Substructural modal logic for optimal resource allocation

Gabrielle Anderson
University College London, UK
gabrielle.anderson@ucl.ac.uk

David Pym
University College London, UK
d.pym@ucl.ac.uk

We introduce a substructural modal logic for reasoning about (optimal) resource allocation in models of distributed systems. The underlying logic is a variant of the modal logic of bunched implications, and based on the same resource semantics, which is itself closely related to concurrent separation logic. By considering notions of cost, strategy, and utility, we are able to formulate characterizations of Pareto optimality, best responses, and Nash equilibrium within resource semantics.

1 Introduction

Mathematical modelling and simulation modelling are fundamental tools of engineering, science, and social sciences such as economics, and provide decision-support tools in management. The components of distributed systems (as described, e.g., in [9]) are typically modelled using various algebraic structures for the structural components — location, resource, and process — and probability distributions to represent stochastic interactions with the environment. A key aspect of modelling distributed systems is resource allocation. For example, when many processes execute concurrently, they compete for resources.

A common desire of system designers, managers, and users is to determine, if possible, the optimal allocation of resources required in order to solve a specific problem or deliver a specific service. The notion of optimality of resource allocation is a central topic in economics, where game theory plays a significant role. For all elementary notions from economics required for this short paper, including ideas from utility theory and game theory, a suitable source is [20].

Building on a mathematical systems and security modelling framework — described in, for example, [8, 6, 7], which builds on ideas in [2] and which has been widely deployed (e.g., [15, 1, 5, 3, 4]) — we sketch the development of a systems modelling framework that provides a theory of (optimal) resource allocation.

The key systems components of our resource semantics-based framework (which in turn builds on BI and its resource semantics [17, 18, 10, 8, 6]) are the following: *environment* (within which the system resides), *locations* (the architecture of the system), *resources* (that are manipulated — e.g., consumed, created, moved — by the system), and *processes* (that operate the system and deliver services). We integrate these components into an algebra of locations, resources, and processes that is defined by an operational semantics [8, 7] with a judgement of the form $L,R,E \xrightarrow{a} L',R',E'$ in which the process E evolves by action a, using resources R at locations L, to become the process E', able to evolve further using the resources R' at locations L'. A key component of this operational semantics is a (partial) *modification function*,

J. Gutierrez, F. Mogavero, A. Murano, and M. Wooldridge (Eds.):
3rd International Workshop on Strategic Reasoning 2015 (SR15)

$\mu : \mathit{Actions} \times \mathit{Resources} \times \mathit{Locations} \rightharpoonup \mathit{Resources} \times \mathit{Locations}$, that specifies the effects of actions on resources and locations.

Properties of systems, including optimality properties, can be expressed logically. Specifically, we make use of a substructural modal logic [8, 7] that is naturally associated with the process algebra above in the Hennessy–Milner sense [12, 16, 8] — that is, it is defined by a (truth-functional) satisfaction relation of the form $L, R, E \models \phi$, for logical formulae ϕ — with transitions between worlds defined by the operational semantics.

For the purposes of this paper, however, we make two simplifications. First, we elide locations, which can be coded in terms of resources if necessary. Second, we neglect the structure of processes, using modification functions to describe the effects of actions on processes. Thus we are able to define a logic with a satisfaction relation between resource states R and formulae ϕ (i.e., $R \models \phi$) in which the meaning of formulae involving action modalities, such as $\langle a \rangle \phi$, is given by transitions as specified by $\mu(a, R)$.

To this logic we add, in Section 4, a simple account of utility, building on simple notions of strategy and cost that we introduce in Section 3. Then, in Section 4, we consider a range of examples about resource allocation and optimality, including Pareto optimality, best responses, and Nash equilibrium. We begin by introducing, in Section 2, resource semantics.

2 Resource semantics and modal logic for systems modelling

We present our resource model and semantics, along with its key technical properties. We define resources, actions, and an operational semantics for resources. We define our notion of bisimulation, and note that resource composition forms a congruence with respect to the bisimulation relation. We sketch a modal logic, and describe how it can be used for systems modelling.

First, we introduce our notion of resource, following [8, 7].

Definition 1 (Resource monoid)**.** *A resource monoid is a structure* $\boldsymbol{R} = (\boldsymbol{R}, \circ, e)$ *with carrier set* $\boldsymbol{R}$*, commutative partial binary operation* $\circ : \boldsymbol{R} \times \boldsymbol{R} \rightharpoonup \boldsymbol{R}$*, and unit* $e \in \boldsymbol{R}$*.* □

We assume a commutative monoid, **Act**, of actions, freely generated from a set of atomic actions. The actions correspond to the events of the system.

Definition 2 (Actions)**.** *Let **Act** be the free commutative monoid formed by combinations of atomic actions, with operation* $\cdot$ *and unit* 1*. Let* ab *denote* $a \cdot b$*.* □

We set up a function that describes how actions transform resources.

