On Computational Responsibility

William T. Wallis

School of Computing Science
Sir Alwyn Williams Building
University of Glasgow
G12 8QQ

Masters project proposal

Contents

1 Introduction

Computational formalisms of social constructs are an increasingly common research area. For example, researchers have so far tackled a variety of social notions through computational formalism:

- Marsh's seminal work on Trust[Mar94]
- Stricter formal definitions on Trust, from a cognitive standpoint[CF]
- Some responsibility modelling, from a logical formalisation[SS15]
- Some work on reputation [CE11]
- Models of computational comfort models[MBEK+11].

Finish reading this!

These models of social constructs are useful in a variety of ways: Marsh's model, for example, gave rise to new methods in solving problems in fields as diverse as HCI[PMB05] and systems modelling[HJS06]. However, responsibility as a social construct has been neglected: no literature on responsibility formalism has been published to date. This is curious, as responsibility modelling is a field which has proven particularly useful — therefore, a logical next step for responsibility as a subject of study within sociotechnical systems analysis would be the computational formalism of the trait. As will be demonstrated, responsibility as a computational concept may yield a great number of research opportunities in fields such as sociotechnical systems modelling, machine learning and decision theory, and even humanities such as the philosophy of mind.

A responsibility formalism is useful in the same ways that formalisms of human traits such as reputation and trust might be; however, a computational theory of responsibility has the potential to have impacts in areas trust and reputation might not. For example, imbuing an intelligent agent with a sense of responsibility might provide it a greater degree of corrigibility[SFYA15]. An agent overseeing network security which understands its responsibilities within a much larger security system might better prioritise its duties when confronted with an unusual situation. Computational responsibility frameworks might help better model the emergent phenomena in sociotechnical systems, combine with traits like trust and comfort to make a more anthropomorphic device for better HCI, or perhaps help predict human actions in large computational models of human actors. We will explore some of these practical applications in § 4.

However, it is certain that a uses for these formalisms present themselves at every turn.

Write something concrete here regarding fields it might be applicable in, like decision theory or AI safety or network security or sociotechnical modelling

1.1 An early rebuttal of some common criticisms

One easy criticism made of these anthropomorphic formalisms is the argument that, say, a trust formalism doesn't represent "true" trust. To address this point early, a responsi-

bility formalism such as the one proposed need not be an entirely human-like representation of responsibility for every definition. Rather, there is a utility in an agent giving the *appearance* of responsibility. (If one follows the deterministic school of thought, there is also an argument that there is no difference[Hon].)

Whether one considers it "true" responsibility should arguably be secondary to whether responsibility-like traits are useful to have computational frameworks for; we will see that these traits are indeed useful, and so that the criticism is moot. Computational trust formalisms are well documented as a valuable asset in solving HCI problems and designing aspects of intelligent agents, such as decision functions. We will see that computational responsibility follows in these footsteps, and has applications in AI and HCI just like trust. There are added benefits to responsibility formalisms, however, such as applications to a wider range of interdisciplinary study, and a very direct application in solving problems such as decision problems.

1.2 The scope of the model

Useful context for considering what sort of agents might be "responsible" can be found in exploring the agents our formalism might apply to. As will be explored in § 3.3.1, there exist types of agents which we might not consider responsible in an ordinary setting, or for whom irresponsible behaviour might affect an assessment of responsibleness. A human agent who is mentally handicapped, compared to a human agent with ordinary brain function but is lazy, wouldn't be seen as irresponsible when failing to hand an assignment in on time. To account for this difference in how agents' actions are accounted for in the proposed formalism, we limit the scope of what the formalism might model to agents who are:

- Reflective
- Interpretive
- Reactive

1.2.1 Reflective Agents

Sloman's work in the "space of minds" [Slo84] shows that artificial "minds" need not be remotely human-like. In order to limit the space of agents the proposed formalism would apply to, then, one might limit the space of minds those agents might inhabit.

There are several ways to limit the space of mind of the agents a formalism concerns: for example, C&F define a "cognitive" agent as the lower limit of an agent's requirements for human traits for trust. C&F define a cognitive agent as:

> Only a cognitive agent can "trust" another agent; only an agent *endowed with goals and beliefs*.

