MPRI 2-27-1 Exam

Duration: 3 hours
Paper documents are allowed. The numbers in front of questions are indicative of hardness or duration.

1 Model-Theoretic Syntax

Exercise 1 (Propositional Dynamic Logic)**.** Recall that the syntax of PDL can be seen as follows. Let A be a countable set of atomic predicates. Then PDL formulæ can be defined by the abstract syntax:

$$\varphi ::= a \mid \top \mid \neg\varphi \mid \varphi \vee \varphi \mid \langle\pi\rangle\varphi \qquad \text{(node formulæ)}$$
$$\alpha ::= \varphi? \mid \downarrow \mid \uparrow \mid \rightarrow \mid \leftarrow \qquad \text{(atomic paths)}$$
$$\pi ::= \alpha \mid \pi + \pi \mid \pi;\pi \mid \pi^* \qquad \text{(path formulæ)}$$

where a ranges over A. Put differently, path formulæ are built as *rational languages* over an alphabet of atomic paths.

The *semantics* of a node formula on a tree structure $\mathfrak{M} = \langle W, \downarrow, \rightarrow, (P_a)_{a\in A}\rangle$ is a set of tree nodes $[\![\varphi]\!] = \{w \in W \mid \mathfrak{M}, w \models \varphi\}$, while the semantics of a path formula is a binary relation over W:

$$\begin{array}{rclrclrcl}
[\![a]\!] & \stackrel{\text{def}}{=} & \{w \in W \mid P_a(w)\} & [\![\downarrow]\!] & \stackrel{\text{def}}{=} & \downarrow & [\![\pi_1 + \pi_2]\!] & \stackrel{\text{def}}{=} & [\![\pi_1]\!] \cup [\![\pi_2]\!] \\
[\![\top]\!] & \stackrel{\text{def}}{=} & W & [\![\rightarrow]\!] & \stackrel{\text{def}}{=} & \rightarrow & [\![\pi_1;\pi_2]\!] & \stackrel{\text{def}}{=} & [\![\pi_1]\!] \mathbin{⨟} [\![\pi_2]\!] \\
[\![\neg\varphi]\!] & \stackrel{\text{def}}{=} & W \backslash [\![\varphi]\!] & [\![\uparrow]\!] & \stackrel{\text{def}}{=} & (\downarrow)^{-1} & [\![\pi^*]\!] & \stackrel{\text{def}}{=} & [\![\pi]\!]^\star \\
[\![\varphi_1 \vee \varphi_2]\!] & \stackrel{\text{def}}{=} & [\![\varphi_1]\!] \cup [\![\varphi_2]\!] & [\![\leftarrow]\!] & \stackrel{\text{def}}{=} & (\rightarrow)^{-1} & & & \\
[\![\langle\pi\rangle\varphi]\!] & \stackrel{\text{def}}{=} & [\![\pi]\!]^{-1}([\![\varphi]\!]) & [\![\varphi?]\!] & \stackrel{\text{def}}{=} & \{(w,w) \in W \times W \mid w \in [\![\varphi]\!]\}\ , & & &
\end{array}$$

where '⨟' denotes relational composition: for two binary relations R and R' over W, $R \mathbin{⨟} R' = \{(w, w'') \in W \times W \mid \exists w' \in W, (w, w') \in R \wedge (w', w'') \in R'\}$.

[2] 1. Prove the following equivalences:

$$\langle\pi_1;\pi_2\rangle\varphi \equiv \langle\pi_1\rangle\langle\pi_2\rangle\varphi$$
$$\langle\pi_1 + \pi_2\rangle\varphi \equiv (\langle\pi_1\rangle\varphi) \vee (\langle\pi_2\rangle\varphi)$$
$$\langle\pi^*\rangle\varphi \equiv \varphi \vee \langle\pi;\pi^*\rangle\varphi$$
$$\langle\varphi_1?\rangle\varphi_2 \equiv \varphi_1 \wedge \varphi_2\ .$$

[2] 2. Let us assume that we distinguish three disjoint subsets of labels: nonterminal labels in $N \subseteq A$, part-of-speech labels $\Theta \subseteq A$, and an open lexicon $L \subseteq A$. For example, in the tree in Figure 1 below, we have {S, NP, VP, PP} $\subseteq N$, {PRP, VBD, DT, NN, IN} $\subseteq \Theta$, and {*He, hurled, the, ball, into, basket*} $\subseteq L$.

