Формальна специфікація системи "Автоматичний шлагбаум". Реалізація системи у вигляді мережі таймованих автоматів за допомогою "UPPAAL". Формальна верифікація реалізації

Третяков Єгор

6 травня 2025 р.

Вступне слово. Застереження

У даній роботі представлено опис системи "автоматичний шлагбаум сформований виключно на основі особистих уявлень автора щодо можливого принципу її функціонування. Матеріал містить значну кількість припущень і не претендує на точність або повну відповідність реальним технічним реалізаціям. Автор не несе відповідальності за використання цієї інформації в практичних або інженерних цілях.

Специфікацію системи подано із застосуванням мови комбінаційної темпоральної логіки ієрархічних номінативних даних. Реалізацію (симуляцію) та верифікацію виконано у вигляді мережі таймованих автоматів із використанням середовища UPPAAL.

Специфікація

Виходячи з характеру системи, а саме з ознаки "автоматичності"в її назві, систему "автоматичний шлагбаум"доцільно описувати за допомогою логіки, здатної оперувати часовими аспектами. Найприроднішим вибором для цього є темпоральна логіка. З огляду на те, що подальша реалізація системи здійснюється у вигляді автоматів — одного з найпростіших способів формального задання систем, придатних для верифікації темпоральних властивостей — як основу (носія) для формалізації специфікації було обрано алгебру ієрархічних номінативних даних.

Система "автоматичний шлагбаум"виконує, по суті, одну основну функцію — відкриватися у разі наближення автомобіля чи іншого об'єкта та залишатися відкритою доти, доки перед нею присутній цей об'єкт. Закриття шлагбаума допускається лише за умови, що зона перед ним є повністю вільною. У межах цієї моделі вважається, що шлагбаум обслуговує лише один напрямок руху.

Будемо вважати, що шлагбаум може перебувати в одному з чотирьох станів (*State*):

- Відкритий (*Opened*) — шлагбаум знаходиться в положенні, що дозволяє пропускати об'єкти;

- Закритий ($Closed$) — шлагбаум знаходиться в положенні, що перешкоджає пропуску об'єктів;
- Відкривається ($Opening$) — шлагбаум знаходиться в процесі відкриття;
- Закривається ($Closing$) — шлагбаум знаходиться в процесі закриття;

Також припустимо, що шлагбаум оснащений сенсором, який відповідає за фіксацію наявності об'єкта поруч. Позначимо ці елементи як BG (шлагбаум) та $Sensor$ (сенсор).

Нехай сенсор ($Sensor$) може бути у наступних станах:

- Об'єкт зафіксовано ($Detected$) — сенсор виявляє об'єкт поруч зі шлагбаумом;
- Об'єкт не зафіксовано ($NotDetected$) — сенсор не виявляє об'єкт поруч зі шлагбаумом;

Тобто, задамо $BG \equiv (State, Signal)$, $State \equiv \{Opened,\ Closed,\ Opening,\ Closing\}$, $Signal \equiv \{Detected,\ NotDetected\}$.

Умовами коректності роботи шлагбаума вважатимемо настпне:

- Якщо сенсор шлагбауму фіксує об'єкт поруч, то це означає, що шлагбаум відкритий або відкривається;
- Якщо сенсор шлагбауму не фіксує об'єкт поруч, то це означає, що шлагбаум закритий або закривається;

Або ж, опишемо це мовою комбінаційної темпоральної логіки:

- $AG(BG.Signal = Detected \implies BG.State = Opened \vee BG.State = Opening)$;
- $AG(BG.Signal = NotDetected \implies BG.State = Closed \vee BG.State = Closing)$;

Можуть виникнути суперечки щодо доцільності заміни імплікації в умові на еквівалентність, однак автор вирішив залишити імплікацію, вважаючи, що цього підходу достатньо для коректного опису. Таке життя.

Реалізація

На основі попередньо визначеної специфікації була побудована модель системи у вигляді мережі таймованих автоматів. Такий підхід дає змогу формально описати часову поведінку елементів системи та їхню взаємодію, враховуючи обмеження реального часу. Модель реалізує ключові компоненти системи "автоматичний шлагбаум зокрема механізм виявлення об'єкта, затримки на відкриття та закриття шлагбаума, а також взаємодію з сигналами управління. Реалізація системи була виконана в середовищі "UPPAAL".