This definition doesn't quite fit our purposes — as will be seen, our definition also requires the concept of *obligation*. However, it can be seen that this definition is deliberately high-level in order to simulate the important components of a human trusting agent. A cognitive agent can be seen as an agent which, for the task it is set out to do, is modelled in a *high-level, human like way*.

Therefore, we might define our own high-level requirement of responsible computational agents:

> Only a reflective agent can be "responsible" for its actions; only an agent which can *reflect on its obligations when choosing an action*.

A simpler way to state this, for the purposes of implementation in an artificially intelligent agent, would be that an intelligent agent should parametrise its decision function by its obligations. In this way, obligation to a certain goal or outcome influences actions chosen by that agent; this considering of obligation is required for those actions to be "responsible", because an agent which does not parameterise by its obligations would have no way of accounting for what it ought to do when choosing an action by definition.

1.2.2 Interpretive, Reactive Agents

One can imagine other useful limitations of scope, too. For example, an agent should be able to interpret their own behaviours as responsible or irresponsible, such that they can assert the degree to which they should weight their obligations in their decision process:

> Involved in an "interpretive agent"ś judgement of their responsibilities is a subjective component: an interpretive function which converts information about an obligation or duty into a subjective score of responsibility.

This way, human-like subjectivity of responsibility can be simulated. We might go one step futher, and more tightly constrain the subjective nature of an agent:

> Involved in a "reflective agent"ś judgement of their responsibilities is a subjective component: an interpretive function which converts information about an obligation or duty into a subjective score of responsibility.

One can see that, when this limitation on the agents a formalism concerns comes into place, that formalism becomes useful regardless of its computational application. While the formalism might be algorithmic in nature, the concepts behind it can be applied to social sciences also as the agents it concerns becomes more anthropomorphic. The interdisciplinary nature of a formalism such as this is a great asset in many areas, and allows for a common jargon when, for example, HCI researchers work with ethnographers in understanding the responsibility of a user.

The purpose of describing these terms is twofold. Partly, it is to introduce the notion that a proposed formalism of responsibility wold be sensible in the types of agents it would target; this is useful to bear in mind when considering a trait which is normally human-specific developed as an algorithm. These terms are also introduced to show that the interdisciplinary jargon a formalism creates is naturally and easily defined by the formalism's construction. This shows in a concrete way that the formalism, even in an early form, has clear utility.

1.3 Proposal Overview

This proposal will be split into five main sections:

Is the list of sections up to date?

1. This introduction (§ 1), which lays the foundation for the research to be done and gives context for the background survey in the following section
2. A brief problem statement (§ 2), which explores in specific terms the research intended to be undertaken
3. A background survey (§ 3), which explores related literature to computational responsibility, including:

   Is the list of background topics up to date?

   - Mathematics and Social Sciences
   - Sociotechnical Systems research
   - Philosophical research
4. A proposed approach (§ 4) to undertake the research suggested, which will explore potential options for the formalism and show how the relevant literature informs specific possible formalisms
5. A brief work plan (§ 5), which proposes a timeline for the work outlined in earlier sections

2 Statement of Problem

With some background exploration on what a formalism of responsibility might entail, and an overview of its scope and utility, we can see that some formalism of responsibility has genuine utility. However, assessing how it might apply to artificial agents in practice requires the development of the formalism itself. It also requires that the formalism be applied to the specific category of minds outlines in § 1.2: reflective, interpretive, reactive agents.

We can address the feasibility of a real formalism which applies to these agents — and develop said formalism in the process — by answering the following research questions:

1. How can a computational formalism of responsibility direct the decisions made by an intelligent agent?
2. How can an intelligent agent assume the consequences of actions it makes, the decisions other agents make, and its general environment, so as to direct its interpretation of responsibility?

I propose that this work would provide a valuable addition to the development of anthropomorphic trait formalisms, and that the work is also useful and interesting in its own right.

Finish this section!

3 Background Survey

Unlike Computational Responsibility, Computational Trust is a topic which has a surprising degree of pre-existing literature. Marsh [Mar94] draws inspiration from as early as David Birkhoff's 1930s work in creating an 'Aesthetic Measure', where the famous mathematician created a quantification of Aesthetics. While some dispute that such subjective topics can be boiled down to a single number (or array thereof), much work to the contrary has now been completed. Like Marsh, we should start from the beginning.