Definition 3 (Modification function)**.** *A modification function is a partial function* $\mu : \boldsymbol{Act} \times \mathbf{R} \rightharpoonup \mathbf{R}$ *such that, for all resources* $R, S \in \mathbf{R}$ *and actions* $a, b, c \in \boldsymbol{Act}$*:*

- *If* $\mu(a, R)$*,* $\mu(b, S)$*, and* $R \circ S$ *are all defined, then* $\mu(a, R) \circ \mu(b, S)$ *and* $\mu(ab, R \circ S)$ *are both defined, and* $\mu(ab, R \circ S) = \mu(a, R) \circ \mu(b, S)$ *holds;*
- *If* $R \circ S$ *and* $\mu(c, R \circ S)$ *are defined, then there exist* $a, b \in \boldsymbol{Act}$ *such that* $c = ab$*, and* $\mu(a, R)$ *and* $\mu(b, S)$ *are both defined;*
- $\mu(1, R) = R$. □

If $\mu(a, R)$ is defined, then we say that action a is defined on resource R. We can use the partiality of the resource monoid, along with the modification function, to model straightforwardly key examples in systems modelling [8, 7], such as the following:

Example 4 (Semaphores). *Suppose a resource monoid $(\{s,e\},\circ,e)$, where $s \circ s$ is undefined. Let a be an action. We define a modification function μ such that $\mu(a,s) = s$. Note that μ is undefined for any values that are neither specified explicitly nor required by properties of Definition 3. We then have that, for all resources $R \in \boldsymbol{R}$, $\mu(aa,R)$ is not defined. The resource s acts like a semaphore, in that only one access action a can be performed at any given time.* □

From a resource monoid, action monoid, and modification function, we derive a transition relation. If the modification function is defined for an action a on a resource R, and $\mu(a,R) = S$, then we say that there exists a transition $R \xrightarrow{a} S$, and that S is a successor of R. A notion of *bisimulation* between resources is defined in the standard way.

Definition 5 (Bisimulation). *A bisimulation is a relation $\mathscr{R}$ such that, for all $R \mathscr{R} S$, then, for all actions $a \in \boldsymbol{Act}$,*

- *if $R \xrightarrow{a} R'$, then there exists S' such that $S \xrightarrow{a} S'$ and $R' \mathscr{R} S'$, and*
- *if $S \xrightarrow{a} S'$, then there exists R' such that $R \xrightarrow{a} R'$ and $R' \mathscr{R} S'$.* □

Let $\sim \subseteq \mathbf{R} \times \mathbf{R}$ be the union of all bisimulations. The union of any two bisimulations is also a bisimulation. Hence $\sim$ is well defined, and a bisimulation. In this simple setting, bisimulation equivalence is the same as trace equivalence, but that is not generally true in the more general location-resource-process framework, of which this is an example.

We can now obtain a key property: that bisimulation is a congruence; that is, an equivalence relation that is respected by the composition operator.

Lemma 6 (Bisimulation congruence). *The relation $\sim$ on resources is a congruence for the operation $\circ$: if $R_1 \sim S_1$, $R_2 \sim S_2$, and $R_1 \circ R_2$ and $S_1 \circ S_2$ are defined, then $R_1 \circ R_2 \sim S_1 \circ S_2$.*

Proof. A straightforward argument, similar to many others. □

We can use a substructural modal logic of resources to reason about our models (of distributed systems). The logic freely combines classical propositional logic with action modalities, in the style of Hennessy–Milner logic [12, 8] or dynamic logic [11], and with BI's multiplicatives [17]. Worlds are given by the resources R of a resource monoid. The classical connectives are defined with respect to a fixed world in the usual way: $R \models \bot$ never, $R \models \phi_1 \vee \phi_2$ iff $R \models \phi_1$ or $R \models \phi_2$, and $R \models \neg\phi$ iff $R \not\models \phi$, with satisfying truth $\top = \neg\bot$ and conjunction satisfying $\phi_1 \wedge \phi_2$ = $\neg(\neg\phi_1 \vee \neg\phi_2)$, so that, in its resulting semantics, a resource R is shared by the conjuncts.

Transitions between worlds, used to define the action modalities, are given by modifications:

$$R \models \langle a \rangle \phi \quad \text{iff} \quad \text{there exists } R \xrightarrow{a} R' \text{ such that } R' \models \phi$$

giving the possible truth of ϕ after the action a (with necessity satisfying $[a]\phi = \neg\langle a \rangle \neg\phi$).

The substructural connectives — key to the analysis of resource usage in BI [17, 18, 10] and Separation Logic [13, 19], including the Frame Rule, where the specific resource semantics of a program's stack/heap is analysed — use the monoidal structure of resources to separate properties of different parts of a given model:

$$R \models \phi_1 * \phi_2 \quad \text{iff} \quad \text{there exist } R_1 \text{ and } R_2\text{, where } R \sim R_1 \circ R_2\text{, such that } R_1 \models \phi_1 \text{ and } R_2 \models \phi_2$$

with the corresponding implication, $-\!\!*$, given as the right adjoint to $*$.

Recall Example 4 (semaphores). We can now formally state the property that the action aa cannot be performed on each of the resources in the monoid. The formula $\phi = \neg(\langle aa\rangle\top)$ denotes that there is no transition for the action aa. As $\mu(aa,e)$ and $\mu(aa,s)$ are not defined, we have that $e \not\models \langle aa\rangle\top$ and $s \not\models \langle aa\rangle\top$. We then straightforwardly have that $e \models \phi$ and $s \models \phi$. Note that, as $e \not\sim s$, the equivalence classes generated by $\sim$ are singleton sets, consisting of each of the two resources. We can also state that, on each resource of the monoid, there is no binary decomposition such that each of the two parts can perform an a action. This property is represented by the formula $\psi = \neg(\langle a\rangle\top * \langle a\rangle\top)$. The only S and T such that $e = S \circ T$ are $S = T = e$. The only S and T such that $s = S \circ T$ are $S = s$ and $T = e$, or $S = e$ and $T = s$. For each of these possible binary decompositions, at least one of the two parts cannot perform an a action, and hence at least one of the two parts does not satisfy $\langle a\rangle\top$. Hence, $e \models \psi$ and $s \models \psi$.