Моделювання шлагбаума та його сенсора здійснюється окремо. Водночас, для узгодження їхньої роботи використано механізми синхронізації, що надаються середовищем "UPPAAL". З метою наближення моделі до реальної поведінки сенсора, було вирішено мінімально змоделювати також і його оточення — транспортні засоби, які можуть наближатися до шлагбаума. Модель передбачає наявність лише авто. Таке життя.

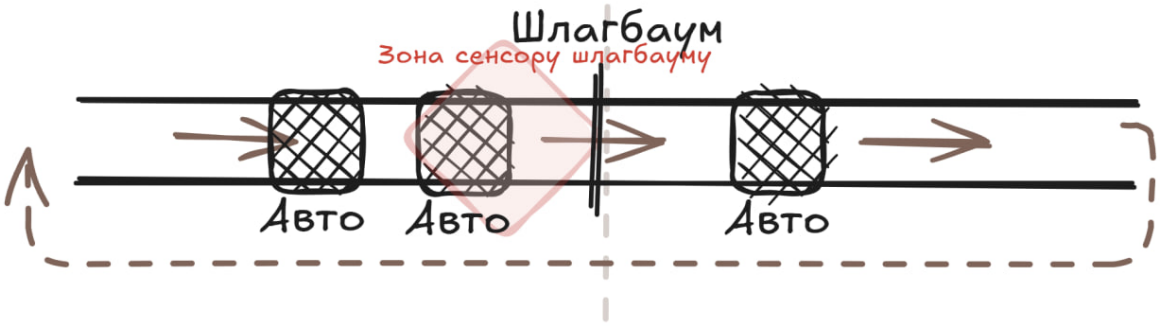

Рис. 1: Схематичне зображення системи "автоматичний автомат"

Легко переконатись, що насправді для моделювання роботи системи достатньо всього двох авто, які можуть їздити кільцевою дорогою.

Сенсор моделюється за допомогою кількох автоматів. Один з автоматів відповідає за визначення наявності автомобіля в зоні сенсора шлагбауму, в той час як інший виступає у ролі тактового генератора, який періодично ініціює передачу інформації від сенсора до шлагбауму. Інформацію про наявність об'єкта в зоні сенсора сенсор отримує з зовнішнього середовища. У нашому випадку ця інформація зберігається в глобальній змінній $car\_is\_seen\_by\_sensor$, значення якої змінюються автомобілями, коли вони наближаються до шлагбауму. Точну реалізацію автоматів можна побачити на рис. 2 та рис. 3.

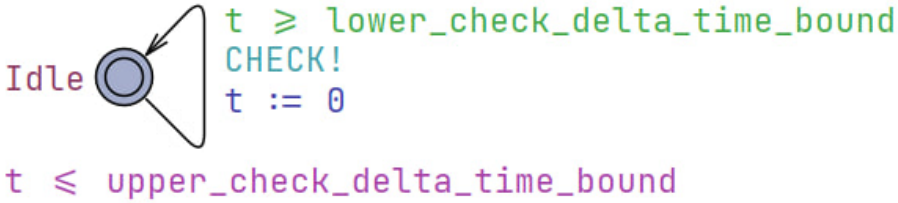

Рис. 2: Реалізація автомату, що задає тактовий генератор сенсору автоматичного шлагбауму

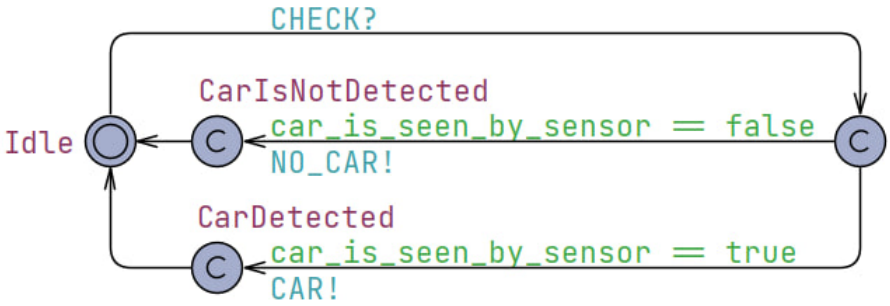

Рис. 3: Реалізація автомату, що задає сенсор автоматичного шлагбауму

Реалізація шлагбауму не має суттєвих особливостей, що потребують окремого висвітлення. Детальнішу реалізацію можна переглянути на рис. 4.

