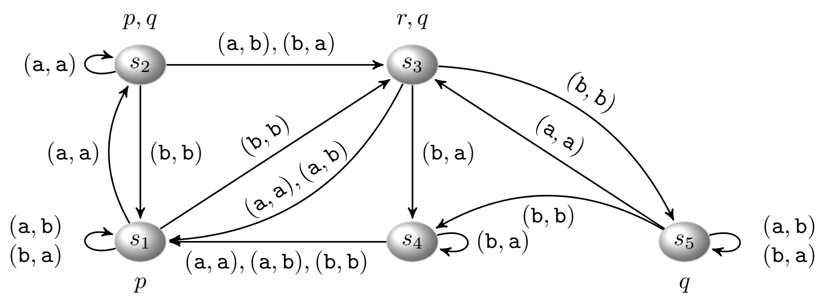
TD: Logical Aspects of Artificial Intelligence
Logics for Multi-Agent Systems
(05/10/2022)

Exercise 1. Consider the CGS below with two agents, two actions $\mathtt{a}, \mathtt{b}$ and the propositional variables p, q, r.



Determine which statements below hold.

1. $\mathfrak{M}, s_1 \models p \wedge \langle\!\langle 1 \rangle\!\rangle \mathsf{X} q$
2. $\mathfrak{M}, s_2 \models \langle\!\langle 1 \rangle\!\rangle (p \vee r) \mathsf{U} \neg q$
3. $\mathfrak{M}, s_1 \models \langle\!\langle 1 \rangle\!\rangle \mathsf{F} \neg \langle\!\langle 2 \rangle\!\rangle \mathsf{X} \neg p$
4. $\mathfrak{M}, s_1 \models \langle\!\langle 1 \rangle\!\rangle \mathsf{G} p \wedge \langle\!\langle 2 \rangle\!\rangle \mathsf{G} p \wedge \langle\!\langle 1, 2 \rangle\!\rangle \mathsf{F} \neg p$
5. $\mathfrak{M}, s_2 \models \neg \langle\!\langle 1 \rangle\!\rangle \mathsf{X}(q \wedge r) \wedge \neg \langle\!\langle 2 \rangle\!\rangle \mathsf{X} p \wedge \neg \langle\!\langle 1, 2 \rangle\!\rangle \mathsf{X}(p \vee r)$
6. $\mathfrak{M}, s_3 \models \langle\!\langle 1 \rangle\!\rangle \mathsf{G} \langle\!\langle 1, 2 \rangle\!\rangle (\neg q \mathsf{U} p)$

Exercise 2. Let $\mathfrak{M} = (Agt, S, Act, \mathtt{act}, \delta, L)$ be a concurrent game structure (CGS) $A, A' \subseteq Agt$ be coalitions such that $A \cap A' = \emptyset$, $s \in S$ and φ, φ' be ATL formulae built over coalitions from Agt.

1. Show that if $\mathfrak{M}, s \models (\langle\!\langle A \rangle\!\rangle \mathsf{G} \varphi) \wedge (\langle\!\langle A' \rangle\!\rangle \mathsf{G} \varphi')$ then $\mathfrak{M}, s \models \langle\!\langle A \cup A' \rangle\!\rangle \mathsf{G}(\varphi \wedge \varphi')$.
2. Is it always the case that if $\mathfrak{M}, s \models (\langle\!\langle A \rangle\!\rangle \mathsf{F} \varphi) \wedge (\langle\!\langle A' \rangle\!\rangle \mathsf{F} \varphi')$ then $\mathfrak{M}, s \models \langle\!\langle A \cup A' \rangle\!\rangle \mathsf{F}(\varphi \wedge \varphi')$?

Exercise 3. (from exam 2021/2022) Given a concurrent game structure $\mathfrak{M} = (Agt, S, Act, \mathtt{act}, \delta, L)$, coalitions $A \subseteq A' \subseteq Agt$ and a state $s \in S$, show that