3 Strategies and cost

We address non-determinism in the transition systems generated by our resource semantics, as introduced in the previous section. We introduce a notion of cost, that represents the preferences of an entity (or agent) in a system. We describe how to systematically determine the cost associated with a resource. We conclude with a brief example.

The transition systems generated by our resource semantics can be non-deterministic, in the sense that multiple actions can be defined on a given resource.

Example 7. *Take a resource monoid* $(\{0,\dots,10\} \times \{0,\dots,10\}, \circ, (0,0))$*, where* $(m_1,m_2) \circ (n_1,n_2) = (m_1+n_1, m_2+n_2)$ *only if* m_1 *or* m_2 *is* 0 *and* n_1 *or* n_2 *is* 0 *(and is undefined otherwise). Suppose actions p and c. Let* $\mu(p,(m,n)) = (m,n+1)$*, if* $n \leq 9$*, and* $\mu(c,(m+1,n)) = (m,n)$*. Then, for the resource* $(2,0)$*, the actions p and c are both defined and, in the generated transition system, there is non-determinism between the distinct, non-unit, actions, p and c.* □

When evolving such non-deterministic transition systems, it is necessary to have a method to decide between possible options. A strategy can be used to determine, for a given resource, which possible action is preferred.

Definition 8 (Strategies)**.** *A strategy is a total function* $\sigma : \mathbf{R} \to \boldsymbol{Act}$ *such that, for all resources* $R,S \in \boldsymbol{R}$*, if* $R \sim S$*, then* $\sigma(R) = \sigma(S)$ *and* $\mu(\sigma(R),R)$ *and* $\mu(\sigma(R),S)$ *are defined.* □

Example 9. *We can define a strategy to resolve the non-determinism we saw in Example 7. Let* σ *be a function such that, if* $1 \leq m$*, then* $\sigma((m,n)) = c$*, and* $\sigma((m,n)) = p$*, otherwise. This strategy chooses the c action, whenever possible, and chooses the p action otherwise.* □

The resource semantics approach to distributed systems modelling abstracts away from the entities that make decisions, and their mechanisms for doing so. A mechanism for resolving choices can be re-introduced into the models through strategies: it does not, however, represent the goals and interests of the entities making the choices. We can model the decision-making-entities' preferences through the use of a map from actions to the rationals. These numbers are interpreted as measures of an agent's level of happiness in the given states [20].

Definition 10 (Action payoff function)**.** *An action payoff function is a partial function* $v : \mathbf{Act} \rightharpoonup \mathbb{Q}$ *s.t.* $v(1) = 0$ *and, for all* $a, b \in \mathbf{Act}$*, if* $v(a)$ *and* $v(b)$ *are defined, then* $v(ab) = v(a) + v(b)$*.* □

Note that it is possible to have that $v(ab)$ is defined, but that $v(a)$ and $v(b)$ are not defined (c.f., Example 18). We use different action payoff functions to represent the preferences of different decision-making entities. Fix an action payoff function v, a strategy σ, and let δ be some rational number in the open interval $(0,1)$. We can then straightforwardly extended the notion of preference over actions to preferences over resources.

Definition 11 (Resource payoff function)**.** *A resource payoff function is a partial function* $u_{v,\sigma,\delta} : \mathbf{R} \rightharpoonup \mathbb{Q}$ *such that*

$$u_{v,\sigma,\delta}(R) = \begin{cases} v(a) + \delta \times u_{v,\sigma,\delta}(\mu(a,R)) & \textit{if } \sigma(R) = a, \textit{ and } v(a) \textit{ and } u_{v,\sigma,\delta}(\mu(a,R)) \textit{ are defined} \\ \textit{undefined} & \textit{otherwise.} \end{cases} \qquad \square$$

The value that can be accumulated from actions performed at resources reachable in the future are worth less than value that can be accumulated immediately. The discount factor δ is used to discount future accumulated values. In the case that the set $\mathbf{R}$ is finite, we generate a finite set of simultaneous equations which can be solved using the methods described in [14]. Henceforth, we assume that all resource monoids have finite carrier sets.

Lemma 12. *For all action payoff functions* v*, strategies* σ*, and discount factors* δ*, if* $\sigma(R) = 1$*, then* $u_{v,\sigma,\delta}(R) = 0$*.*

Proof. By Definitions 3 and 10, we have that $\mu(1,R) = R$ and $v(1) = 0$. By Definition 11, we have that $u_{v,\sigma,\delta}(R) = 0 + \delta \times u_{v,\sigma,\delta}(R)$. As $(1-\delta) \neq 0$, we have that $u_{v,\sigma,\delta}(R) = 0$. □

Example 13. *We can now determine payoffs for various resources in Example 9 (which relies on Example 7). This is a simplification of a distributed systems example, presented fully in Example 16. Let* v *be an action payoff function such that* $v(p) = -1$ *and* $v(c) = 3$*, and* $\delta = 0.8$*. We then have that*

$$\begin{array}{rclcrcl} u_{v,\sigma,\delta}((0,0)) & = & 0 & \quad & u_{v,\sigma,\delta}((2,0)) & = & 3 + 0.8 \times u_{v,\sigma,\delta}((1,0)) \\ u_{v,\sigma,\delta}((1,0)) & = & 3 + 0.8 \times u_{v,\sigma,\delta}((0,0)) & & & = & 5.4 \\ & = & 3. & & & & \end{array}$$

With a different strategy, and the same action payoff, discount factor, and underlying systems model, different payoffs can be achieved. □

4 A modal logic of resources and utilities

We define a modal predicate logic, MBIU, for expressing properties of resources and their utility. Building directly on [8, 6], we define, in Figure 1, a semantics for MBIU in terms of the transition relation of a resource monoid, action monoid, and modification function, and its corresponding bisimulation relation.