Give a PDL formula ensuring that its models are labelled consistently with this style of constituent analysis.

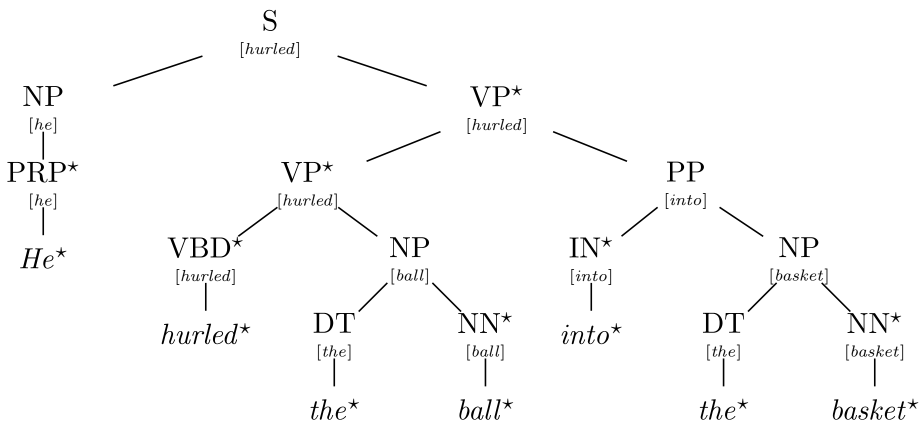

Figure 1: Example of a constituent tree. The head children are starred. The lexical heads of internal nodes are indicated inside brackets.

[3] 3. Recall from the lecture notes that a *head percolation* function $h\colon N \to \{l, r\} \times (N \uplus \Theta)^*$ provides for a given parent label $A \in N$ a pair $(d, X_1 \cdots X_n)$ consisting of a direction d and a list of potential head labels $X_1 \cdots X_n$. The intended semantics of such a function is to identify the head child of an A-labelled node $w \in W$. The direction indicates whether we should process the list of children of w left-to-right (l) or right-to-left (r). If X_1 appears among the children of w, then its leftmost (in case of l, and rightmost in case of r) occurrence is the head child of w. Otherwise, if X_2 appears among the children, then its leftmost (in case of l, and rightmost in case of r) occurrence is the head child of w... If none of $X_1, \ldots, X_n$ appears among the children of w, then its leftmost (in case of l, and rightmost in case of r) child is considered as its head. For instance, the function

$$\begin{aligned} h(\mathrm{S}) &= (r, \mathrm{TO\ IN\ VP\ S\ SBAR} \cdots) \\ h(\mathrm{VP}) &= (l, \mathrm{VBD\ VBN\ VBZ\ VB\ VBG\ VP} \cdots) \\ h(\mathrm{NP}) &= (r, \mathrm{NN\ NNP\ NNS\ NNPS\ JJR\ CD} \cdots) \\ h(\mathrm{PP}) &= (l, \mathrm{IN\ TO\ VBG\ VBN} \cdots) \end{aligned}$$

would result in the starred head children in Figure 1.

Given a head percolation function h, provide a PDL path formula π_h s.t. $(w, w') \in [\![\pi_h]\!]$ iff w is the parent of w' and w' is the head child of w. Your formula should also consider the case where w is labelled by a part-of-speech tag in Θ.

[1] 4. Provide a PDL path formula π_{lex} that holds between a node and its lexical head. The formula should allow to recover the lexical heads as indicated between brackets in Figure 1.

[1] 5. Consider $L(\varphi)$ the set of trees that satisfy the PDL node formula φ at their root. Justify why $\text{yield}(L(\varphi))$ is a context-free word language.

