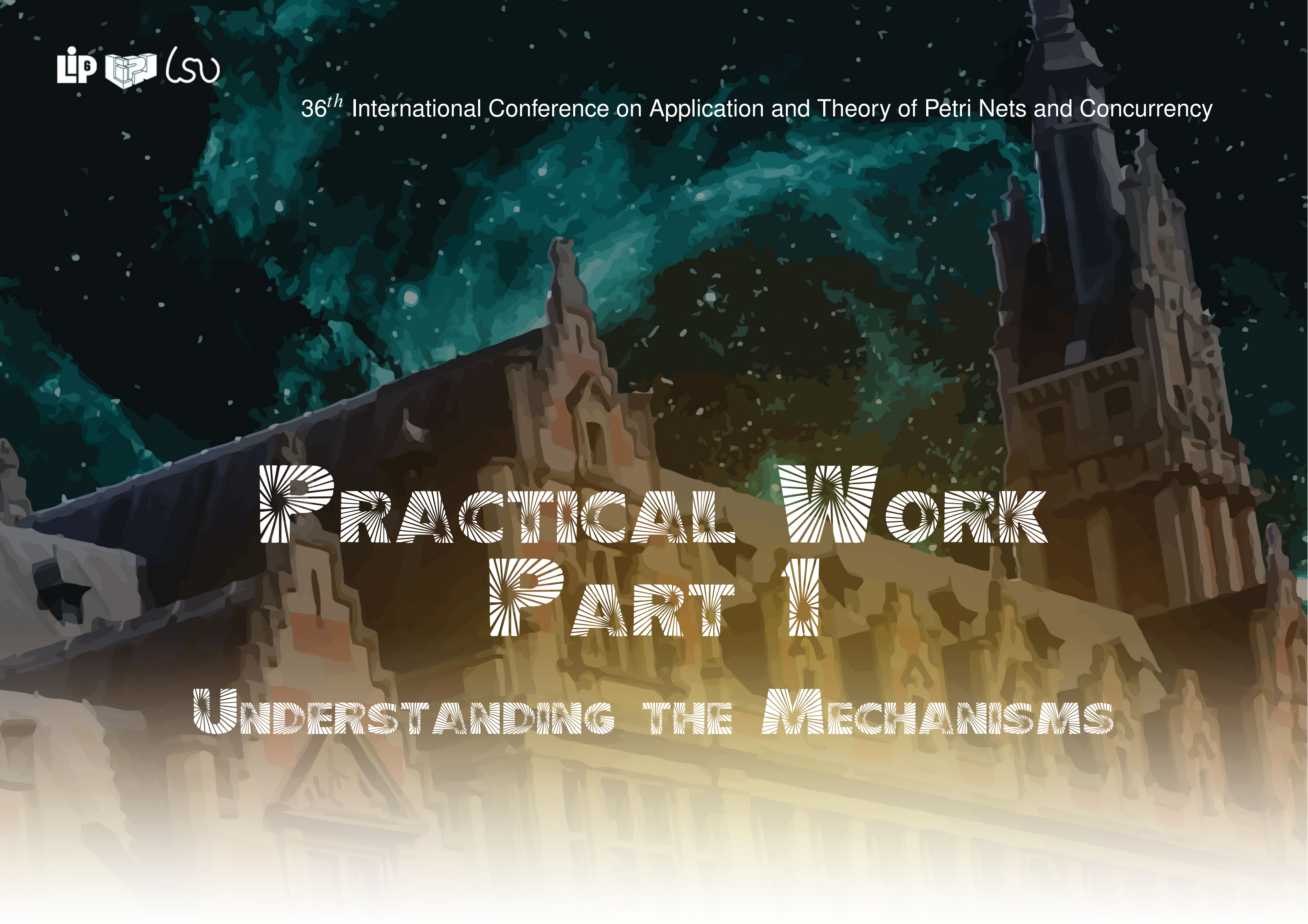

36th International Conference on Application and Theory of Petri Nets and Concurrency

Practical Work Part 1

Understanding the Mechanisms

Contents

1 A Bicycles Sharing Service

The Université Libre de Bruxelles has a new project to improve its commitment to its guests: a service of bicycles sharing. It raises the problem of the availability of the bicycles: when one is being used by a guest, it cannot profit to another guest at the same time. This is what is called a *shared resource*. To help the hotel to develop this new service, we propose to model the problem using Petri nets.

As any good modeller, we start by considering the simple case: there is only one bicycle, and only one guest (we will extend this simple case latter).