Let term variables be denoted x, y, etc., and action variables be denoted α, β, etc.. The action terms of MBIU, building on actions a, b, c, etc., are formed according to the grammar

R	$\models$	$\mathrm{p}(t_1,\ldots,t_n)$	iff	$t_1^{\mathcal{U}(R)},\ldots,t_n^{\mathcal{U}(R)}$ are defined and $(t_1^{\mathcal{U}(R)},\ldots,t_n^{\mathcal{U}(R)},R)\in\mathcal{V}(\mathrm{p})$
R	$\models$	$t_1=t_2$	iff	$t_1^{\mathcal{U}(R)}$ and $t_2^{\mathcal{U}(R)}$ are defined and $t_1^{\mathcal{U}(R)}=t_2^{\mathcal{U}(R)}$
R	$\models$	$s_1=s_2$	iff	$s_1^{\mathcal{U}(R)}=s_2^{\mathcal{U}(R)}$
R	$\models$	$\bot$	never	
R	$\models$	$\top$	always	
R	$\models$	$\phi_1\vee\phi_2$	iff	$R\models\phi_1$ or $R\models\phi_2$
R	$\models$	$\phi_1\wedge\phi_2$	iff	$R\models\phi_1$ and $R\models\phi_2$
R	$\models$	$\neg\phi$	iff	$R\not\models\phi$
R	$\models$	$\phi_1\rightarrow\phi_2$	iff	$R\models\phi_1$ implies $R\models\phi_2$
R	$\models$	I	iff	$R\sim e$
R	$\models$	$\phi_1*\phi_2$	iff	there exist R_1, R_2, with $R\sim R_1\circ R_2$, such that $R_1\models\phi_1$ and $R_2\models\phi_2$
R	$\models$	$\phi_1\mathrel{-\!\!*}\phi_2$	iff	for all S, $S\models\phi_1$ implies $R\circ S\models\phi_2$
R	$\models$	$\langle s\rangle\phi$	iff	there exist a, R' such that $s^{\mathcal{U}(R)}=a$, $R\xrightarrow{a}R'$, and $R'\models\phi$
R	$\models$	$[s]\phi$	iff	for all a, R', $s^{\mathcal{U}(R)}=a$ and $R\xrightarrow{a}R'$ implies $R'\models\phi$
R	$\models$	$\exists\alpha.\phi$	iff	there exists $a\in\mathbf{Act}$ such that $R\models\phi[a/\alpha]$
R	$\models$	$\forall\alpha.\phi$	iff	for all $a\in\mathbf{Act}$, $R\models\phi[a/\alpha]$
R	$\models$	$\exists x.\phi$	iff	there exists $q\in\mathbb{Q}$ such that $R\models\phi[q/x]$
R	$\models$	$\forall x.\phi$	iff	for all $q\in\mathbb{Q}$, $R\models\phi[q/x]$

Figure 1: Satisfaction Relation for MBIU

$s ::= a \mid \alpha \mid s\diamond s$, where a ranges over **Act** and α ranges over action variables. Closed action terms are those that contain no variables. Fix a set of action payoff functions $\mathbf{V}$.

Let q be rational, $\mathrm{u_v}$ be a non-logical symbol denoting the resource payoff function $u_{v,\sigma,\delta}$ corresponding to an action payoff function $v\in\mathbf{V}$ (for a strategy and discount factor that are fixed in the interpretation of the logic). Let $\mathrm{v}(s)$ be the valuation of some action term, for some action payoff function $v\in\mathbf{V}$. Let the numerical terms, denoted t, t', etc., be formed according to the grammar $t ::= x \mid q \mid \mathrm{u_v} \mid \mathrm{v}(s) \mid t+t \mid t\times t$. Let closed terms be those that contain no variables.

We assume a set Pred of predicate symbols, each with a given arity n, with elements denoted p, q, etc.. Then, the formulae of MBIU are given by the following grammar:

$$\begin{array}{rcl}\phi & ::= & \mathrm{p}(t,\ldots,t) \mid t=t \mid s=s \mid \bot \mid \top \mid \phi\vee\phi \mid \phi\wedge\phi \mid \neg\phi \mid \phi\rightarrow\phi \\ & & \mid I \mid \phi*\phi \mid \phi\mathrel{-\!\!*}\phi \\ & & \mid \langle s\rangle\phi \mid [s]\phi \\ & & \mid \exists\alpha.\phi \mid \forall\alpha.\phi \mid \exists x.\phi \mid \forall x.\phi,\end{array}$$

where $|\mathrm{p}|=n$, $(t,\ldots,t)$ is an n-tuple of terms, $=$ is syntactic equality of the rationals, and t, s, x, and α range over terms, action terms, term variables, and action variables, respectively.

