_8+s_9+s_{10}=P\Rightarrow(d))\ \&$$

$$(\ (A\Rightarrow(d)=0)\ \vee\ (\ (A\Rightarrow(d)=1)\ \&\ ((s_5-1)=0)\ \&\ ((s_6+1)=0)\ )\ ) = \text{True}.$$

9. Розглянемо $(S, S')\in SStep$:

$$\{\ (\ (s_1, s_2, s_3, s_4, s_5, s_6, s_7, s_8, s_9, s_{10}, s_{11}, s_{12}, s_{13}, s_{14}, s_{15}, s_{16}, d),\ (s_1, s_2, s_3, s_4, s_5-1, s_6, s_7+1, s_8, s_9, s_{10}, s_{11}, s_{12},$$

$$s_{13}, s_{14}, s_{15}, s_{16}, d)\ )\ |\ s_5>0\ \&\ \forall i\in N_{16}\bullet s_i\in N\ \&\ sem_{\ data\_provided\_correct()}\ (d)=\text{False}\ \}$$

якщо

$$Inv(S) = (s_3+s_4+s_5+s_6\le 1)\ \&\ (s_2+s_3+s_4+s_5+s_6+s_7+s_8+s_9+s_{10}=P\Rightarrow(d))\ \&\ (\ (A\Rightarrow(d)=0)\ \vee\ (\ (A\Rightarrow(d)=1)\ \&$$

$$(s_5=0)\ \&\ (s_6=0)\ )\ ) = \text{True},$$

де

$$S=(s_1, s_2, s_3, s_4, s_5, s_6, s_7, s_8, s_9, s_{10}, s_{11}, s_{12}, s_{13}, s_{14}, s_{15}, s_{16}, d),$$

а також

$$sem_{\ data\_provided\_correct()}\ (d) = (choice\ (\text{True, False})) = \text{False},$$

звідси, враховуючи $s_5>0$, маємо $s_5=1$ та $s_3=s_4=s_6=0$ (випливає з істинності першого кон'юнкта $Inv(S)$), звідси

$$(\ (A\Rightarrow(d)=1)\ \&\ (s_5=0)\ \&\ (s_6=0)\ ) = \text{False, тому } (A\Rightarrow(d)=0) = \text{True},$$

щоб третій кон'юнкт $(\ (A\Rightarrow(d)=0)\ \vee\ (\ (A\Rightarrow(d)=1)\ \&\ (s_5=0)\ \&\ (s_6=0)\ )\ ) = \text{True}$,
тоді

$$Inv(S') = (s_3+s_4+(s_5-1)+s_6\le 1)\ \&\ (s_2+s_3+s_4+(s_5-1)+s_6+(s_7+1)+s_8+s_9+s_{10}=P\Rightarrow(d))\ \&$$

$$(\ (A\Rightarrow(d)=0)\ \vee\ (\ (A\Rightarrow(d)=1)\ \&\ ((s_5-1)=0)\ \&\ (s_6=0)\ )\ ) = \text{True}$$

(перший і другий кон'юнкти випливають з $Inv(S)$, а третій є істинним за рахунок $(A\Rightarrow(d)=0) = \text{True}$).

10. Розглянемо $(S, S')\in SStep$:

$$\{\ (\ (s_1, s_2, s_3, s_4, s_5, s_6, s_7, s_8, s_9, s_{10}, s_{11}, s_{12}, s_{13}, s_{14}, s_{15}, s_{16}, d),\ (s_1, s_2, s_3, s_4, s_5, s_6-1, s_7, s_8, s_9, s_{10}, s_{11}+1, s_{12},$$

$$s_{13}, s_{14}, s_{15}, s_{16}, sem_{\ (A,\ P)\ :=\ (A+1,\ P-1)}\ (d))\ )\ |\ s_6>0\ \&\ \forall i\in N_{16}\bullet s_i\in N\ \}$$

якщо

$$Inv(S) = (s_3+s_4+s_5+s_6\le 1)\ \&\ (s_2+s_3+s_4+s_5+s_6+s_7+s_8+s_9+s_{10}=P\Rightarrow(d))\ \&\ (\ (A\Rightarrow(d)=0)\ \vee$$

$$(\ (A\Rightarrow(d)=1)\ \&\ (s_5=0)\ \&\ (s_6=0)\ )\ ) = \text{True},$$

де

$$S=(s_1, s_2, s_3, s_4, s_5, s_6, s_7, s_8, s_9, s_{10}, s_{11}, s_{12}, s_{13}, s_{14}, s_{15}, s_{16}, d), \text{ а також } s_6>0,$$