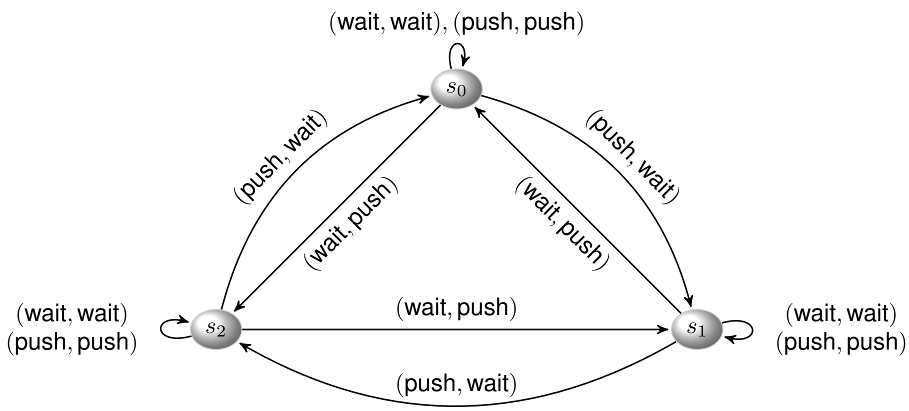
$$\mathfrak{M}, s \models (\langle\!\langle \emptyset \rangle\!\rangle \mathsf{X} p \wedge \langle\!\langle A \rangle\!\rangle \mathsf{X} q) \Rightarrow \langle\!\langle A' \rangle\!\rangle \mathsf{X}(p \wedge q).$$

Exercise 4. Show that $\langle\!\langle A \rangle\!\rangle \mathsf{X} \varphi \wedge \langle\!\langle A' \rangle\!\rangle \mathsf{X} \varphi' \Rightarrow \langle\!\langle A \cup A' \rangle\!\rangle \mathsf{X}(\varphi \wedge \varphi')$ is valid when $A \cap A' = \emptyset$.

Exercise 5. Let $\mathfrak{M}$ be a CGS, φ, ψ be ATL formulae and $A \subseteq Agt$. Show the following characterisations.

1. $[\![\langle\!\langle A\rangle\!\rangle \mathsf{G}\varphi]\!]^{\mathfrak{M}} = \nu Z.([\![\varphi]\!]^{\mathfrak{M}} \cap \mathtt{pre}(\mathfrak{M}, A, Z))$.
2. $[\![\langle\!\langle A\rangle\!\rangle \varphi \mathsf{U} \psi]\!]^{\mathfrak{M}} = \mu Z.([\![\psi]\!]^{\mathfrak{M}} \cup ([\![\varphi]\!]^{\mathfrak{M}} \cap \mathtt{pre}(\mathfrak{M}, A, Z)))$.

Exercise 6. Consider the concurrent game structure below with state space S and set of agents $\{\mathsf{Robot}_1, \mathsf{Robot}_2\}$.



1. Let $\sigma_{\mathsf{Robot}_1}$ be the positional strategy for Robot_1 such that $\sigma_{\mathsf{Robot}_1}(s_0) =$ wait, $\sigma_{\mathsf{Robot}_1}(s_1) =$ push, $\sigma_{\mathsf{Robot}_1}(s_2) =$ wait. Then, determine the following sets of maximal computations

$$\mathtt{Comp}(s_0, \sigma_{\mathsf{Robot}_1}), \quad \mathtt{Comp}(s_1, \sigma_{\mathsf{Robot}_1}), \quad \mathtt{Comp}(s_2, \sigma_{\mathsf{Robot}_1}).$$

   Use ω-regular expressions to define such sets of computations.

2. Let Robot_1 adopt the following memoryful strategy $\sigma^m_{\mathsf{Robot}_1}$. Below, "$\sigma(E) =$ a" for a regular expression E, indicates that the value of σ for every element of E is a. So, a is the action choosen by Robot_1 (below, we do not use anymore the notation with the joint action $\mathfrak{f}$)

$$\sigma^m_{\mathsf{Robot}_1}(\{s_0, s_1\}^+) = \mathsf{wait} \ , \quad \sigma^m_{\mathsf{Robot}_1}(\{s_0, s_1\}^* s_2 S^*) = \mathsf{push} \ .$$

   That is, the strategy prescribes waiting until the state s_2 is visited, if ever, and then pushing forever. Define a Büchi automaton $\mathcal{B}$ over the alphabet $\Sigma = \{s_1, s_2, s_3\}$ such that the language of ω-words accepted by $\mathcal{B}$ is the set of maximal computations $\mathtt{Comp}(s_1, \sigma^m_{\mathsf{Robot}_1})$ (omitting the joint actions between two successive states). For instance, $\mathcal{B}$ should accept the word $s_1 s_0 s_2^\omega$.