Question 1.1: Simple Case Model. Model the problem as a Petri net. There should be one place for the idle guest, one place for the idle bicycle, one place to indicate a bicycle in use, and two transitions: one to borrow a bicycle, and the other to return it.

Check the syntax of your model through the services of the platform.

Question 1.2: State Space. Build the reachability graph with the tool Prod, in the submenu "behavioural properties". You should first ask to build the graph then request the visualisation.

Question 1.3: Increasing the Number of Guests. We now have several guests. Modify your model to include them (you can limit yourself to 2 or 3 guests). Note that it is important to keep track of the identity of the guest using a given bicycle, as a penalty applies when a bicycle is damaged or not returned (and the hotel certainly does not want to charge the wrong customer. Seriously, what a mess it would be?)

What is the incidence of this modification on the size of the net?

Question 1.4: Increasing the Number of Bicycles. Following the same principle, how could you modify your net to include several bicycles? Do you consider drawing the net for as few as 10 guests and 5 bicycles? And for more realistic values such as 200 guests and 50 bicycles?

Question 1.5: Adopting an Appropriate Approach. If you have answered "yes" to the last question, please proceed to the reception in order to discuss your wage for the months it will take you to draw the whole net. Otherwise, simply proceed with the rest of the exercise.

In the above approach, the identity of a guest or a bicycle is modelled by a new place. Instead, their identities can be modelled by adding information to the token in the same place. This additional information is called a *colour*. The set of guests identities (or room numbers) is a first colour class G. The set of bicycles identities is a second colour class B. Each token in the guests place is assigned a colour in G (a room number), and each bicycle is assigned a colour in B (a bicycle ID). The place indicating which guest is using which bicycle contains tokens whose colour is a pair in the set $G \times B$.

Transform your first model into a Symmetric net (SN) by adding this information on colours. Notice how its size gently does not grow.

Question 1.6: Coloured State Space. Using the platform, check the syntax of your model. Generate and visualise the graph of accessible markings for $|G| = 2$ and $|B| = 1$. Compare with the graph of accessible markings for the model of Question 1.3.

Question 1.7: Unfolding. Considering the semantic equivalence (the state spaces) and the modelling equivalence of the identities, what do you think of the term of "unfolding with respect to the colours"?

2 Modelling a Swimming Pool (for those who are fast)

We would like to model a swimming pool where customers use:

- booths to change clothes;
- the pool to bathe.

A customer behaves as follows:

1. enter the swimming pool;
2. ask the key of a booth that he/she will keep until he/she leaves;
3. enter the booth and change;
4. bathe;
5. enter the booth, dry off and change;
6. return the key of the booth;
7. leave the swimming pool and go on with his/her life.

We would like to study the dynamics of this system, and in particular its drawbacks. We assume there are 10 potential customers and 8 booths.

Question 2.1: Modelling the System. Model the behaviour of a customer and his/her usage of the different resources. Take care to model only what is necessary. Do you note a similarity with a previously studied system?

Question 2.2: Analysing a Property. What is the maximal capacity of the pool? How do you compute it?

The manager of the swimming pool wishes to optimise the procedure and adopts a new operating mode based on shared booths and baskets to store customers's clothes.
Now a customer behaves as follows:

1. enter the swimming pool;
2. ask for a booth and get a key;
3. ask for a basket that he/she will keep until he/she leaves;
4. enter the booth, change, then free the booth by returning the key;
5. store the basket with his/her clothes at the lockers office;
6. bathe;
7. ask for a new booth and get a second key;
8. get the basket with his/her clothes back from the lockers office;

9. enter the second booth, dry off, change, then free the booth by returning the key;
10. return the basket;
11. leave the swimming pool and go on with his/her life.

We now consider 4 baskets, 2 booths and 6 potential clients.

Question 2.3: Modelling the System. Model the new behaviour of the customer and his/her usage of the different resources.

Question 2.4: Analysing the Behaviour. Compute the graph of reachable markings. What do you observe? Use the different services offered by Prod to propose a diagnosis. You may attempt, by identifying states with numbers, a scenario of what happens (by convention, the initial state is state 0).

Hint: Before trying to visualise the graph, consider its size...

Question 2.5: Impact on the Real System. Help the manager to modify the specification to avoid the problem you have identified.

Question 2.6: A user entering the swimming pool always get into the bath. Write the CTL or LTL property stating that any user entering the swimming pool building will get into the pool.