то

$$Inv(S') = (s_3+s_4+s_5+(s_6-1)\le 1)\ \&\ (s_2+s_3+s_4+s_5+(s_6-1)+s_7+s_8+s_9+s_{10}=P\Rightarrow(d'))\ \&\ (\ (A\Rightarrow(d')=0)\ \vee$$

$$(\ (A\Rightarrow(d')=1)\ \&\ (s_5=0)\ \&\ ((s_6-1)=0)\ )\ ),$$

де

$$d' = sem_{(A,\,P) := (A+1,\,P-1)}(d) = d \nabla [\, A \mapsto (A \Rightarrow (d) + 1),\ P \mapsto (P \Rightarrow (d) - 1)\,].$$

Таким чином, $P \Rightarrow (d') = P \Rightarrow (d) - 1$, а $A \Rightarrow (d') = A \Rightarrow (d) + 1$.

Тепер, з $Inv(S)$ = True випливає, що

$$s_3+s_4+s_5+s_6 \le 1,\ s_2+s_3+s_4+s_5+s_6+s_7+s_8+s_9+s_{10} = P \Rightarrow (d) \text{ та } (A \Rightarrow (d)=0) \vee (\,(A \Rightarrow (d)=1) \,\&\, (s_5=0) \,\&\, (s_6=0)\,) = \text{True}.$$

Оскільки $s_6>0$, то

$$\text{True} = (\,(A \Rightarrow (d)=0) \vee (\,(A \Rightarrow (d)=1) \,\&\, (s_5=0) \,\&\, (s_6=0)\,)\,) \,\&\, (s_6>0) = (A \Rightarrow (d)=0) \,\&\, (s_6>0) \vee (A \Rightarrow (d)=1) \,\&$$

$$(s_5=0) \,\&\, (s_6=0) \,\&\, (s_6>0) = (A \Rightarrow (d)=0) \,\&\, (s_6>0) \vee \text{False} = (A \Rightarrow (d)=0) \,\&\, \text{True} = (A \Rightarrow (d)=0).$$

Далі, за визначенням спрощеного стану, перші 16 компонент є невід'ємними цілими числами, а отже, з $s_3+s_4+s_5+s_6 \le 1$ та $s_6>0$ випливає, що $s_6=1$ та $s_3=s_4=s_5=0$.

Звідси,

$$Inv(S') = (s_3+s_4+s_5+(s_6-1) \le 1) \,\&\, (s_2+s_3+s_4+s_5+(s_6-1)+s_7+s_8+s_9+s_{10} = P \Rightarrow (d)-1) \,\&$$

$$(\,(A \Rightarrow (d)+1=0) \vee (\,(A \Rightarrow (d)+1=1) \,\&\, (s_5=0) \,\&\, ((s_6-1)=0)\,)\,).$$

Перші два кон'юнкти мають значення True (очевидно, випливає з $Inv(S)$ = True). Далі, з $A \Rightarrow (d)=0$, $s_5=0$ та $s_6=1$ випливає, що

$$(A \Rightarrow (d)+1=1) \,\&\, (s_5=0) \,\&\, ((s_6-1)=0) = \text{True},$$

тобто і третій кон'юнкт має значення True. Таким чином, $Inv(S')$ = True.

11. Розглянемо $(S, S') \in SStep$:

$$\{\,(\,(s_1, s_2, s_3, s_4, s_5, s_6, s_7, s_8, s_9, s_{10}, s_{11}, s_{12}, s_{13}, s_{14}, s_{15}, s_{16}, d),\ (s_1, s_2, s_3, s_4, s_5, s_6, s_7-1, s_8, s_9, s_{10}, s_{11}+1, s_{12},$$

$$s_{13}, s_{14}, s_{15}, s_{16}, sem_{P := P-1}(d))\,) \mid s_7>0 \,\&\, \forall i \in N_{16} \bullet s_i \in N\,\}$$

якщо

$$Inv(S) = (s_3+s_4+s_5+s_6 \le 1) \,\&\, (s_2+s_3+s_4+s_5+s_6+s_7+s_8+s_9+s_{10} = P \Rightarrow (d)) \,\&\, (\,(A \Rightarrow (d)=0) \vee$$

$$(\,(A \Rightarrow (d)=1) \,\&\, (s_5=0) \,\&\, (s_6=0)\,)\,) = \text{True},$$

де

$$S = (s_1, s_2, s_3, s_4, s_5, s_6, s_7, s_8, s_9, s_{10}, s_{11}, s_{12}, s_{13}, s_{14}, s_{15}, s_{16}, d),$$

то

$$Inv(S') = (s_3+s_4+s_5+s_6 \le 1) \,\&\, (s_2+s_3+s_4+s_5+s_6+(s_7-1)+s_8+s_9+s_{10} = P \Rightarrow (d')) \,\&$$

$$(\,(A \Rightarrow (d')=0) \vee (\,(A \Rightarrow (d')=1) \,\&\, (s_5=0) \,\&\, (s_6=0)\,)\,),$$

де $d' = sem_{P := P-1}(d) = d \nabla [\, P \mapsto (P \Rightarrow (d) - 1)\,]$.

