Titolo

Autore1, Autore2, Autore3

Academic Year 2022-2023

images/title.png

1 Introduction

The work consists in developing a formal model of a Lego Mindstorms production plant, verifying key properties, and analyzing the behavior of the plant under different configurations, both deterministic and stochastic, using the Uppaal tool, used for modeling and verifying real-time systems, involving two phases: exhaustive model checking and statistical model checking.

The production plant operates with a one-directional conveyor belt that carries workpieces through different processing stations. Each station can process only one piece at a time, and the duration of processing may vary between stations. The flow of workpieces is controlled by a flow controller that can either direct them to a specific station or divert them to an alternative branch of the belt. Laser-based sensors detect the presence of pieces and prevent station overcrowding by blocking new pieces from entering when a station is busy.

In the stochastic version of the production plant, certain aspects of the system involve randomness or uncertainty. The processing time at each station follows a probabilistic distribution, specifically a normal distribution. Additionally, there is a probability of sensor malfunctions, where sensors may detect or fail to detect pieces incorrectly.

2 Model Description

2.1 Data Structures

In our system, we rely on five key arrays to effectively manage the production plant. These data structures serve as integral components for representing and tracking the state and positions of various elements within the system. The most vital data structure is the `belt` array, which serves as a representation of the conveyor belt in the production plant. It consists of a boolean array where each element corresponds to a specific slot on the belt. The binary value assigned to each element (0 or 1) signifies the presence or absence of a workpiece within the respective slot.

The remaining arrays, namely `stations`, `in_sensors`, `out_sensors`, and `preprocessing`, follow a similar pattern and have the same length as the conveyor belt. These arrays are essential for accurately recording the positions and statuses of associated elements. For instance, if a station occupies slot 16, the element at index 16 in the `stations` array will store the unique identifier (ID) of the station. The same principle applies to the other arrays, facilitating the identification and tracking of input sensors, output sensors, and preprocessing components.

By utilizing these data structures, we establish a robust framework to effectively manage, monitor, and synchronize the state and locations of various elements within the production plant.

The auxiliary arrays `status_stations`, `status_in_sensors`, and `status_preprocessing` are utilized to determine the occupation status of the corresponding components they represent. These arrays have a length equal to the total number of components they represent, and each position corresponds to the ID of the represented component. A value of 1 indicates that the component is occupied, while a value of 0 indicates that it is available. By monitoring these arrays, the system can effectively track the occupancy status of the components.

2.2 Channels

The system utilizes several channels:

- `enter_preprocessing`: This channel is indexed by the number of stations in the system. It is used to signal the entry of a workpiece into the preprocessing stage.

- `free_in_sensor`: This channel is indexed by the number of input sensors in the system. It is used to indicate the availability of an input sensor for processing a workpiece.

- `sensor_busy`: This channel is also indexed by the number of input sensors in the system. It is used to notify the system that an input sensor is currently occupied and cannot process a new workpiece.

- `step`: This is a broadcast channel used to synchronize the system steps. It enables the components to advance together in a synchronized manner.

- `initialize`: This is a broadcast channel used for system initialization. It triggers the initialization process and sets the parameters for the system components.

2.3 Conveyor Belt

2.4 Processing Station

2.5 Initializer

2.6 Assumptions

3 Evaluation and Results

3.1 Queries explanation

3.2 Scenario 1

3.3 Scenario 2

3.4 Scenario 3