3.1 Social Sciences and Mathematics

3.1.1 Birkhoff's Aesthetic Measure

One of the earlier formalisms of a human factor[1][2] was Birkhoff's definition of Aesthetic Measure[Bir]. In it, Birkhoff defines the notion of Aesthetic Measure as a ratio of Order to Complexity:

$$M = \frac{O}{C}$$

Birkhoff's work inadvertently gave rise to the notion that human factors can be represented by mathematical equations and systems. Birkhoff's formalism of aesthetics became popular for a few reasons, but one of particular interest to later Trust modelling work was that Birkhoff put a great degree of effort into backing his work up with psychological theory. In this way, Birkhoff's formalism could be said to be a *psychological* formalism.

Later trust modelling work followed in Birkhoff's footsteps here. Indeed, Birkhoff gives a solid foundation for the model-creating method later employed by Marsh[Mar94] and Castelfranchi & Falcone, as it is:

- Founded on mathematical or logical principles which are *quantifiable*
- Heavily inspired and directed by related work in psychology, sociology, and philosophy

[1]For the sake of clarification, we define a "human factor" as an element of a social or sociotechnical system which arises from human behaviour, such as Trust.

[2]Also for the sake of clarifying a sociotechnical system, a sociotechnical system is a system composed of human tendencies and behaviours, such as Trust, alongside technical activity, such as a computer or a steam engine. An example might be a coffee shop:

- Humans take orders and manage the running of the shop
- Technology is responsible for complex activities such as taking payments and forcing steam through coffee at high pressure

so there are both social and technical actors and behaviours in the "system" of a day-to-day coffee shop.

The marriage of social studies with mathematical rigour will be a recurring theme of the work related to Computational Trust.

3.1.2 Deutsch

Following the quantifiable, mathematical work done by Birkhoff, logical and arithmetic formalisms of human factors followed. One of the earlier and more widely adopted models for Trust came from Deutsch in 1962. Deutsch is a psychologist who did swathes of work in the topic of cooperation, touching on Trust during the 60s.

Deutsch's formalism of trust wasn't immediately quantifiable, but presented one of the earliest well-defined definitions of trust. To paraphrase Deutsch's formalism in "Cooperation and Trust: Some Theoretical Notes", 1962:

CITE THIS

- An actor is presented with a choice between two paths.

  A: No change

  B: The actor takes some action, of ambiguous outcome. A possible gain is associated, *P*, and some possible risk is associated, *R*.

- The actor assesses that the outcome of choice *B* relies on the behaviour of another actor.

- The actor assesses the action they may take and resolves that the strength of *R*, likelihood of *R* as an outcome, or both are higher than the respective *P* measurements.

- The actor is said to be *trusting* they take path *B*.

This formalism introduces some interesting notions. For example, it is unclear as to whether the outcome of choice *B* can rely on the same actor making the decision; can one trust oneself by Deutsch's definition? Another interesting analysis of the implications of Deutsch's model is that it does not rely on the *accurate* measurement of risk and utility, but just its perception — trust is subjective, and based on the trusting actor's perspective on the world.

Rather than characterising trust by the parties involved, Deutsch's formalism is characterised by *risk and utility*. A simple quantification of Deutsch's formalism could be devised, therefore, where risk and utility are quantified by simple assessments using utility functions and a form of risk analysis. Even so, the outcome of this quantified system is a single bit: trusting or not trusting. This does quantify trust, but only technically speaking, and this quantification is weak in its expressiveness. It gives no remit to suggest that one might trust one person over another, for example, as there are no order-able degrees of trust.

Deutsch offers many different ideas as to why and how trust or trust-like behaviour can come about, however. This list is taken from Marsh 1994[Mar94], where explanations of all nine can be found:

1. Trust as Despair
2. Trust as Social Conformity
3. Trust as Innocence
4. Trust as Impulsiveness
5. Trust as Virtue
6. Trust as Masochism
7. Trust as Faith
8. Risk-taking or Gambling
9. Trust as Confidence

Deutsch's given model above specifically targets formalisation of trust as confidence.