The (additive) modalities are the standard *necessarily* and *possibly* connectives familiar from modal logics, in particular Hennessy–Milner-style logics for process algebras [12, 16]. As such, they implicitly use meta-theoretic quantification to make statements about reachable resources. Multiplicative modalities can also be defined [8, 7]. The connectives $*$ and $-\!\!*$ are the multiplicative conjunction (with unit I) and implication (right-adjoint to $*$), respectively.

We define how atomic predicates are interpreted with respect to resources in Figure 1. Let

ϕ, ψ, etc. denote predicate formulae. The quantifiers $\exists\alpha$ and $\forall\alpha$ bind occurrences of action variables within predicate formulae and the modalities, and $\exists x$ and $\forall y$ bind occurrences of term variables within predicate formulae. Closed formulae contain no free term variables. The formula $\phi[q/x]$ is the formula formed by the (capture-avoiding) substitution of q for the term variable x that is free in ϕ. The formula $\phi[a/\alpha]$ is defined similarly.

The mathematical structure in which we interpret MBIU is the cartesian product of the set $\bigcup_{n\in\mathbb{N}}\mathbb{Q}^n$ of finite tuples of elements of the rationals and the set $\mathbf{R}$ of resources. In an interpretation, we fix a strategy σ and a discount factor δ. Recall that each resource generates a transition structure, via the modification function. An interpretation is given with respect to a particular resource R, and is written as $\mathscr{U}(R)$. The denotations of rationals and their addition and multiplication are the obvious ones in $\mathbb{Q}$. The denotation of the symbol u_v is given by $u_{v,\sigma,\delta}(R)$, as specified in Definition 11. Note that the corresponding interpretation of u_v is a constant, at a given resource R, and is given with respect to the fixed strategy and discount factor. The denotation of actions are themselves. The denotation of $\diamond$ is action composition $\cdot$.

Recall the bisimulation relation $\sim$. A set Σ of finite tuples of elements of the rational numbers and resources is said to be $\sim$-closed if it satisfies the property that, for all resources R and S, and for all rational numbers $q_1,\ldots,q_n$, $(q_1,\ldots,q_n,R)\in\Sigma$ and $R\sim S$ implies $(q_1,\ldots,q_n,S)\in\Sigma$. Let $\mathscr{P}_\sim(\bigcup_{n\in\mathbb{N}}\mathbb{Q}^n\times\mathbf{R})$ be the set of all $\sim$-closed sets of the cartesian product of the set of finite tuples of rational numbers and the set of resources. A valuation is a function $\mathscr{V}:\mathrm{Pred}\rightarrow\mathscr{P}_\sim(\bigcup_{n\in\mathbb{N}}\mathbb{Q}^n\times\mathbf{R})$, together with a fixed strategy and dicount factor. Every valuation extends in a canonical way to an interpretation for closed MBIU-formulae, the satisfaction relation for which is indicated in Figure 1. A model for MBIU consists of the resource monoid, action monoid, and modification function, together with such an interpretation. Satisfaction in a given model is then denoted $R\vDash\phi$, read as 'for the given model, the resource R has property ϕ', and is defined as in Figure 1.

An alternative formulation of MBIU with intuitionistic additives (cf. [17, 8]) can be taken if desired. Its used in modelling applications remains to be explored in future work.

We can now formally describe payoff properties of resources, in the following sense:

Example 14. *Recall Examples 7, 9, and 13. The formula*

$$\phi=\exists x,y.(\langle p\rangle\mathrm{u}_\mathrm{v}=x)\wedge(\langle c\rangle\mathrm{u}_\mathrm{v}=y)\wedge(v(p)+(\delta\times x)<v(c)+\delta\times y)$$

denotes that it is possible to perform actions p and c, and that the payoff obtained by performing p is less than that obtained by performing c. Note that $u_{v,\sigma,\delta((2,1))}=5.4$ *and* $u_{v,\sigma,\delta((1,0))}=3$. *As a result, we have that* $(2,0)\vDash\phi$. □

To obtain some key theoretical properties of our resource modelling framework, we require some additional properties. When we perform a composition of resources, it is necessary to take account of the partiality of the composition operator. As a result, we shall also require the following $\circ$-$\sim$-*closed* property of resource monoids. A resource monoid is $\circ$-$\sim$-closed if, for all resources R_1, S_1, R_2, $S_2\in\mathbf{R}$, if $R_1\sim S_1$, $R_2\sim S_2$, and $R_1\circ R_1$ are defined, then $S_1\circ S_2$ is defined. Henceforth, all resource monoids are assumed to be $\circ$-$\sim$-closed. When we interpret the payoff of resources, it is necessary to take account of bisimilarity. A model is payoff-$\sim$-closed if, for all $v\in\mathbf{V}$, $R,S\in\mathbf{R}$, $R\sim S$ and $u_{v,\sigma,\delta}(R)$ is defined implies that $u_{v,\sigma,\delta}(S)$ is defined and $u_{v,\sigma,\delta}(R)=u_{v,\sigma,\delta}(S)$. From this point onwards, all models are assumed to be payoff-$\sim$-closed.

With this set-up, we can prove the Hennessy–Milner soundness and completeness theorem. The soundness direction of the Hennessy–Milner completeness theorem — operational equivalence implies logical equivalence — requires the congruence property.