Найцікавішою складовою моделі є автомобілі. Їх поведінку представлено на рис.5. Автомобілі переміщуються між окремими ділянками, які доцільно інтерпретувати як

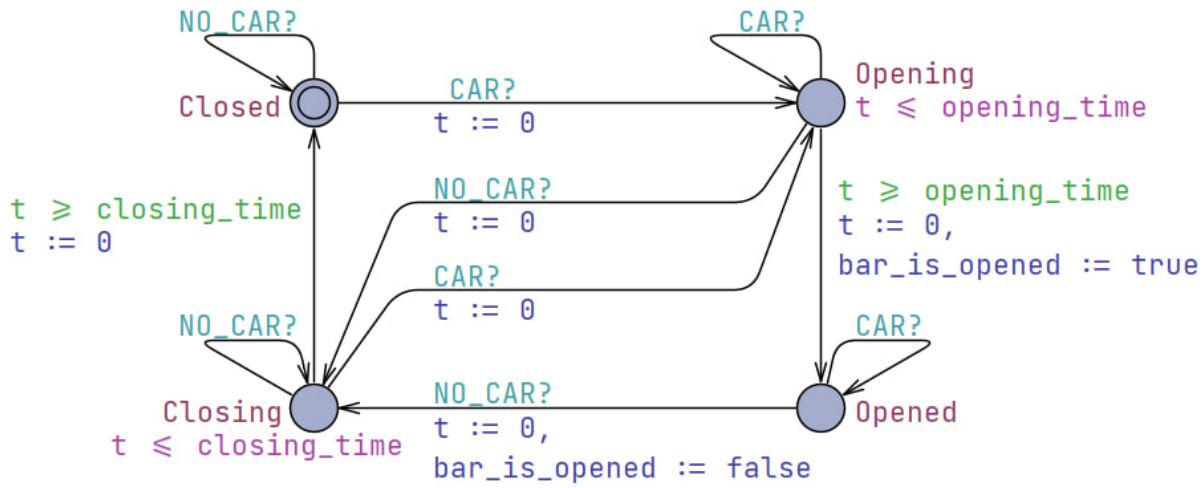

Рис. 4: Реалізація автомату, що задає шлагбаум (стрілу шлагбауму)

зони, оскільки точне положення всередині кожної з них не є суттєвим для роботи системи. Схематичне розташування цих зон наведено на рис.6. Оскільки саме зона перебування є визначальною, кожен стан автомата, що моделює автомобіль, відповідає певній зоні:

- Вільний світ. Всі катаються (*MovingAround*) — Авто знаходиться у зоні, що нас не цікавить у контексті взаємодії з шлагбаумом;
- Очікування на шлагбаум (*NearGate*) — автомобіль знаходиться біля шлагбаума та проїде далі, щойно шлагбаум відкриється;
- Черга до шлагбауму (*InQueue*) — автомобіль чекає перед іншим автомобілем і рушає вперед, коли той покидає зону очікування;
- Виїзд з шлагбауму (*PassingGate*) — автомобіль знаходиться в зоні, де шлагбаум ще не можна безпечно закривати, хоча сам об'єкт уже не перешкоджає проїзду;