Таким чином, $P \Rightarrow (d') = P \Rightarrow (d) - 1$, а $A \Rightarrow (d') = A \Rightarrow (d)$.

Звідси,

$$Inv(S') = (s_3+s_4+s_5+s_6 \le 1) \,\&\, (s_2+s_3+s_4+s_5+s_6+(s_7-1)+s_8+s_9+s_{10} = P \Rightarrow (d)-1) \,\&\, (\,(A \Rightarrow (d)=0) \vee$$

$$(\,(A \Rightarrow (d)=1) \,\&\, (s_5=0) \,\&\, (s_6=0)\,)\,) = Inv(S) = \text{True}.$$

12. Розглянемо $(S, S') \in SStep$:

$$\{\,(\,(s_1, s_2, s_3, s_4, s_5, s_6, s_7, s_8, s_9, s_{10}, s_{11}, s_{12}, s_{13}, s_{14}, s_{15}, s_{16}, d),\ (s_1, s_2, s_3, s_4, s_5, s_6, s_7, s_8-1, s_9, s_{10}, s_{11}+1, s_{12},$$

$$s_{13}, s_{14}, s_{15}, s_{16}, sem_{P := P-1}(d))\,) \mid s_8>0 \,\&\, \forall i \in N_{16} \bullet s_i \in N\,\}$$

якщо

$$Inv(S) = (s_3+s_4+s_5+s_6 \le 1) \,\&\, (s_2+s_3+s_4+s_5+s_6+s_7+s_8+s_9+s_{10} = P \Rightarrow (d)) \,\&\, (\,(A \Rightarrow (d)=0) \vee (\,(A \Rightarrow (d)=1) \,\&\, (s_5=0) \,\&\,$$

$$(s_6=0)\,)\,) = \text{True}, \text{ де } S = (s_1, s_2, s_3, s_4, s_5, s_6, s_7, s_8, s_9, s_{10}, s_{11}, s_{12}, s_{13}, s_{14}, s_{15}, s_{16}, d),$$

то

$$Inv(S') = (s_3+s_4+s_5+s_6 \le 1) \,\&\, (s_2+s_3+s_4+s_5+s_6+s_7+(s_8-1)+s_9+s_{10} = P \Rightarrow (d')) \,\&\, (\,(A \Rightarrow (d')=0) \vee$$

$$(\,(A \Rightarrow (d')=1) \,\&\, (s_5=0) \,\&\, (s_6=0)\,)\,),$$

де $d' = sem_{P := P-1}(d) = d \nabla [\, P \mapsto (P \Rightarrow (d) - 1)\,]$.

Таким чином, $P \Rightarrow (d') = P \Rightarrow (d) - 1$, а $A \Rightarrow (d') = A \Rightarrow (d)$.

Звідси,

$$Inv(S') = (s_3+s_4+s_5+s_6 \le 1) \,\&\, (s_2+s_3+s_4+s_5+s_6+s_7+(s_8-1)+s_9+s_{10} = P \Rightarrow (d)-1) \,\&$$

$$(\,(A \Rightarrow (d)=0) \vee (\,(A \Rightarrow (d)=1) \,\&\, (s_5=0) \,\&\, (s_6=0)\,)\,) = Inv(S) = \text{True}.$$

13. Розглянемо $(S, S') \in SStep$:

$$\{ ((s_1, s_2, s_3, s_4, s_5, s_6, s_7, s_8, s_9, s_{10}, s_{11}, s_{12}, s_{13}, s_{14}, s_{15}, s_{16}, d), (s_1, s_2, s_3, s_4, s_5, s_6, s_7, s_8, s_9-1, s_{10}, s_{11}+1, s_{12}, s_{13}, s_{14}, s_{15}, s_{16}, sem_{P:=P-1}(d))) \mid s_9>0 \ \& \ \forall i \in N_{16} \bullet s_i \in N \}$$

якщо

$$Inv(S) = (s_3+s_4+s_5+s_6 \le 1) \ \& \ (s_2+s_3+s_4+s_5+s_6+s_7+s_8+s_9+s_{10}=P \Rightarrow (d)) \ \& \ ((A \Rightarrow (d)=0) \vee ((A \Rightarrow (d)=1) \ \& \ (s_5=0) \ \& \ (s_6=0))) = \text{True},$$

де

$$S=(s_1, s_2, s_3, s_4, s_5, s_6, s_7, s_8, s_9, s_{10}, s_{11}, s_{12}, s_{13}, s_{14}, s_{15}, s_{16}, d),$$

то

$$Inv(S') = (s_3+s_4+s_5+s_6 \le 1) \ \& \ (s_2+s_3+s_4+s_5+s_6+s_7+s_8+(s_9-1)+s_{10}=P \Rightarrow (d')) \ \& \ ((A \Rightarrow (d')=0) \vee ((A \Rightarrow (d')=1) \ \& \ (s_5=0) \ \& \ (s_6=0))),$$

де $d' = sem_{P:=P-1}(d) = d \nabla [P \mapsto (P \Rightarrow (d) - 1)]$.