Theorem 15. *$R \sim S$ iff, for any model of MBIU and all ϕ, $R \vDash \phi$ iff $S \vDash \phi$.*

Proof. For soundness — operational equivalence implies logical equivalence — by induction over the structure of the formulae, using Theorem 6 and the satisfaction relation. Completeness — logical equivalence implies operational equivalence — follows [8, 7]. □

Theorem 15 provides basic assurance that the logic is well formulated, and supports the formulation of proof systems and reasoning tools, such as model checking.

5 Examples and optimality

To illustrate the logical set-up we have introduced, we begin with a classic example from distributed systems modelling: mutual producer–consumer. We then explain, using a generic example, how our set-up can be used to express Pareto optimality. This example leads naturally into a discussion of game-theoretic examples and concepts. We consider here the prisoner's dilemma, the best-response property, and Nash equilibrium.

Example 16 (Mutual producer–consumer)**.** *A classic example of distributed systems modelling is distributed coordination without mutual exclusion, the most common form of which is that of the producer–consumer system [7, Section 2.3.5]. In such a scenario, one entity generates work that another entity can handle at a later point. We modify this slightly to the scenario with two entities, where each entity can generate work for, and consume work from, the other.*

We extend Example 7. Suppose a resource monoid $(\{0,\ldots,10\} \times \{0,\ldots,10\}, \circ, (0,0))$, where $(m_1,m_2) \circ (n_1,n_2) = (m_1+n_1, m_2+n_2)$ if either m_1 or m_2 is 0 and either n_1 or n_2 is 0.

The elements of the resource monoid are pairs of natural numbers, where the first element of the pair denotes the number of work packages that the first entity can consume, and the second element of the pair denotes the number of work packages that the second entity can consume.

Suppose actions p_1, p_2, c_1, and c_2, where $\mu(p_1,(m,n)) = (m,n+1)$ if $n \leq 9$, $\mu(c_1,(m+1,n)) = (m,n)$, $\mu(p_2,(m,n)) = (m+1,n)$ if $m \leq 9$, and $\mu(c_2,(m,n+1)) = (m,n)$. The p_1 action denotes production of a work package by the first entity for the second entity, and the c_1 action denotes the consumption of a work package by the first entity. The p_2 and c_2 actions have the obvious converse denotations.

Consider the situation where the processes 'profit' from the consumption of work packages, and must 'pay' to create work packages. A pair of possible payoff functions v_1 and v_2, for the two entities, which represents this situation is $v_1(p_1) = -1$, $v_1(c_1) = 3$, $v_1(p_2) = 0$, $v_1(c_2) = 0$ $v_2(p_1) = 0$, $v_2(c_1) = 0$, $v_2(p_2) = -2$, and $v_2(c_2) = 4$.

Let σ be a function such that, if $1 \leq m$ and $1 \leq n$, then $\sigma((m,n)) = c_1c_2$, if $1 \leq m$, then $\sigma((m,0)) = c_1$, if $1 \leq n$, then $\sigma((0,n)) = c_2$, and $\sigma((0,0)) = p_1p_2$. Let the discount factor δ be 0.8. *Consider the unit resource,* $(10,0)$. *As there are only work packages available for the first entity, the actions defined on the resource are the consume action c_1, the produce action p_1, and the unit. Each entity incurs costs by performing a produce action, which only benefits the other entity. We have $v_1(p_1) + \delta \times u_{v_1,\sigma,\delta}(10,1) \approx -1 + \delta * 13.4 \approx 9.7$, $v_1(c_1) + \delta \times u_{v_1,\sigma,\delta}(9,0) \approx$*

13.4, $v_2(p_1) + \delta \times u_{v_2,\sigma,\delta}(10,1) = 0 + 0.8 * 4 = 3.2$, and $v_2(c_1) + \delta \times u_{v_2,\sigma,\delta}(9,0) = 0$. The action c_1 gains the most for the first entity and p_1 gains the most for the second. □

For either action, it is not possible to swap to an alternative action that makes one of the entities better off, without making the other entity worse off. This notion is called *Pareto optimality*.

Definition 17 (Pareto optimality)**.** *A state R is Pareto optimal if there exists an action a such that, for all other actions b, if some entity (weakly) prefers that action b be performed, then there is some other agent that strongly prefers that action a be performed. Formally, the state R is Pareto optimal if, for entities with payoff functions $v_1, \ldots, v_n$,*

$$R \models \exists\alpha\,.\,\forall\beta\,.\,(\neg(\beta = \alpha)) \rightarrow$$

$$\left(\begin{array}{l} \forall x,x'.\exists y,y'. \\ \quad(((\langle\alpha\rangle \mathtt{u}_{\mathtt{v}_1} = x) \wedge (\langle\beta\rangle \mathtt{u}_{\mathtt{v}_1} = x') \wedge (x \leq x')) \rightarrow \\ \qquad(((\langle\alpha\rangle \mathtt{u}_{\mathtt{v}_2} = y) \wedge (\langle\beta\rangle \mathtt{u}_{\mathtt{v}_2} = y') \wedge (y' < y)) \\ \qquad\vee \ldots \vee \\ \qquad(((\langle\alpha\rangle \mathtt{u}_{\mathtt{v}_\mathtt{n}} = y) \wedge (\langle\beta\rangle \mathtt{u}_{\mathtt{v}_\mathtt{n}} = y') \wedge (y' < y)) \end{array}\right) \vee \ldots \vee \left(\begin{array}{l} \forall x,x'.\exists y,y'. \\ \quad(((\langle\alpha\rangle \mathtt{u}_{\mathtt{v}_\mathtt{n}} = x) \wedge (\langle\beta\rangle \mathtt{u}_{\mathtt{v}_\mathtt{n}} = x') \wedge (x \leq x')) \rightarrow \\ \qquad(((\langle\alpha\rangle \mathtt{u}_{\mathtt{v}_1} = y) \wedge (\langle\beta\rangle \mathtt{u}_{\mathtt{v}_1} = y') \wedge (y' < y)) \\ \qquad\vee \ldots \vee \\ \qquad(((\langle\alpha\rangle \mathtt{u}_{\mathtt{v}_{\mathtt{n}-1}} = y) \wedge (\langle\beta\rangle \mathtt{u}_{\mathtt{v}_{\mathtt{n}-1}} = y') \wedge (y' < y)) \end{array}\right)$$