Exercise 2 (Relational PDL)**.** We extend the syntax of PDL to allow for 'relational paths'. To simplify matters, we shall only consider binary relational paths. Define $\varepsilon \stackrel{\text{def}}{=} \top?$. Then binary relational paths are defined by the following abstract syntax:

$$\beta ::= \alpha : \varepsilon \mid \varepsilon : \alpha \qquad \text{(atomic relations)}$$
$$\rho ::= \beta \mid \rho + \rho \mid \rho ; \rho \mid \rho^* \qquad \text{(relational paths)}$$

and adding the construction $\langle \rho \rangle$ to the syntax of node formulæ. Put differently, relational paths are constructed as *rational relations* over atomic paths. The semantics of a relational path on a tree structure $\mathfrak{M} = \langle W, \downarrow, \rightarrow, (P_a)_{a \in A} \rangle$ is a 4-ary relation in $W^2 \times W^2$, i.e. a binary relation on paths, defined by:

$$[\![\alpha : \varepsilon]\!] \stackrel{\text{def}}{=} \{(w, w', w'', w'') \mid (w, w') \in [\![\alpha]\!] \wedge w'' \in W\}$$
$$[\![\varepsilon : \alpha]\!] \stackrel{\text{def}}{=} \{(w, w, w', w'') \mid w \in W \wedge (w', w'') \in [\![\alpha]\!]\}$$

for atomic relations, while the sematics for '+', ';', and '*' are the obvious ones when seeing $[\![\rho]\!]$ as a binary relation on *pairs* of nodes. Finally,

$$[\![\langle \rho \rangle]\!] \stackrel{\text{def}}{=} \{w \in W \mid \exists w', w'' \in W, (w, w', w, w'') \in [\![\rho]\!]\} ,$$

meaning that we should find two paths starting from w and related by ρ.

[2] 1. Provide a relational path formula ρ_ℓ such that

$$[\![\rho_\ell]\!] = \{(w_1, w_1 w_2, w'_1, w'_1 w'_2) \mid |w_2| = |w'_2| \in \mathbb{N}^*\} .$$

Intuitively, ρ_ℓ relates two paths $(w_1, w_1 w_2)$ and $(w'_1, w'_1 w_2)$, both in $[\![\downarrow^*]\!]$, such that $w_1 w_2$ is as far below w_1 as $w'_1 w'_2$ is below w'_1.

[2] 2. Deduce that relational PDL allows to define some non-regular tree languages.

[4] 3. Recall from the classes that some natural languages, including Swiss German, exhibit cross-serial dependencies of the form $L_{\text{cross}} \stackrel{\text{def}}{=} \{a^n b^m c^n d^m \mid n, m > 1\}$. Provide a relational PDL node formula φ_{cross} such that $\text{yield}(L(\varphi_{\text{cross}})) = L_{\text{cross}}$.

[4] 4. Show that the satisfiability problem for relational PDL is undecidable.

Hint: Reduce from the Post Correspondence Problem.

2 Event semantics and adverbial modification

Exercise 3. One considers the three following signatures:

$$
(\Sigma_{\text{ABS}}) \quad
\begin{aligned}
\textsc{john} &: NP \\
\textsc{mary} &: NP \\
\textsc{kissed} &: NP \to NP \to V \\
\textsc{kissed}_{\circ} &: NP \to NP \to V_{\circ} \\
\textsc{not} &: (NP \to S_{\circ}) \to (NP \to S) \\
\textsc{e-clos} &: V \to S \\
\textsc{e-clos}_{\circ} &: V_{\circ} \to S_{\circ}
\end{aligned}
$$

$$
(\Sigma_{\text{S-FORM}}) \quad
\begin{aligned}
\boldsymbol{John} &: string \\
\boldsymbol{Mary} &: string \\
\boldsymbol{kissed} &: string \\
\boldsymbol{kiss} &: string \\
\boldsymbol{did} &: string \\
\boldsymbol{not} &: string
\end{aligned}
$$

$$
(\Sigma_{\text{L-FORM}}) \quad
\begin{aligned}
\mathbf{j}, \mathbf{m} &: \mathsf{e} \\
\mathbf{kiss}, \mathbf{past} &: \mathsf{v} \to \mathsf{t} \\
\mathbf{agent}, \mathbf{patient} &: \mathsf{v} \to \mathsf{e} \to \mathsf{t}
\end{aligned}
$$

