# Encoding Causality via Modal Formulae

Georgiana Caltais[1] and Mohammad Reza Mousavi[2]

[1] Department for Computer and Information Science, University of Konstanz
[2] Department of Informatics, University of Leicester

**Abstract.** This work introduces an encoding of causality for labelled transition systems and Hennessy Milner logic, in terms of modal formulae with data. The approach paves the way to the automatic identification of causalities using the mCRL2 model checker.

**Introduction.** Determining and computing causalities is a frequently addressed issue in the philosophy of science and engineering. A notion of causality that is frequently used in relation to technical systems relies on counterfactual reasoning [8]. In short, the counterfactual argument defines when an event is considered a cause for some effect, in the following way: a) whenever the event presumed to be a cause occurs, the effect occurs as well, and b) when the presumed cause does not occur, the effect will not occur either. The seminal papers [4, 5] describes an event model and a notion of actual causation encompassing the counterfactual argument. Most relevant for our work are the contributions in [7, 1]. The results in [7] provide an interpretation of the results in [4] in the context of transition systems and trace models for concurrent system computations. In [1] we adopt the aforementioned trace-based interpretation to the context of labelled transitions systems (LTS's) and Hennesy Milner logic (HML) [6] and devise a series of preliminary results on compositionality of causality.

*The objective of this paper* is to provide an encoding of causality as in [1] in terms of modal formulae with data [3], thus paving the way to the identification of causalities in an algorithmic fashion using the mCRL2 model-checker [2].

**Preliminaries.** Next, we provide a brief overview of LTS's and their computations, and HML. A *labeled transition system* (LTS) is a triple $T = (\$, s_0, A, \rightarrow)$, where $\$$ is the set of states, $s_0 \in \$$ is the initial state, $A$ is the action alphabet and $\rightarrow \subseteq \$ \times A \times \$$ is the transition relation. Let $A^*$ be the set of words over $A$, and let $\varepsilon$ be the empty word. We write $\twoheadrightarrow\ \subseteq \$ \times A^* \times \$$, to denote the extension of $\rightarrow$ to words, defined recursively in the expected way: $s \overset{a}{\twoheadrightarrow} s'$ iff $s \xrightarrow{a} s'$, $s \overset{\varepsilon}{\twoheadrightarrow} s$, $s \overset{aw}{\twoheadrightarrow} s'$ iff $s \overset{a}{\twoheadrightarrow} s'$ and $s \overset{w}{\twoheadrightarrow} s'$, for $a \in A$ and $w \in A^*$.

Let $\mathcal{D}$, $\mathcal{D}_i$ range over possibly infinite lists of words in $A^*$. We say that two such lists are *size-compatible* if they are finite lists of the same length, or if they are all infinite lists.

Let $\pi = (s_0, l_0, \mathcal{D}_0), \ldots (s_n, l_n, \mathcal{D}_n), s_{n+1} \in (\$ \times A \times [A^*])^* \times \$$. Assume that $\mathcal{D}_0, \ldots, \mathcal{D}_n$ are size-compatible. We write $traces(\pi)$ to denote the pairwise extensions of $l_0 \ldots l_n$ with words "at the same level" in $\mathcal{D}_0, \ldots, \mathcal{D}_n$. For instance, if $\pi = (s_0, l_0, [w_1^0, w_2^0]), (s_1, l_1, [w_1^1, w_2^1]), s_2$, then $traces(\pi) = \{l_0 w_1^0 l_1 w_1^1,\ l_0 w_2^0 l_1 w_2^1\}$. $\pi$ is a *computation* of $T$ whenever the following hold: (i) $s_0 \xrightarrow{l_0} s_1 \ldots \xrightarrow{l_n} s_{n+1}$,

(ii) $\mathcal{D}_0, \ldots, \mathcal{D}_n$ are size-compatible, and (iii) for all $w \in \mathit{traces}(\pi)$ there exists $s \in \$$ s.t. $s_0 \overset{w}{\twoheadrightarrow} s$. $\mathit{sub}(\pi)$ stands for the set of all computations $\pi' = (s_0, l'_0, \mathcal{D}'_0), \ldots, (s_m, l'_m, \mathcal{D}'_m), s'_{m+1}$ s.t. $l'_0 \ldots l'_m$ is a sub-word of $l_0 \ldots l_n$.