We abbreviate the above formula as $PO(v_1, \ldots, v_n)$. □

In Example 16, the resource $(10,0)$ is Pareto optimal, witnessed by both the actions p_1 and c_1. Note that optimality is defined in terms of actions; this is as, here, we take seriously the representation of actions that *perform* resource allocations. A transition is then an (actively performed) resource allocation.

One field in which notions of optimality have been studied significantly is that of games and decision theory. We can model games in our resource semantics. A classic decision-making example from game theory is the prisoner's dilemma.

Example 18 (Prisoner's dilemma)**.** *Two individuals have been arrested, and are kept separately, so that they cannot collude in their decision making. Each is offered the choice of attempting to 'defect', and give evidence against their partner, or to 'collaborate', and say nothing. If one person collaborates and the other defects, then the collaborating partner goes to jail for a long time, and the defecting partner goes free. If both people defect, then they both go to jail for a moderate time. If both people collaborate, then they both go to jail for a short time.*

Suppose a resource monoid $(\{r_1, r_2, r_{1,2}, e\}, \circ, e)$, where $r_1 \circ r_2 = r_{1,2}$. The r_1 resource denotes a resource where the first person can make a choice, the r_2 resource denotes a resource where the second person can make a choice, and the $r_{1,2}$ resource denotes a resource where both people can make a choice at the same time. Suppose actions c_1, d_1, c_2, and d_2, where $\mu(c_1, r_1) = \mu(d_1, r_1) = e$, $\mu(c_2, r_2) = \mu(d_2, r_2) = e$, and $\mu(c_1c_2, r_{1,2}) = \mu(c_1d_2, r_{1,2}) = \mu(d_1c_2, r_{1,2}) = \mu(d_1d_2, r_{1,2}) = e$. The c_1 action denotes collaboration by the first person, and the d_1 action denotes defection by the person. The c_2 and d_2 actions have the obvious denotations for the second person. We make use of the trivial strategy $\sigma(R) = 1$. The action payoff functions v_1 and v_2 for the two people are $v_1(c_1c_2) = -2$, $v_1(c_1d_2) = -6$, $v_1(d_1c_2) = 0$, $v_1(d_1d_2) = -4$, $v_2(c_1c_2) = -2$, $v_2(c_1d_2) = 0$, $v_2(d_1c_2) = -6$, and $v_2(d_1d_2) = -4$. Hence, if the first person collaborates and the second defects, then the first person receives six years in prison (cost $v_1(c_1d_2) = -6$), while the second receives no time in prison (cost $v_2(c_1d_2) = 0$). □

We can define notions of *best response* and *Nash equilibrium.*

Example 19 (Best response)**.** *An action a is a best response for a given entity to a particular choice of action b by another entity, at a given resource, if the (former) entity has no other action c available to it such that the action cb is defined on the resource and the entity (strongly) prefers cb to ab. Formally, a is the best response to action b at resource R if*

$$\begin{array}{rcl} R & \models & \forall\alpha.\exists x,y.\Big(\big(((\langle a\rangle\top\wedge\langle\alpha\rangle\top)*(\langle b\rangle\top)\big)\wedge\big([ab](\mathtt{u}_{\mathtt{v}}=x)\wedge[\alpha b](\mathtt{u}_{\mathtt{v}}=y)\big)\Big) \\ & & \to\big((v(\alpha b)+\delta\times y)\leq(v(ab)+\delta\times x)\big). \end{array} \qquad \Box$$

We abbreviate the above formula, denoting that a is the best response to action b for the agent whose payoff function is v, as $BR(a,b,v)$. In the prisoner's dilemma example, the best response for the first agent to the action c_2 is d_1, and $BR(d_1,c_2,v_1)$ holds.

We generalize this notation slightly, so that we write $BR(a,b_1,\ldots,b_n,v)$ to denote that a_1 is the best response the the composite action $b_1\ldots b_n$, for the payoff function v. Formally,

$$\begin{array}{rcl} R & \models & \forall\alpha.\exists x,y.\Big(\big(((\langle a\rangle\top\wedge\langle\alpha\rangle\top)*(\langle b_1\ldots b_n\rangle\top))\rangle\big)\wedge\big([ab_1\ldots b_n](\mathtt{u}_{\mathtt{v}}=x)\wedge[\alpha b_1\ldots b_n](\mathtt{u}_{\mathtt{v}}=y)\big)\Big) \\ & & \to\big((v(\alpha b_1\ldots b_n)+\delta\times y)\leq(v(ab_1\ldots b_n)+\delta\times x)\big). \end{array}$$

Here, for simplicity, we suppress all issues concerned with the structure of the composite action $b_1\ldots b_n$: In general, a process-theoretic treatment, allowing control over the presumed nature of the concurrent composition, can be given [8, 7]. Now we can express Nash equilibrium.