Таким чином, $P \Rightarrow (d') = P \Rightarrow (d) - 1$, а $A \Rightarrow (d') = A \Rightarrow (d)$.

Звідси,

$$Inv(S') = (s_3+s_4+s_5+s_6 \le 1) \ \& \ (s_2+s_3+s_4+s_5+s_6+s_7+s_8+(s_9-1)+s_{10}=P \Rightarrow (d)-1) \ \& \ ((A \Rightarrow (d)=0) \vee ((A \Rightarrow (d)=1) \ \& \ (s_5=0) \ \& \ (s_6=0))) = Inv(S) = \text{True}.$$

14. Розглянемо $(S, S') \in SStep$:

$$\{ ((s_1, s_2, s_3, s_4, s_5, s_6, s_7, s_8, s_9, s_{10}, s_{11}, s_{12}, s_{13}, s_{14}, s_{15}, s_{16}, d), (s_1, s_2, s_3, s_4, s_5, s_6, s_7, s_8, s_9, s_{10}-1, s_{11}+1, s_{12}, s_{13}, s_{14}, s_{15}, s_{16}, sem_{P:=P-1}(d))) \mid s_{10}>0 \ \& \ \forall i \in N_{16} \bullet s_i \in N \}$$

якщо

$$Inv(S) = (s_3+s_4+s_5+s_6 \le 1) \ \& \ (s_2+s_3+s_4+s_5+s_6+s_7+s_8+s_9+s_{10}=P \Rightarrow (d)) \ \& \ ((A \Rightarrow (d)=0) \vee ((A \Rightarrow (d)=1) \ \& \ (s_5=0) \ \& \ (s_6=0))) = \text{True},$$

де

$$S=(s_1, s_2, s_3, s_4, s_5, s_6, s_7, s_8, s_9, s_{10}, s_{11}, s_{12}, s_{13}, s_{14}, s_{15}, s_{16}, d),$$

то

$$Inv(S') = (s_3+s_4+s_5+s_6 \le 1) \ \& \ (s_2+s_3+s_4+s_5+s_6+s_7+s_8+s_9+(s_{10}-1)=P \Rightarrow (d')) \ \& \ ((A \Rightarrow (d')=0) \vee ((A \Rightarrow (d')=1) \ \& \ (s_5=0) \ \& \ (s_6=0))),$$

де $d' = sem_{P:=P-1}(d) = d \nabla [P \mapsto (P \Rightarrow (d) - 1)]$.

Таким чином, $P \Rightarrow (d') = P \Rightarrow (d) - 1$, а $A \Rightarrow (d') = A \Rightarrow (d)$.

Звідси,

$$Inv(S') = (s_3+s_4+s_5+s_6 \le 1) \ \& \ (s_2+s_3+s_4+s_5+s_6+s_7+s_8+s_9+(s_{10}-1)=P \Rightarrow (d)-1) \ \& \ ((A \Rightarrow (d)=0) \vee ((A \Rightarrow (d)=1) \ \& \ (s_5=0) \ \& \ (s_6=0))) = Inv(S) = \text{True}.$$

15. Розглянемо $(S, S') \in SStep$:

$$\{ ((s_1, s_2, s_3, s_4, s_5, s_6, s_7, s_8, s_9, s_{10}, s_{11}, s_{12}, s_{13}, s_{14}, s_{15}, s_{16}, d), (s_1, s_2, s_3, s_4, s_5, s_6, s_7, s_8, s_9, s_{10}, s_{11}-1, s_{12}+1, s_{13}, s_{14}, s_{15}, s_{16}, d)) \mid s_{11}>0 \ \& \ \forall i \in N_{16} \bullet s_i \in N \ \& \ sem_{T>2}(d)=\text{True} \}$$

якщо

$$Inv(S) = (s_3+s_4+s_5+s_6 \le 1) \ \& \ (s_2+s_3+s_4+s_5+s_6+s_7+s_8+s_9+s_{10}=P \Rightarrow (d)) \ \& \ ((A \Rightarrow (d)=0) \vee ((A \Rightarrow (d)=1) \ \& \ (s_5=0) \ \& \ (s_6=0))) = \text{True},$$