We consider formulae in *Hennessy-Milner logic* (HML) [6] given by the following grammar:

$$\phi, \psi ::= \top \mid \langle a \rangle \phi \mid [a]\phi \mid \neg \phi \mid \phi \wedge \psi \mid \phi \vee \psi \qquad (a \in A). \tag{1}$$

We say that an HML formula $\phi$ as above is *built over $A$*. The associated satisfaction relation $\models$ is defined in the standard way, over states $s \in \$$ and HML formulae. Intuitively, $s \models \langle a \rangle \phi$ states that $s$ can execute $a$ and reach a state that satisfies $\phi$ afterwards. Orthogonally, $s \models [a]\phi$ refers to the fact that no matter what state is reached from $s$ by executing $a$, the reached state satisfies $\phi$. $\top$ is the formula that holds in any state, whereas $\wedge, \vee$ and $\neg$ stand for conjunction, disjunction and negation.

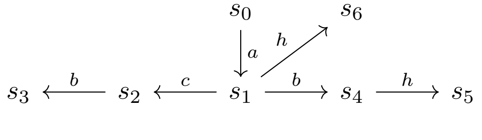
**Defining causality.** Our notion of causality is an adoption of the "actual causation" proposed in [4], previously adapted to the setting of concurrent systems in [7]. Consider an LTS $T = (\$, s_0, A, \rightarrow)$ and a "hazard" HML formula $\phi$. A causal analysis of $T$ w.r.t. $\phi$ is justified under the assumption that $\phi$ does not hold in all states of $T$, *i.e.*, $T$ can display correct behaviour as well (item 2 below). A computation $\pi = (s_0, l_0, \mathcal{D}_0), \ldots, (s_n, l_n, \mathcal{D}_n), s_{n+1}$ is a *causal trace* if, intuitively:

- the execution of $l_0 \ldots l_n$ leads to a state satisfying the hazard (item 1 below)
- the occurrence of the actions $l_0, \ldots, l_n$ along a trace $\chi'$ in $T$ guarantees that executing $\chi'$ leads to the hazard (item 3), as long as $\chi'$ does not encode elements in $\mathcal{D}_0, \ldots, \mathcal{D}_n$ which are causal by their non-occurrence (item 4). Non-occurrence is useful to explain situations in which the hazard holds if certain words in $\mathcal{D}_0, \ldots, \mathcal{D}_n$ are not executed, whereas executing these words removes the hazard
- $\pi$ is the "shortest" computation satisfying the above properties (item 5)

Formally, *causal traces* in $T$ w.r.t. $\phi$, denoted by $\mathit{Causes}(\phi, T)$ , is the set of all computations $\pi = (s_0, l_0, \mathcal{D}_0), \ldots, (s_n, l_n, \mathcal{D}_n), s_{n+1}$ s.t. :

1. $s_0 \xrightarrow{l_0} \ldots s_n \xrightarrow{l_n} s_{n+1} \wedge s_{n+1} \models \phi$ *(Positive causality)*
2. $\exists \chi \in A^*, s' \in \$ : s_0 \overset{\chi}{\twoheadrightarrow} s' \wedge s' \models \neg \phi$ *(Counter-factual)*
3. $\forall \chi' = l_0 \chi_0 \ldots l_n \chi_n \in \{l_0 \ldots l_n\} \cup (A^* \setminus \mathit{traces}(\pi)),\ s' \in \$$ :
   $s_0 \overset{\chi'}{\twoheadrightarrow} s' \Rightarrow s' \models \phi$ *(Occurrence)*
4. $\forall \chi' \in \mathit{traces}(\pi) \setminus \{l_0 \ldots l_n\},\ s' \in \$ : s_0 \overset{\chi'}{\twoheadrightarrow} s' \Rightarrow s' \models \neg \phi$ *(Non-occurrence)*
5. $\forall \pi' \in \mathit{sub}(\pi) : \pi'$ *does not satisfy items 1. – 4. above (Minimality)*