3.1.3 Luhmann

Luhmann, a sociologist who also worked in Trust and related fields, had his own take on formalisms of Trust: that trust was a social tool for reducing the complexity of a social system. Specifically, Luhmann sees trust as being a method whereby agents in a social system can reduce their exposure of *risk* to each other. According to Luhmann, "Trust... presupposes a situation of risk."

CITE THIS

Luhmann's work is therefore difficult to form quantitative formalisms from, as his thesis stems from a risk analysis perspective, which can be particularly difficult in a sociotechnical system. However, Luhmann's work remains interesting; a formalism of a human factor like trust would be incomplete without considering the properties of individual human actors as well as these properties' emergent effects in the larger sociotechnical space. For small systems, these social-level properties may not present themselves very strongly; however, most human factors are present regardless of the scale of the system being modelled. Therefore, a formalism of a human factor which fails to consider both psychological and sociological aspects cannot be complete.

3.2 Modern [Computational] Trust methods

3.2.1 Marsh's formalism

The earliest quantifiable formalism of trust which provides computability, flexibility, and an inspiration from the sociological and psychological work above is that of Stephen Marsh in 1994[Mar94]. Marsh's work breaks trust up into three core quantifications, where each variable takes some value in the range $[-1, 1)$:

1. Basic Trust
   This is the general degree of "trustingness" about an agent, or that agent's ordinary inclination to trust.

2. General Trust
   General trust is trust in the context of the agent being trusted. Marsh's original description begins[Mar94]:

   > Given two agents, $x, y \in \mathcal{A}$, to notate 'x trusts y' we use: $T_x(y)$. ...The value represents the amount of trust x has in y here.

   So, General Trust can be seen to be the trust that an agent x has in y.

3. Situational Trust
   Trust doesn't exist in a vacuum, and the only variable isn't the subject of x's trust; y may have varying degrees of competency in performing an action. Therefore, Situational Trust can be seen to be the trust x holds that y can actually perform some task, α. Marsh helpfully gives the example[Mar94]:

   > ...whilst I may trust my brother to drive me to the airport, I certainly would not trust him to fly the plane!

Marsh's three types of trust are helpful in breaking down what matters when discussing trust — notions like competency, for example — as well as establishing a jargon for trust. Often, one might say that a person is "*trusting*": Marsh's formalism accounts for concepts like this, but establishes it as a less detailed type of trust, and a type of trust which doesn't account for the action being trusted for, or whether the trusted agent is able to complete the action.

Marsh also succeeds in introducing concrete examples of computational formalisms of ordinarily human traits — here Trust. The key aspect of Marsh's advancement is that it goes one step further than a *quantitative* model, and introduces reinforcement learning algorithms which model how trust *changes*, and not just its current state. As seen when discussing Birkhoff's work (§ 3.1.1), quantitative formalisms of human traits like Aesthetics had been studied and achieved long before Marsh's work.

Since Marsh's work, many trust models have been developed. A small subset of these are reviewed here; offshoots from this seminal work include REGRET, Eigentrust, FIRE,

Should we discuss this too?

and others.

Finish notes on Marsh's computational trust model!

3.2.2 Castelfranchi & Falcone

As it turns out, cognitive computational trust models that already exist are almost but not quite appropriate for modelling responsibility. The C&F trust model requires only four main ingredients to formulate a cognitive trust model:

1. x, a truster
2. y, a subject of trust
3. g, a goal of x
4. α, an action of y

This model gets us close to where we need to be to model responsibility; like responsibility modelling often does, it assumes two agents. There also exists some goal which can be met, which — to use C&F terminology — is *delegated* by x to y. Y can achieve this goal through some action, α. So far, all of this forms the beginning of a foundation for cognitive responsibility; what turns delegation of a task into the consignment of responsibility is that of obligation, and the understanding of obligation.

It is evident that trust and responsibility models are, even in the human-like cognitive approach, very similar. However, there are drawbacks which mean that we cannot directly apply C&F theory to the idea of computational responsibility: it does not represent any degree of obligation or address the specific problem of judging responsibility at all.

Nevertheless, this presents an exciting insight into work to be done to produce a formalism of responsibility. Particularly, it is evident that there is at least some technical value in listing the individual components as C&F do. Their simple, reduced approach implies that with the correct identification of elements of responsibility, our formalism can be similarly simple. It is also encouraging that connections between trust and responsibility modelling seem to readily present themselves. We can therefore expect our formalism to rightly exhibit a similar structure and features.