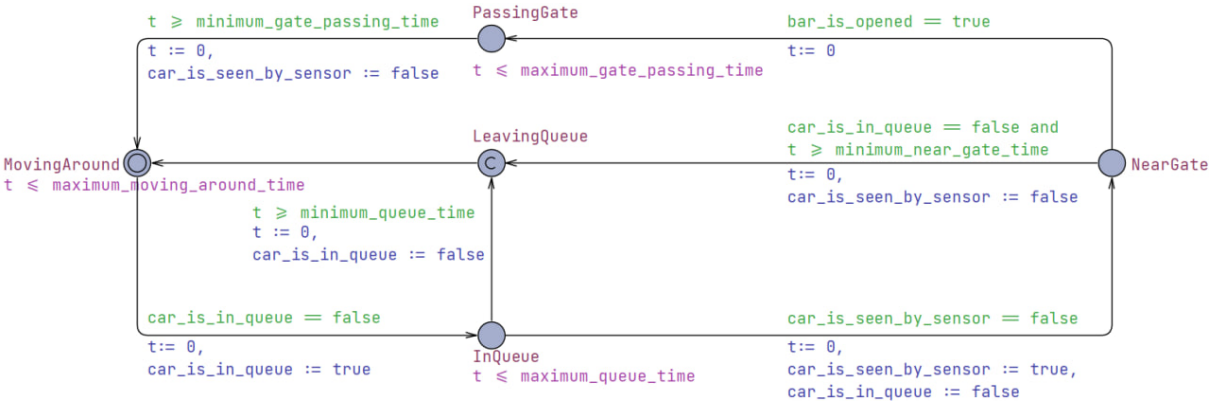

Рис. 5: Реалізація автомату, що імітує авто

Рис. 6: Схематичне зображення важливих зон з погляну авто

Верифікація

За допомогою вбудованого функціоналу "UPPAAL"можна здійснити формальну верифікацію — перевірку істинності виразів темпоральної логіки стосовно поведінки заданої моделі. У нашому випадку модель системи має такий вигляд:

- Два авто (Car_1, Car_2);
- Шлагбаум (Bar_gate);
- Тактовий генератор сенсора ($Sensor\_clock$);
- Сенсор шлагбауму (Car_sensor);

Те, як це виглядає у "UPPAAL"— представлено на рис.7 (з реалізацією певними цілочисельними константами).

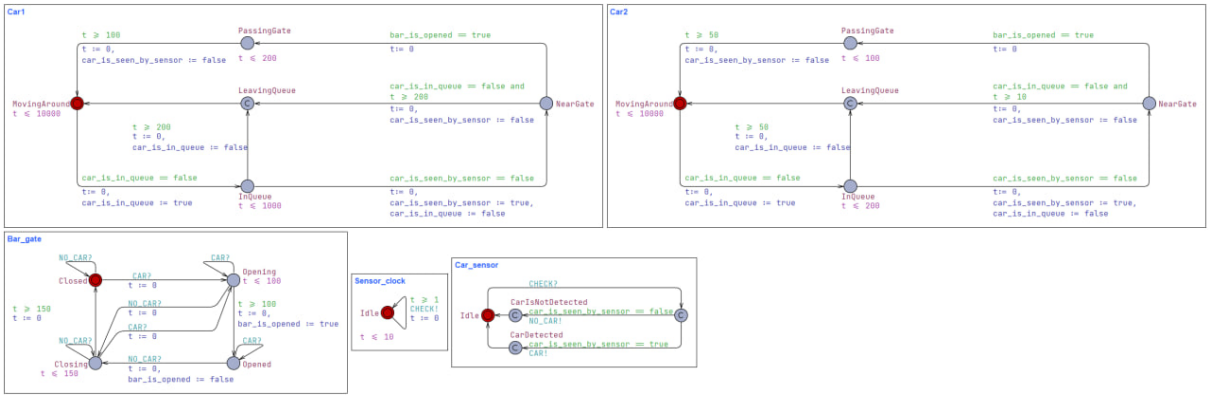

Рис. 7: Реалізація системи "автоматичний шлагбаум"

Логічні умови (описано у розділі специфікація) мовою, що використовується у "UPPAAL"буде записано наступним чином:

- $A[]\ Car\_sensor.CarDetected\ imply\ ((Bar\_gate.Opening\ or\ Bar\_gate.Opened)\ and\ not\ (Bar\_gate.Closed\ or\ Bar\_gate.Closing))$;
- $A[]\ Car\_sensor.CarIsNotDetected\ imply\ (not\ (Bar\_gate.Opening\ or\ Bar\_gate.Opened)\ and\ (Bar\_gate.Closed\ or\ Bar\_gate.Closing))$;

Те, як це виглядає у "UPPAAL"— представлено на рис.8 (Разом із верифікацією того, що авто не перетинаються у зонах).

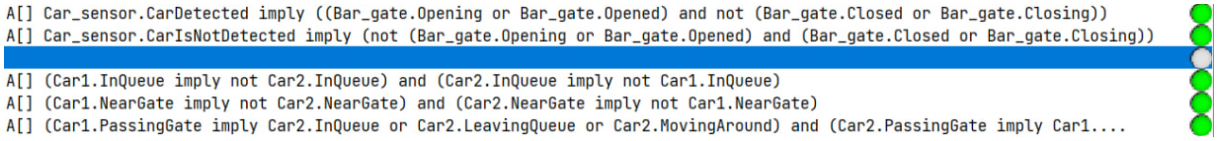

Рис. 8: Результат верифікації умов темпоральної логіки за допомогою функціоналу середовища "UPPAAL"