де

$$S=(s_1, s_2, s_3, s_4, s_5, s_6, s_7, s_8, s_9, s_{10}, s_{11}, s_{12}, s_{13}, s_{14}, s_{15}, s_{16}, d),$$

а також $sem_{T>2}(d) = (T \Rightarrow (d) > 2) = \text{True}$,

то

$$Inv(S') = (s_3+s_4+s_5+s_6 \le 1) \ \& \ (s_2+s_3+s_4+s_5+s_6+s_7+s_8+s_9+s_{10}=P \Rightarrow (d)) \ \& \ ((A \Rightarrow (d)=0) \vee ((A \Rightarrow (d)=1) \ \& \ (s_5=0) \ \& \ (s_6=0))) = Inv(S) = \text{True}.$$

16. Розглянемо $(S, S') \in SStep$:

$$\{ ((s_1, s_2, s_3, s_4, s_5, s_6, s_7, s_8, s_9, s_{10}, s_{11}, s_{12}, s_{13}, s_{14}, s_{15}, s_{16}, d), (s_1, s_2, s_3, s_4, s_5, s_6, s_7, s_8, s_9, s_{10}, s_{11}-1, s_{12}, s_{13}+1, s_{14}, s_{15}, s_{16}, d)) \mid s_{11}>0 \ \& \ \forall i \in N_{16} \bullet s_i \in N \ \& \ sem_{T>2}(d)=\text{False} \}$$

якщо

$$Inv(S) = (s_3+s_4+s_5+s_6 \le 1) \ \& \ (s_2+s_3+s_4+s_5+s_6+s_7+s_8+s_9+s_{10}=P \Rightarrow (d)) \ \& \ ((A \Rightarrow (d)=0) \vee ((A \Rightarrow (d)=1) \ \& \ (s_5=0) \ \& \ (s_6=0))) = \text{True},$$

де

$$S=(s_1, s_2, s_3, s_4, s_5, s_6, s_7, s_8, s_9, s_{10}, s_{11}, s_{12}, s_{13}, s_{14}, s_{15}, s_{16}, d),$$

а також $sem_{T>2}(d) = (T\Rightarrow(d) > 2) = \text{False}$,
то

$$Inv(S') = (s_3+s_4+s_5+s_6\le 1) \,\&\, (s_2+ s_3+s_4+s_5+s_6+s_7+s_8+s_9+s_{10}=P\Rightarrow(d)) \,\&\, (\,(A\Rightarrow(d)=0) \vee$$

$$(\,(A\Rightarrow(d)=1) \,\&\, (s_5=0) \,\&\, (s_6=0)\,)\,) = Inv(S) = \text{True}.$$

17. Розглянемо $(S, S')\in SStep$:

$$\{\,(\,(s_1, s_2, s_3, s_4, s_5, s_6, s_7, s_8, s_9, s_{10}, s_{11}, s_{12}, s_{13}, s_{14}, s_{15}, s_{16}, d), (s_1, s_2, s_3, s_4, s_5, s_6, s_7, s_8, s_9, s_{10}, s_{11},$$

$$s_{12}-1, s_{13}, s_{14}+1, s_{15}, s_{16}, sem_{B:=1}(d))\,) \mid s_{12}>0 \,\&\, \forall i\in N_{16} \bullet s_i\in N\,\}$$

якщо

$$Inv(S) = (s_3+s_4+s_5+s_6\le 1) \,\&\, (s_2+s_3+s_4+s_5+s_6+s_7+s_8+s_9+s_{10}=P\Rightarrow(d)) \,\&\, (\,(A\Rightarrow(d)=0) \vee$$

$$(\,(A\Rightarrow(d)=1) \,\&\, (s_5=0) \,\&\, (s_6=0)\,)\,) = \text{True},$$

де

$$S=(s_1, s_2, s_3, s_4, s_5, s_6, s_7, s_8, s_9, s_{10}, s_{11}, s_{12}, s_{13}, s_{14}, s_{15}, s_{16}, d),$$

то

$$Inv(S') = (s_3+s_4+s_5+s_6\le 1) \,\&\, (s_2+s_3+s_4+s_5+s_6+s_7+s_8+s_9+s_{10}=P\Rightarrow(d')) \,\&\, (\,(A\Rightarrow(d')=0) \vee$$

$$(\,(A\Rightarrow(d')=1) \,\&\, (s_5=0) \,\&\, (s_6=0)\,)\,),$$

де $d' = sem_{B:=1}(d) = d\,\nabla\,[\,B\mapsto 1\,]$.

Таким чином, $P\Rightarrow(d') = P\Rightarrow(d)$, а $A\Rightarrow(d') = A\Rightarrow(d)$.