In Σ_{ABS}, the atomic type NP stands for the syntactic category of *noun phrases*, the atomic types S and $S_{\circ}$ for the syntactic category of *sentences* (positive and negative), and the atomic type V and $V_{\circ}$, the syntactic categories of *"open" sentences* (positive and negative). The reason for distinguishing between the categories of positive and negative (open) sentences is merely syntactic. Without such a distinction, the surface realization of a negative expression such as:

$$\textsc{not}\,(\textsc{kissed}\ \textsc{mary})\ \textsc{john}$$

would be:

*John did not kissed Mary

Without this distinction, it would also be possible to iterate negation. This would allow the following ungrammatical sentences to be generated:

*John did not did not kissed Mary
*John did not did not did not kissed Mary
⋮

In $\Sigma_{\text{S-FORM}}$, as usual, *string* is defined to be $o \to o$ for some atomic type o. This allows concatenation $(+)$ to be defined as functional composition, and the empty word (ϵ) as the identity.

In $\Sigma_{\text{L-FORM}}$, the atomic type $\mathtt{e}$ stands for the semantic category of *entities*, the atomic types $\mathtt{t}$ for the semantic category of *truth values*, and the atomic types $\mathtt{v}$ for the semantic category of *events*.

One then defines two morphism ($\mathcal{L}_{\text{SYNT}} : \Sigma_{\text{ABS}} \to \Sigma_{\text{S-FORM}}$, and $\mathcal{L}_{\text{SEM}} : \Sigma_{\text{ABS}} \to \Sigma_{\text{L-FORM}}$) as follows:

$$(\mathcal{L}_{\text{SYNT}}) \qquad \begin{aligned} \textsc{john} &:= \boldsymbol{John} \\ \textsc{mary} &:= \boldsymbol{Mary} \\ \textsc{kissed} &:= \lambda xy.\, y + \boldsymbol{kissed} + x \\ \textsc{kissed}_\circ &:= \lambda xy.\, y + \boldsymbol{kiss} + x \\ \textsc{not} &:= \lambda fx.\, x + \boldsymbol{did} + \boldsymbol{not} + (f\,\epsilon) \\ \textsc{e-clos},\ \textsc{e-clos}_\circ &:= \lambda x.\, x \end{aligned}$$

$$(\mathcal{L}_{\text{SEM}}) \qquad \begin{aligned} \textsc{john} &:= \mathbf{j} \\ \textsc{mary} &:= \mathbf{m} \\ \textsc{kissed},\ \textsc{kissed}_\circ &:= \lambda xye.\, (\mathbf{kiss}\, e) \wedge (\mathbf{agent}\, e\, y) \wedge (\mathbf{patient}\, e\, x) \wedge (\mathbf{past}\, e) \\ \textsc{not} &:= \lambda px.\, \neg(p\, x) \\ \textsc{e-clos},\ \textsc{e-clos}_\circ &:= \lambda p.\, \exists e.\, p\, e \end{aligned}$$

These two morphisms are such that:

$$\mathcal{L}_{\text{SYNT}}(\textsc{e-clos}\,(\textsc{kissed mary john})) = \boldsymbol{John} + \boldsymbol{kissed} + \boldsymbol{Mary}$$

$$\mathcal{L}_{\text{SEM}}(\textsc{e-clos}\,(\textsc{kissed mary john})) = \exists e.\, (\mathbf{kiss}\, e) \wedge (\mathbf{agent}\, e\, \mathbf{j}) \wedge (\mathbf{patient}\, e\, \mathbf{m}) \wedge (\mathbf{past}\, e)$$

The last term may be paraphrased as follows: *there is an event e such that: e is a kissing event; the agent of this kissing event is John; the patient of this kissing event is Mary; and this event e happened in the past.*

[1] 1. Give a term t such that

$$\mathcal{L}_{\text{SYNT}}(t) = \boldsymbol{John} + \boldsymbol{did} + \boldsymbol{not} + \boldsymbol{kiss} + \boldsymbol{Mary},$$

then compute $\mathcal{L}_{\text{SEM}}(t)$.