Consider, for an example, the following LTS and the HML formula $\phi = \langle h \rangle \top$:



(2)

Item 1 suggests that action $a$ should be a cause for the hazard $\phi$. Item 2 indicates that $\phi$ does not hold trivially everywhere as, for instance, $s_0 \overset{acb}{\twoheadrightarrow} s_3$ and $s_3 \not\models \phi$. Item 4 states that $(s_0, a, [\varepsilon]), s_1$ is not a cause for $\phi$ because extending $a$ with $cb$, for instance, violates $\phi$ and thereby violates item 3. However, $(s_0, a, [h, c, cb, bh]), s_1$ is a good candidate as all possible extensions of $a$ with anything but $h$, $c$, $cb$ or $bh$ also keep the hazard, and thus satisfies items 3 and 4. Item 5 states that $(s_0, a, [\varepsilon, c]), (s_1, b, [h, \varepsilon]), s_4$ is not a cause as it is not minimal. This is because its sub-computation $(s_0, a, [h, c, cb, bh])$, $s_1$ satisfies items 1–4 as previously discussed.

**Causality as modal formulae with data.** In this section we introduce an attempt of encoding causality via modal formulae with data. This paves the way to the automatic identification of causes in mCRL2.

We proceed by first introducing modal formulae with data as in [3], used in order to model various real world phenomena. For space considerations, we only provide the fragment relevant for our work:

$$\begin{aligned}
R &::= a \mid \varepsilon \mid R \cdot R \mid R + R \mid R^* \mid R^+ \qquad (a \in A)\\
\phi &::= \mathit{true} \mid \mathit{false} \mid \neg\phi \mid \phi \vee \phi \mid \phi \wedge \phi \mid \forall d : D.\phi \mid \exists d : D.\phi \mid \langle R\rangle\phi \mid [R]\phi\\
&\qquad \mu X(d_1 : D_1 := t_1, \ldots, d_n : D_n := t_n).\phi \mid\\
&\qquad \mathcal{V} X(d_1 : D_1 := t_1, \ldots, d_n : D_n := t_n).\phi \mid X(t_1, \ldots, t_n)
\end{aligned} \tag{3}$$

Formulae $R$ are defined as regular expressions in the standard way. Formulae $\langle R\rangle$ and, respectively, $[R]$ extend the diamond $\langle -\rangle$ and, respectively, box $[-]$ modalities in (1) to regular expressions. Existential and universal quantifiers for ranging over data domains are also introduced. $\mu$ and, respectively, $\mathcal{V}$ stand for the minimal and, respectively, the maximal fixed point equations.

Assume a computation $\pi = (s_0, a_0, \mathcal{D}_0), \ldots, (s_n, a_n, \mathcal{D}_n), s_{n+1}$. We write $l$ for the list $a_0 : \ldots : a_n : []$, and $\mathcal{LD}$ for the list of lists $\mathcal{D}_0 : \ldots : \mathcal{D}_n$. In order to check whether $\pi$ is a cause w.r.t. a HML formula $\phi$, we propose a straightforward encoding the corresponding items $1 - 5$ in terms of modal formulae with data as below.