Example 20 (Nash equilibrium)**.** *A state R is a Nash equilibrium for a set of entities* $I=\{1,\ldots,n\}$ *if there is a collection of actions* $a_1,\ldots,a_n$ *such that, for each entity* $i\in I$ *with payoff function* v_i*, the action* a_i *is the best response to the composition of actions* a_j*, where* $j\in I\setminus\{i\}$*.*

Formally, the state R is a Nash equilibrium if

$$R\models\exists\alpha_1\ldots\alpha_n.BR(\alpha_1,\alpha_2,\ldots,\alpha_n,v_1)\wedge\ldots\wedge BR(\alpha_n,\alpha_1,\ldots,\alpha_{n-1},v_n). \qquad \Box$$

We abbreviate the above formula as $NE(v_1,\ldots,v_n)$. In the prisoner's dilemma example, the Nash equilibrium is the state $r_{1,2}$, witnessed by the actions d_1 and d_2, for payoff functions v_1 and v_2, and the property $NE(v_1,v_2)$ holds.

6 Discussion

Notice, in the examples of Section 5, the key role played in the formulae *BR* by the multiplicative conjunction, $*$. Used with the additives, it allows the separation of the resources allocated locally to different actions (the *a*s and *b*s) to be enforced when required whilst allowing utility properties of the overall system to be expressed relative to the overall resources, as required.

In a richer set-up, retaining explicit process structure — recall the discussion of Section 1 — the trace leading to the optimal and equilibrium states, together with its history of resource usage, would be represented explicitly (though at some technical cost in the development). Presentation of this richer view is deferred to another occasion.

By developing such a view we should be able to incorporate the analysis of utility and optimality presented here into the widely deployed systems and security modelling tools established in, for example, [8, 6, 7], with deployments described in, for example, [15, 1, 5, 3, 4].

References

[1] Y. Beres, D. Pym & S. Shiu (2010): *Decision Support for Systems Security Investment*. In: *Proc. 5th BDIM*, IEEE Xplore, pp. 118–125.

[2] G. Birtwistle (1987): *Discrete event modelling on Simula*. Springer.

[3] T. Caulfield & D. Pym (2015): *Improving Security Policy Decisions with Models*. *To appear, IEEE Security and Privacy*.

[4] T. Caulfield & D. Pym (2015): *Modelling and Simulating Systems Security Policy*. In: *To appear, Proc. 8th. SIMUTools*, ACM Digital Library.

[5] T. Caulfield, D. Pym & J. Williams (2014): *Compositional Security Modelling: Structure, Economics, and Behaviour*. *LNCS* 8533, pp. 233–245.

[6] M. Collinson, B. Monahan & D. Pym (2010): *Semantics for Structured Systems Modelling and Simulation*. In: *Proc. SIMUTools*, ICST, Brussels, Belgium, pp. 34:1–34:10.

[7] M. Collinson, B. Monahan & D. Pym (2012): *A Discipline of Mathematical Systems Modelling*. College Publications.

[8] M. Collinson & D. Pym (2009): *Algebra and Logic for Resource-based Systems Modelling*. *Mathematical Structures in Computer Science* 19(5), pp. 959–1027.

[9] G. Coulouris, J. Dollimore & T. Kindberg (2000): *Distributed Systems: Concepts and Design*, 3rd edition. Addison Wesley.

[10] D. Galmiche, D. Méry & D. Pym (2005): *The Semantics of BI and Resource Tableaux*. *Mathematical Structures in Computer Science* 15(6), pp. 1033–1088.

[11] D. Harel, J. Tiuryn & D. Kozen (2000): *Dynamic Logic*. MIT Press, Cambridge, MA, USA.

[12] M. Hennessy & G. Plotkin (1980): *On Observing Nondeterminism and Concurrency*. *Lecture Notes in Computer Science* 85, pp. 299–308.

[13] S. Ishtiaq & P. O'Hearn (2001): *BI as an Assertion Language for Mutable Data Structures*. In. *Proc. 28th POPL*, ACM, pp. 14–26.

[14] J.-B. Jeannin, D. Kozen & A. Silva (2013): *Language Constructs for Non-well-Founded Computation*. In: *Proc. 22nd ESOP*, Springer-Verlag Berlin, Heidelberg, pp. 61–80.

[15] Hewlett-Packard Laboratories: *Towards a Science of Risk Analysis*. Available at `http://www.hpl.hp.com/news/2011/oct-dec/security_analytics.html`.

[16] R. Milner (1989): *Communication and Concurrency*. Prentice Hall, New York.

[17] P. O'Hearn & D. Pym (1999): *The Logic of Bunched Implications*. *Bulletin of Symbolic Logic* 5(2), pp. 215–244.

[18] D. Pym, P. O'Hearn & H. Yang (2003): *Possible Worlds and Resources: The Semantics of BI*. *Theoretical Computer Science* 315(1), pp. 257–305.

[19] J. Reynolds (2002): *Separation Logic: A Logic for Shared Mutable Data Structures*. In: *Proc. 17th LICS*, IEEE, pp. 55–74.

[20] Y. Shoham & K. Leyton-Brown (2008): *Multiagent Systems: Algorithmic, Game-Theoretic, and Logical Foundations*. Cambridge University Press, New York, NY, USA.