Solution:

$t = \textsc{not}\,(\lambda x.\, \textsc{e-clos}_\circ\,(\textsc{kissed}_\circ\, \textsc{mary}\, x))\, \textsc{john}$.

$\mathcal{L}_{\text{SEM}}(t) = \neg(\exists e.\, (\mathbf{kiss}\, e) \wedge (\mathbf{agent}\, e\, \mathbf{j}) \wedge (\mathbf{patient}\, e\, \mathbf{m}) \wedge (\mathbf{past}\, e))$

[2] 2. Suppose that one modifies Σ_{ABS} and $\mathcal{L}_{\text{SEM}}$ as follows:

$$(\Sigma_{\text{ABS}}) \qquad \begin{aligned} &\vdots \\ \textsc{not} : (NP \to V_\circ) &\to (NP \to V) \\ &\vdots \end{aligned}$$

$$(\mathcal{L}_{\text{SEM}}) \quad \begin{aligned} &\vdots \\ \textsc{not} &:= \lambda pxe.\, \neg(p\,x\,e) \\ &\vdots \end{aligned}$$

What would be wrong?

Solution:

Let $t = \textsc{e-clos}\,(\textsc{not}\,(\lambda x.\, \textsc{kissed}_{\circ}\,\textsc{mary}\,x)\,\textsc{john})$. We would have

$$\mathcal{L}_{\text{SYNT}}(t) = \textbf{\textit{John}} + \textbf{\textit{did}} + \textbf{\textit{not}} + \textbf{\textit{kiss}} + \textbf{\textit{Mary}},$$

and

$$\mathcal{L}_{\text{SEM}}(t) = \exists e.\, \neg((\mathbf{kiss}\,e) \wedge (\mathbf{agent}\,e\,\mathbf{j}) \wedge (\mathbf{patient}\,e\,\mathbf{m}) \wedge (\mathbf{past}\,e)).$$

This last term does not assert that there is no *past kissing event between John and Mary*, but that there is an event which is not a *past kissing event between John and Mary*. Consequently, in a situation where John kissed both Mary and Sue, we would consider "John did not kissed Mary" to be true.

Exercise 4. One extends Σ_{ABS}, $\Sigma_{\text{S-FORM}}$, $\Sigma_{\text{L-FORM}}$, $\mathcal{L}_{\text{SYNT}}$, and $\mathcal{L}_{\text{SEM}}$, respectively, as follows:

$$(\Sigma_{\text{ABS}}) \quad \begin{aligned} \textsc{hour} &: N_u \\ \textsc{one} &: N_u \to NP_\tau \\ \textsc{for} &: NP_\tau \to ((V \to V) \to S) \to S \\ \textsc{for}_\circ &: NP_\tau \to ((V_\circ \to V_\circ) \to S) \to S \\ \textsc{for}_{\circ\circ} &: NP_\tau \to ((V_\circ \to V_\circ) \to S_\circ) \to S_\circ \end{aligned}$$

where N_u is the syntactic category of *nouns that name units of measurement*, and NP_τ is the syntactic the category of *noun phrases that denote time intervals*;

$$(\Sigma_{\text{S-FORM}}) \quad \begin{aligned} \textbf{\textit{hour}} &: \textit{string} \\ \textbf{\textit{one}} &: \textit{string} \\ \textbf{\textit{for}} &: \textit{string} \end{aligned}$$

$$(\Sigma_{\text{L-FORM}}) \quad \begin{aligned} \mathbf{hour} &: \mathsf{i} \to \mathsf{n} \to \mathsf{t} \\ 1 &: \mathsf{n} \\ \mathbf{duration} &: \mathsf{v} \to \mathsf{i} \to \mathsf{t} \end{aligned}$$

where i and n stand for the semantic categories of time intervals and scalar quantities, respectively.

