Exercise 16 (3 Points) Consider the LTL following transition system over the action $A = \{\alpha, \beta, \gamma\}$ (without atomic propositions).

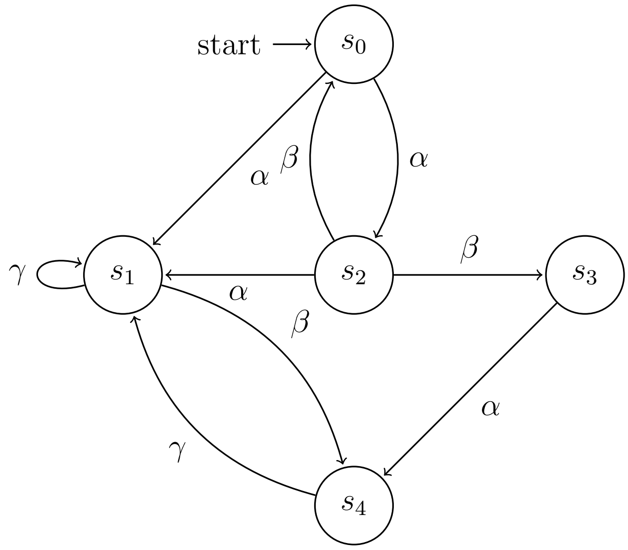

Figure 1: Transition system T

Definition 1. *Let T be a transition system.*

A fairness property $\mathcal{F}$ below is a triple $(\mathcal{F}_{uncond}, \mathcal{F}_{strong}, \mathcal{F}_{weak})$, where:

unconditional fairness $\mathcal{F}_{uncond} \subseteq 2^A$ *is a set of actions that are unconditionally A-fair: For every infinite execution fragment $s_0 \xrightarrow{\alpha_1} s_1 \xrightarrow{\alpha_2} s_2 \cdots$: $\exists^\infty j : \alpha_j \in A$*

strong fairness $\mathcal{F}_{strong} \subseteq 2^A$ *is a set of actions that are strongly A-fair: For every infinite execution fragment $s_0 \xrightarrow{\alpha_1} s_1 \xrightarrow{\alpha_2} s_2 \cdots$: $\exists^\infty j : \{s_j \xrightarrow{\alpha_j} s_{j+1} \mid \alpha_j \in A\} \neq \emptyset$, then $\exists^\infty k : \alpha_k \in A$*

weak fairness $\mathcal{F}_{strong} \subseteq 2^A$ *is a set of actions that are weakly A-fair: For every infinite execution fragment $s_0 \xrightarrow{\alpha_1} s_1 \xrightarrow{\alpha_2} s_2 \cdots$: $\exists j : \forall k > j : \{s_k \xrightarrow{\alpha_k} s_{k+1} \mid \alpha_j \in A\} \neq \emptyset$, then $\exists^\infty j : \alpha_j \in A$*

Definition 2. *Let T be a transition system with the set of actions A and $\mathcal{F}$ a fairness assumption for A. $\mathcal{F}$ is called* realisable *for T if for every reachable state s there exists an infinite execution fragment starting in s and satisfying $\mathcal{F}$.*

Decide, which one of the following fairness assumptions $\mathcal{F}_i$ is realisable for T.

$$\mathcal{F}_1 = (\{\{\alpha\}\}, \{\{\gamma\}\}, \{\{\alpha, \beta\}\}) \tag{1}$$

$$\mathcal{F}_2 = (\{\{\alpha, \gamma\}\}, \{\{\alpha, \beta\}\}, \{\{\gamma\}\}) \tag{2}$$

$$\mathcal{F}_3 = (\{\{\alpha, \gamma\}, \{\beta\}\}, \{\{\alpha, \beta\}\}, \{\{\gamma\}\}) \tag{3}$$

.