We write $A^*$ for the "type" of words of actions in $A$ (*e.g.*, $l : A^*$), $[A^*]$ for the "type" of lists of words of actions in $A$ (*e.g.*, $\mathcal{D}_0 : [A^*]$), $[[A^*]]$ for the "type" of lists of lists of words of actions in $A$ (*e.g.*, $\mathcal{LD} : [[A^*]]$), $[l]\phi$ to denote the formula $[a_0 \cdot \ldots \cdot a_n]\phi$ (symmetrically for the diamond modality $\langle -\rangle$). We use the notations $\overset{\mu}{=}$ and, respectively, $\overset{\mathcal{V}}{=}$ in order to represent minimal and, respectively, maximal fixed point equations. The encodings are:

$$PC(l : A^*, \mathcal{LD} : [[A^*]]) \overset{\mu}{=} \langle l\rangle\phi \qquad \text{(encoding Positive Causality)}$$

$$C(l : A^*, \mathcal{LD} : [[A^*]]) \overset{\mu}{=} \exists l' : A^*.\langle l'\rangle\neg\phi \qquad \text{(encoding Counter-factual)}$$

$$\begin{array}{l}
CON(l : A^*, \mathcal{LD} : [[A^*]], n : \mathbb{N}, k : \mathbb{N}, j : \mathbb{N}) \overset{\nu}{=} \\
\forall l_0 : A^*.\ \dots\ .\forall l_n : A^*.\exists l' : A^*.\ (j == k) \vee \\
((j \neq k) \wedge (l' == zip(l, l_0 : \dots : l_n)) \wedge (l' {\neq} zip(l, row(\mathcal{LD}, j)) \wedge [l']\phi \wedge \\
CON(l, \mathcal{LD}, n, k, j+1))) \vee \qquad (\clubsuit) \\
((j \neq k) \wedge (l' {==} zip(l, l_0 : \dots : l_n)) \wedge (l' {==} zip(l, row(\mathcal{LD}, j)) \wedge [l']\neg\phi \wedge \\
CON(l, \mathcal{LD}, n, k, j+1))) \qquad (\spadesuit)
\end{array}$$

(encoding Causality of (non-)occurrence)

where $n+1$ is the size of $l$, and $k+1$ is the length of the size-compatible lists $\mathcal{D}_i$ in $\mathcal{LD}$. Additionally, $j$ is an index used for iterating through the words at location $j$ in each of the lists $\mathcal{D}_i$. The words at location $j$ in all $\mathcal{D}_i$'s are given by the "row" $j$ in $\mathcal{LD}$: $row(\mathcal{LD}, j)$. Variables $l, n, k$ and $j$ are examples of data tokens for formulae as in (3). Furthermore, $zip(l, l_0 : \dots : l_n)$ denotes the pairwise extension of $l$ with $l_0 : \dots : l_n$ as expected: $a_0 : l_0 : \dots : a_n : l_n$. Intuitively, $zip(l, l_0 : \dots : l_n)$ corresponds to an element in $traces(\pi)$. Hence, the disjunct ($\clubsuit$) encodes *causality of occurrence*, whereas the disjunct ($\spadesuit$) encodes *causality of non-occurrence.*

$$\begin{array}{l}
M(l : A^*, \mathcal{LD} : [[A^*]]) \overset{\mu}{=} \\
\qquad \exists l' : A^*.\exists \mathcal{LD}' : [[A^*]].\ (subtrace(l', l) == true) \wedge \\
\qquad (\mid l' \mid +1 ==\mid \mathcal{LD}' \mid) \wedge szCompatible(\mathcal{LD}') \wedge \\
\qquad PC(l', \mathcal{LD}') \wedge C(l', \mathcal{LD}') \wedge CON(l', \mathcal{LD}', \mid l' \mid, \mid \mathcal{LD}'[0] \mid, 0)
\end{array}$$

(encoding Minimality)

In the formula above, we write $subtrace(l', l) == true$ whenever $l' \in sub(l)$. Moreover, $szCompatible(\mathcal{LD}') == true$ whenever the elements of $\mathcal{LD}'$ are size-compatible lists. A non-empty solution w.r.t. $M(l, \mathcal{LD})$ denotes that $\pi$ violates the minimality condition.