$$(\mathcal{L}_{\text{SYNT}}) \qquad \begin{aligned} \textsc{hour} &:= \boldsymbol{hour} \\ \textsc{one} &:= \lambda x.\, \boldsymbol{one} + x \\ \textsc{for},\ \textsc{for}_{\circ},\ \textsc{for}_{\circ\circ} &:= \lambda x f.\, f\,(\lambda x.\, y + \boldsymbol{for} + x) \end{aligned}$$

$$(\mathcal{L}_{\text{SEM}}) \qquad \begin{aligned} \textsc{hour} &:= \lambda xy.\, \mathbf{hour}\, x\, y \\ \textsc{one} &:= \lambda pt.\, p\, t\, 1 \\ \textsc{for},\ \textsc{for}_{\circ},\ \textsc{for}_{\circ\circ} &:= \lambda pq.\, \exists t.\, (p\, t) \wedge (q\,(\lambda pe.\, (p\, e) \wedge (\mathbf{duration}\, e\, t))) \end{aligned}$$

[4] 1. Give two different terms, say t_0 and t_1, such that:

$$\mathcal{L}_{\text{SYNT}}(t_0) = \mathcal{L}_{\text{SYNT}}(t_1) = \boldsymbol{John} + \boldsymbol{did} + \boldsymbol{not} + \boldsymbol{kissed} + \boldsymbol{Mary} + \boldsymbol{for} + \boldsymbol{one} + \boldsymbol{hour}$$

Solution:

$$\begin{aligned} t_0 = {} & \textsc{for}_{\circ}\ (\textsc{one}\ \textsc{hour}) \\ & \quad (\lambda q.\, \textsc{not}\,(\lambda x.\, \textsc{e-clos}_{\circ}\,(q\,(\textsc{kissed}_{\circ}\ \textsc{mary}\ x)))\ \textsc{john}) \end{aligned}$$

$$\begin{aligned} t_1 = {} & \textsc{not}\ (\lambda x.\, \textsc{for}_{\circ\circ}\ (\textsc{one}\ \textsc{hour}) \\ & \qquad\qquad (\lambda q.\, \textsc{e-clos}_{\circ}\,(q\,(\textsc{kissed}_{\circ}\ \textsc{mary}\ x)))) \\ & \quad \textsc{john} \end{aligned}$$

[2] 2. Compute $\mathcal{L}_{\text{SEM}}(t_0)$ and $\mathcal{L}_{\text{SEM}}(t_1)$.

Solution:

$\mathcal{L}_{\text{SEM}}(t_0) =$
$\exists t.\, (\mathbf{hour}\, t\, 1) \wedge \neg(\exists e.\, (\mathbf{kiss}\, e) \wedge (\mathbf{agent}\, e\, \mathbf{j}) \wedge (\mathbf{patient}\, e\, \mathbf{m}) \wedge (\mathbf{past}\, e) \wedge (\mathbf{duration}\, e\, t))$

$\mathcal{L}_{\text{SEM}}(t_1) =$
$\neg(\exists t.\, (\mathbf{hour}\, t\, 1) \wedge (\exists e.\, (\mathbf{kiss}\, e) \wedge (\mathbf{agent}\, e\, \mathbf{j}) \wedge (\mathbf{patient}\, e\, \mathbf{m}) \wedge (\mathbf{past}\, e) \wedge (\mathbf{duration}\, e\, t)))$

[1] 3. Explain the difference between $\mathcal{L}_{\text{SEM}}(t_0)$ and $\mathcal{L}_{\text{SEM}}(t_1)$.

Solution: t_0 and t_1 correspond respectively to the following interpretations:

1. *For one hour, it was not the case that John kissed Mary.*
2. *It was not the case that John kissed Mary for one hour.*