Звідси,

$$Inv(S') = (s_3+s_4+s_5+s_6\le 1) \,\&\, (s_2+s_3+s_4+s_5+s_6+s_7+s_8+s_9+s_{10}=P\Rightarrow(d)) \,\&\, (\,(A\Rightarrow(d)=0) \vee$$

$$(\,(A\Rightarrow(d)=1) \,\&\, (s_5=0) \,\&\, (s_6=0)\,)\,) = Inv(S) = \text{True}.$$

18. Розглянемо $(S, S')\in SStep$:

$$\{\,(\,(s_1, s_2, s_3, s_4, s_5, s_6, s_7, s_8, s_9, s_{10}, s_{11}, s_{12}, s_{13}, s_{14}, s_{15}, s_{16}, d), (s_1, s_2, s_3, s_4, s_5, s_6, s_7, s_8, s_9, s_{10},$$

$$s_{11}, s_{12}, s_{13}-1, s_{14}+1, s_{15}, s_{16}, sem_{Id}(d))\,) \mid s_{13}>0 \,\&\, \forall i\in N_{16} \bullet s_i\in N\,\}$$

якщо

$$Inv(S) = (s_3+s_4+s_5+s_6\le 1) \,\&\, (s_2+s_3+s_4+s_5+s_6+s_7+s_8+s_9+s_{10}=P\Rightarrow(d)) \,\&\, (\,(A\Rightarrow(d)=0) \vee$$

$$(\,(A\Rightarrow(d)=1) \,\&\, (s_5=0) \,\&\, (s_6=0)\,)\,) = \text{True},$$

де

$$S=(s_1, s_2, s_3, s_4, s_5, s_6, s_7, s_8, s_9, s_{10}, s_{11}, s_{12}, s_{13}, s_{14}, s_{15}, s_{16}, d),$$

то

$$Inv(S') = (s_3+s_4+s_5+s_6\le 1) \,\&\, (s_2+s_3+s_4+s_5+s_6+s_7+s_8+s_9+s_{10}=P\Rightarrow(d')) \,\&$$

$$(\,(A\Rightarrow(d')=0) \vee (\,(A\Rightarrow(d')=1) \,\&\, (s_5=0) \,\&\, (s_6=0)\,)\,),$$

де $d' = sem_{Id}(d) = d$.

Таким чином, $P\Rightarrow(d') = P\Rightarrow(d)$, а $A\Rightarrow(d') = A\Rightarrow(d)$.

Звідси,

$$Inv(S') = (s_3+s_4+s_5+s_6\le 1) \,\&\, (s_2+s_3+s_4+s_5+s_6+s_7+s_8+s_9+s_{10}=P\Rightarrow(d)) \,\&\, (\,(A\Rightarrow(d)=0) \vee$$

$$(\,(A\Rightarrow(d)=1) \,\&\, (s_5=0) \,\&\, (s_6=0)\,)\,) = Inv(S) = \text{True}.$$

19. Розглянемо $(S, S')\in SStep$:

$$\{\,(\,(s_1, s_2, s_3, s_4, s_5, s_6, s_7, s_8, s_9, s_{10}, s_{11}, s_{12}, s_{13}, s_{14}, s_{15}, s_{16}, d), (s_1, s_2, s_3, s_4, s_5, s_6, s_7, s_8, s_9, s_{10}, s_{11},$$

$$s_{12}, s_{13}, s_{14}, s_{15}-1, s_{16}+1, sem_{B:=0}(d))\,) \mid s_{15}>0 \,\&\, \forall i\in N_{16} \bullet s_i\in N\,\}$$

якщо

$$Inv(S) = (s_3+s_4+s_5+s_6\le 1) \,\&\, (s_2+s_3+s_4+s_5+s_6+s_7+s_8+s_9+s_{10}=P\Rightarrow(d)) \,\&\, (\,(A\Rightarrow(d)=0) \vee$$

$$(\,(A\Rightarrow(d)=1) \,\&\, (s_5=0) \,\&\, (s_6=0)\,)\,) = \text{True},$$

де

$$S=(s_1, s_2, s_3, s_4, s_5, s_6, s_7, s_8, s_9, s_{10}, s_{11}, s_{12}, s_{13}, s_{14}, s_{15}, s_{16}, d),$$

то

$$Inv(S') = (s_3+s_4+s_5+s_6\le 1) \,\&\, (s_2+s_3+s_4+s_5+s_6+s_7+s_8+s_9+s_{10}=P\Rightarrow(d')) \,\&$$

$$(\,(A\Rightarrow(d')=0) \vee (\,(A\Rightarrow(d')=1) \,\&\, (s_5=0) \,\&\, (s_6=0)\,)\,),$$

де $d' = sem_{B:=0}(d) = d\,\nabla\,[\,B\mapsto\ 0\,]$.