Let $\pi = (s_0, a, [h, c, cb, bh]), s_1$ be the causal computation of the LTS in (2), w.r.t. the HML formula $\langle h \rangle \top$. We fix $l = a$ and $\mathcal{LD} = [[h], [c], [cb], [bh]]$. It is straightforward to see that $s_0$ satisfies the formula in (encoding Positive Causality). It follows immediately that (encoding Counter-factual) holds in $s_0$ when $l' = acb$, for instance. Moreover, $s_0$ satisfies ($\spadesuit$) for all $l'$ ranging over $\{ah, ac, acb, abh\}$, as $s_0 \overset{l'}{\twoheadrightarrow} s_i \Rightarrow s_i \not\models \langle h \rangle \top$. Symmetrically, $s_0$ satisfies ($\clubsuit$) for all remaining transitions $l'$ ranging over $\{a, ab\}$ as $s_0 \overset{l'}{\twoheadrightarrow} s_i \Rightarrow s_i \models \langle h \rangle \top$. Similarly, it can be shown that (encoding Minimality) is not satisfied by any state in (2). Hence, the proposed modal formulae confirm $\pi$ as being causal.

**Conclusions.** We provide an encoding of the causality for LTS's and HML in [1] in terms of modal formulae with data. This is the first step towards the implementation of an algorithm for computing such causalities. As future work we consider implementing the corresponding encodings in mCRL2 [2]. While the trace component $l = l_0, \dots, l_n$ in the proposed definition of causality can be easily identified as a counterexamples violating the specification, one of the biggest challenges remains the automatic identification of the words in $\mathcal{D}_i$ causal by their non-occurrence as in item 4. Corresponding case studies and comparison with other approaches (*e.g.*, [7]) will be considered as well.

## References

1. G. Caltais, S. Leue, and M. R. Mousavi. (De-)composing causality in labeled transition systems. In G. Gößler and O. Sokolsky, editors, *Proceedings First Workshop on Causal Reasoning for Embedded and safety-critical Systems Technologies, CREST@ETAPS 2016, Eindhoven, The Netherlands, 8th April 2016.*, volume 224 of *EPTCS*, pages 10–24, 2016.
2. S. Cranen, J. F. Groote, J. J. A. Keiren, F. P. M. Stappers, E. P. de Vink, W. Wesselink, and T. A. C. Willemse. An overview of the mCRL2 toolset and its recent advances. In N. Piterman and S. A. Smolka, editors, *Tools and Algorithms for the Construction and Analysis of Systems - 19th International Conference, TACAS 2013, Held as Part of the European Joint Conferences on Theory and Practice of Software, ETAPS 2013, Rome, Italy, March 16-24, 2013. Proceedings*, volume 7795 of *Lecture Notes in Computer Science*, pages 199–213. Springer, 2013.
3. J. F. Groote and M. R. Mousavi. *Modeling and Analysis of Communicating Systems.* MIT Press, 2014.
4. J. Halpern and J. Pearl. Causes and explanations: A structural-model approach. Part I: Causes. *The British Journal for the Philosophy of Science*, 2005.
5. J. Y. Halpern. A modification of the Halpern-Pearl definition of causality. In Q. Yang and M. Wooldridge, editors, *Proceedings of the Twenty-Fourth International Joint Conference on Artificial Intelligence, IJCAI 2015, Buenos Aires, Argentina, July 25-31, 2015*, pages 3022–3033. AAAI Press, 2015.
6. M. Hennessy and R. Milner. On observing nondeterminism and concurrency. In J. W. de Bakker and J. van Leeuwen, editors, *Automata, Languages and Programming, 7th Colloquium, Noordweijkerhout, The Netherland, July 14-18, 1980, Proceedings*, volume 85 of *Lecture Notes in Computer Science*, pages 299–309. Springer, 1980.
7. F. Leitner-Fischer and S. Leue. Causality checking for complex system models. In R. Giacobazzi, J. Berdine, and I. Mastroeni, editors, *Verification, Model Checking, and Abstract Interpretation, 14th International Conference, VMCAI 2013, Rome, Italy, January 20-22, 2013. Proceedings*, volume 7737 of *Lecture Notes in Computer Science*, pages 248–267. Springer, 2013.
8. D. Lewis. *Counterfactuals.* Blackwell Publishers, 1973.