Exercise 17 (4 Points) Prove the following theorem:

Let $\mathcal{F} = \{A_1, A_2, \ldots, A_k\} \subseteq 2^A$ be fairness assumption. Define $\mathcal{F}_{uncond} = (\mathcal{F}, \emptyset, \emptyset)$, $\mathcal{F}_{strong} = (\emptyset, \mathcal{F}, \emptyset)$, and $\mathcal{F}_{weak} = (\emptyset, \emptyset, \mathcal{F})$. Prove for every linear-time property and any transition system T:

1. $T \models_{\mathcal{F}_{weak}} P \implies T \models_{\mathcal{F}_{strong}} P$
2. $T \models_{\mathcal{F}_{strong}} P \implies T \models_{\mathcal{F}_{uncond}} P$

Hint: Is a path that is strongly fair also a path that is weakly fair? What does that mean for the implication?

Find for each reverse implication above a counter example, thus showing that unconditional fairness does not imply strong fairness and strong fairness does not imply weak fairness.

Exercise 18 (8 Points) Consider the LTL formula $\varphi \triangleq \Box(a \rightarrow (\neg b \mathcal{U}(a \wedge b)))$ over the set of propositions $P = \{a, b\}$ and check $T \models \varphi$ for the transition system T displayed below.

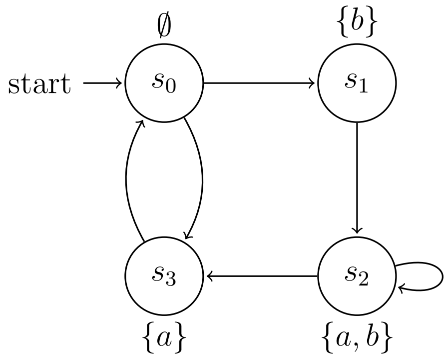

Figure 2: Transition system T

1. Transform the formula into an equivalent basic LTL formula ψ, i.e. one that is accepted by the grammar:
$$\psi ::= \mathit{true} \mid a \mid b \mid \psi \wedge \psi \mid \neg\psi \mid \bigcirc\psi \mid \psi \mathcal{U} \psi$$
2. Give the elementary sets with respect to $\mathit{closure}(\psi)$.
3. Construct the GNBA $\mathcal{G}_\psi$
4. Construct the NBA $\mathcal{A}_{\neg\varphi}$. You may build the NBA *directly* from $\neg\varphi$, without relying on $\mathcal{G}_\psi$. Hint: Four states suffice.
5. Construct $T \otimes \mathcal{A}_{\neg\varphi}$
6. Check whether $T \models \varphi$. Sketch the algorithm's main steps and interprete its outcome!

Exercise 19 (8 Points) We assume N processes in a ring topology, connected by unbounded queues. A process can only send messages to its clockwise neighbour. Initially, each process has a unique identifier *ident* (which is assumed to be a natural number). A process can either be active or relaying. Initially all processes are active. In Peterson's leader election algorithm (1982) in the ring carries out the following task:

```
  d = ident;
  for (;;) {
    send(d);
    receive(e);
    if (e == ident) announce elected;
    if (d > e) then send(d) else send(e);
    receive(f);
    if (f == ident) announce elected;
    if (e >= max(d, f)) d = e else goto relay;
  }
relay:
  for (;;) {
    receive(d);
    if (d == ident) announce elected;
    send(d);
  }
```

Solve the following problems concerning this leader election problem using SPIN.

1. Model Peterson's leader election protocol in Promela. Avoid invalid end states.
2. Verify the following properties:
   (a) There is at most one leader.
   (b) Eventually always a leader will be elected.
   (c) The elected leader will be the process with the highest number.
   (d) The maximum total amount of messages sent in order to elect the leader is at most $2N\lfloor \log_2 N \rfloor + N$
3. Which is the largest N for which your verification was successful?

Handing in this Assignment Please submit your hand-written solutions to exercise 16 and 17 on paper no later than November 27, 2009, 12:15 (before the lecture).

The models shall be placed in a directory that carries the last name of one of the group members. Add a `README` file, or better, a `Makefile`, that explains or automates the modelling and checking procedures. Explain, how to interprete the results of model checking in an accompanying PDF or ASCII file.

Put all this into a tape archive that shares the name with the directory and send it by e-mail to marcel.kyas@fu-berlin.de. Use "Model checking 09 Series 5 *your last names*" as the subject line.