Таким чином, $P\Rightarrow(d') = P\Rightarrow(d)$, а $A\Rightarrow(d') = A\Rightarrow(d)$.
Звідси,

$$Inv(S') = (s_3+s_4+s_5+s_6\le 1)\ \&\ (s_2+s_3+s_4+s_5+s_6+s_7+s_8+s_9+s_{10}=P\Rightarrow(d))\ \&$$

$$(\ (A\Rightarrow(d)=0)\ \vee\ (\ (A\Rightarrow(d)=1)\ \&\ (s_5=0)\ \&\ (s_6=0)\ )\ ) = Inv(S) = \text{True}.$$

Доведення завершене.

Маємо:

- $\forall S\in SStartStates \bullet (\ PreCond(S) \rightarrow Inv(S)\ )$,

- $\forall S\in SStates \bullet (\ Inv(S) \rightarrow PostCond(S)\ )$,

- $\forall (S, S')\in SStep \bullet (\ Inv(S) \rightarrow Inv(S')\ )$.

Тобто,

$$InvCond(Inv, PreCond, PostCond) = \text{True}.$$

Звідси $\{A\le 1\}\ Program\ \{A\le 1\}$.
Очевидно, що

$$\forall S\in SStartStates \bullet PreCond(S),$$

адже на початку роботи *Program* записів в таблиці *Auth* немає і $[\ A \mapsto 0\ ]$. Тоді автоматично отримаємо

$$\forall S\in SStopStates \bullet PostCond(S)$$

як наслідок щойно доведеного

$$InvCond(Inv, PreCond, PostCond) = \text{True}.$$

Але

$$\forall S\in SStates \bullet (\ Inv(S) \rightarrow PostCond(S)\ ),$$

звідси необхідна умова ($PostCond(S)$) виконується на кожному досяжному кроці виконання програми *Program*. Тобто, фактично, було доведено більш сильне твердження – а саме, що програма є коректною і не порушує задану умову ($A\le 1$) не лише на початку і в кінці, а і на протязі всього часу виконання, тобто на кожному кроці.

Тотальна коректність

Оскільки програма не має циклів, ані блокувань, ані інших затримуючих або непередбачуваних факторів, які можуть «зациклити», або зупинити, або якимось чином затримати виконання – вона обов'язково завершить роботу за скінчену кількість кроків. Очевидне обмеження на кількість кроків – $n * 9 + m$, де 9 – діаметр графу переходів між мітками програми *Auth_Prog*.

Висновки

Застосовано спрощений метод [7] доведення властивостей interleaving concurrency програм у IPCL [1–3] для доведення властивості коректної роботи банківської системи виплати міжнародних грошових переказів. Це доведення є черговим підтвердженням [8–10] та демонстрацією зручності та адекватності застосування методу для доведення властивостей програмних систем, що виконуються у режимі паралелізму з переключенням та взаємодіють через спільну пам'ять – наприклад, серверних компонент клієнт-серверних комплексів, програм у архітектурі SMP та паралельних середовищ на кшталт суперкомп'ютерів.

1. *Панченко Т.В.* Метод доведення властивостей програм в композиційно-номінативних мовах IPCL // Проблеми програмування. – 2008. – №1. – С. 3–16.
2. *Панченко Т.В.* Методологія доведення властивостей програм в композиційних мовах IPCL // Доповіді Міжнародної конференції "Теоретичні та прикладні аспекти побудови програмних систем" (TAAPSD'2004). – К., 2004. – С. 62–67.
3. *Панченко Т.В.* Композиційні методи специфікації та верифікації програмних систем. – Автореферат дис. … канд. фіз.-мат. наук. – К., 2006. – 17 с.
4. *Беренсон Х., Бернштейн Ф., Грэй Д. и др.* Критика уровней изолированности в стандарте ANSI SQL // СУБД. – 1996. – № 2. – С. 45–60.
5. *Редько В.Н., Брона Ю.Й., Буй Д.Б., Поляков С.А.* Реляційні бази даних: табличні алгебри та SQL-подібні мови. – К.: Видавничий дім "Академперіодика", 2001. – 198 с.
6. *Басараб И.А., Никитченко Н.С., Редько В.Н.* Композиционные базы данных. – К.: Либідь, 1992. – 191 с.
7. *Панченко Т.В.* Модель спрощеного стану для методу доведення властивостей в мовах IPCL та її застосування і переваги // Доповіді міжнародної наукової конференції TAAPSD`2007. – Berdyansk, 2007. – С. 319–322.
8. *Polishchuk N.V., Kartavov M.O. and Panchenko T.V.* Safety Property Proof using Correctness Proof Methodology in IPCL // Proceedings of the 5th International Scientific Conference "Theoretical and Applied Aspects of Cybernetics". – Kyiv: Bukrek, 2015. – P. 37–44.

9. *Картавов М.О., Панченко Т.В., Поліщук Н.В.* Доведення тотальної коректності системи Infosoft e-Detailing у IPCL // Вісник Київського національного університету імені Тараса Шевченка. Серія: фіз.-мат. науки. – 2015. – Вип. 3. – С. 80–83.
10. *Kartavov M.* Properties Proof Method in IPCL Application To Real-World System Correctness Proof / M. Kartavov, T. Panchenko, N. Polishchuk // International Journal "Information Models and Analyses". – Sofia, Bulgaria: ITHEA. – 2015. – Vol. 4, N 2. – P. 142–155.

References

1. PANCHENKO, T. (2008) The Method for Program Properties Proof in Compositional Nominative Languages IPCL [in Ukrainian]. Problems of Programming. 1. pp. 3–16.
2. PANCHENKO, T. (2004) The Methodology for Program Properties Proof in Compositional Languages IPCL [in Ukrainian]. In Proceedings of the International Conference "Theoretical and Applied Aspects of Program Systems Development" (TAAPSD'2004). Kyiv. pp. 62–67.
3. PANCHENKO, T. (2006) Compositional Methods for Software Systems Specification and Verification (PhD Thesis Synopsis) [in Ukrainian]. Kyiv. 17 p.
4. BERENSON, Kh., BERNSTEIN, F. And GREY, D. (1996) Critique of Isolation Levels in ANSI SQL Standard [in Russian]. DBMS, no. 2, pp. 45–60.
5. REDKO, V., BRONA, Yu., BUY, D. and POLYAKOV, S. (2001) Relational Databases: Table Algebras and SQL-like Languages [in Ukrainian]. Kyiv, "Akademperiodika", 198 p.
6. BASARAB, I., NIKITCHENKO, M. and REDKO, V. (1992) Compositional Databases [in Russian]. Kyiv, "Lybid", 191 p.
7. PANCHENKO, T. (2007) The Simplified State Model for Properties Proof Method in IPCL Languages and its use and advantages [in Ukrainian]. In Proceedings of the International Conference "Theoretical and Applied Aspects of Program Systems Development" (TAAPSD'2007). Berdyansk. pp. 319–322.
8. POLISHCHUK, N.V., KARTAVOV, M.O. and PANCHENKO, T.V. (2015) Safety Property Proof using Correctness Proof Methodology in IPCL. In Proceedings of the 5th International Scientific Conference "Theoretical and Applied Aspects of Cybernetics", Kyiv, Bukrek, pp.37-44.
9. KARTAVOV, M.O., PANCHENKO, T.V. and POLISHCHUK, N.V. (2015) Infosoft e-Detailing System Total Correctness Proof in IPCL [in Ukrainian], Bulletin of Taras Shevchenko National University of Kyiv. Series: Physical and Mathematical Sciences, no. 3, pp. 80–83.
10. KARTAVOV, M., PANCHENKO, T. and POLISHCHUK, N. (2015) Properties Proof Method in IPCL Application To Real-World System Correctness Proof. International Journal "Information Models and Analyses", Sofia, Bulgaria, ITHEA, Vol. 4, no. 2, pp. 142–155.

Про авторів:

Остаповська Юлія Аркадіївна,
студентка 4 курсу кафедри Теорії та технології програмування факультету кібернетики.
Кількість наукових публікацій в українських виданнях – 1.
http://orcid.org/0000-0002-8865-2139,

Панченко Тарас Володимирович,
кандидат фізико-математичних наук, доцент,
доцент кафедри Теорії та технології програмування факультету кібернетики.
Кількість наукових публікацій в українських виданнях – 27.
Кількість наукових публікацій в іноземних виданнях – 2.
http://orcid.org/0000-0003-0412-1945,

Поліщук Наталія Володимирівна,
студентка 4 курсу кафедри Теорії та технології програмування факультету кібернетики.
Кількість наукових публікацій в українських виданнях – 3.
Кількість наукових публікацій в іноземних виданнях – 1.
http://orcid.org/0000-0001-6936-0053,

Картавов Микита Олексійович,
студент 4 курсу кафедри Теорії та технології програмування факультету кібернетики.
Кількість наукових публікацій в українських виданнях – 3.
Кількість наукових публікацій в іноземних виданнях – 1.
http://orcid.org/0000-0002-2020-3468.

Місце роботи авторів:

Київський національний університет імені Тараса Шевченка,
03680, Київ, проспект Академіка Глушкова, 4Д.
Тел.: 259 0519.
E-mail: tp@infosoft.ua