3.2.3 Ian Sommerville, Sociotechnical Systems, and Responsibility Modelling

Sommerville's work focuses largely on sociotechnical systems and responsibility modelling — in this way, Sommerville's work is not typically concerned with computational models of trust, as the above were. However, his work does begin to border on our own advancements, providing responsibility modelling formalisms.

Finish discussing Sommerville's contribution to the state of the art.

It is important to note that Ian Sommerville has been a particularly prolific writer for a researcher in the sociotechnical systems scene. Sommerville's modelling systems are sometimes graphical[LSSB10]. Unfortunately, graphical modelling systems do not lend themselves particularly well to computational formalism: they don't yield naturally to numerical analysis; they are generally designed for the purposes of human visual analysis, instead of logical reasoning; they are often difficult to represent non-graphically, which arguably makes input and manipulation too complex for the purposes of designing a complex intelligent agent around.

Ultimately, though, graphical responsibility modelling systems are designed for representing the responsibilities of a single agent at a given point in time; a responsibility formalism, by contrast, should be a series of metrics and rules which can apply to arbitrary reasoning of an agent's responsibility through time, with that agent using the formalism to reason about its changing responsibilities. In other words, graphical responsibility modelling differs from a responsibility formalism in that a responsibility formalism needs to *generalise* reasoning about how responsibility, as a concept, "behaves".

Nevertheless, some sociotechnical work on responsibility prevents an invaluable addition to relevant literature for developing its computational formalism. In particular is Sommerville's work on "Causal" and "Consequential" responsibilities. In defining these terms, Sommerville writes[Som07]:

> Consequential responsibility can only be assigned to a person, a role or an organisation automated components cannot be blamed. Causal responsibility reflects who or what is responsible for making something happen or avoiding some undesirable system state. It is often the case that these are separated.

The separation of concerns between consequential and causal responsibilities can help us to inform the structure and nature of a responsibility formalism. Particularly, one can simplify these definitions to be that:

> Consequential responsibilities are responsibilities which relate to a state arrived at in the past, and the relationship of actors and actions with said state.

> Causal responsibilities are responsibilities which relate to future states, and the actors and actions which are potentially assigned with the goal of arriving at said state.

This separation of past responsibilities and future responsibilities means that we can structure the concerns and operation of a responsible agent according to a given formalism. Particularly, for our purposes, the assessment of one's responsibility to arrive at future states — one's assessment of its *Causal Responsibilities* — might be informed

by information an agent's information about its previous responsibilities and its ability to act responsibly — one's *Consequential Responsibilities*. Therefore, we might decide to create a responsibility formalism which specifically models the change in causal responsibility, by assessing consequential responsibility.

As we approach a review of philosophical literature in § 3.3, we will find that a formalism geared toward causal responsibility is not only a very attractive way to model from a philosophical perspective, but that the philosophical literature on responsibility has converged on similar definitions of responsibility. Therefore, there appears to be a strong case that causal-first models — aside from their simple structure and readiness to be tied into an agent's decision functions — are a sound way of approaching the problem of framing problems concerning responsibility, because of the consensus across fields which rarely interact.

3.3 Philosophy of Moral Responsibility

Philosophy regarding moral responsibility is an area whose literature is both wide and deep. That said, not all moral responsibility literature is relevant to a computational responsibility project; lots of it is designed from a social analysis perspective which would be difficult to implement in any useful way. Other areas, however, present more promise for studies regarding formalisms.

3.3.1 Peter F. Strawson

One example of research with utility in a computational way is that of Peter F. Strawson, particularly in his seminal essay, Freedom and Resentment [Str62]. Strawson's topic actually revolves around whether determinism has any impact on "free will" — a discussion clearly outside the scope of this project — but in forming his argument creates some key concepts that we can use to consider the applicability of a responsibility formalism to a computational system, as well as touching on what that formalism would look like.

Strawson's fundamental argument can be construed as being that determinism doesn't affect what human factors — like Responsibility and Trust — mean, because these concepts are founded on the relationships between human actors, rather than being inherent to the human actors themselves. As computer scientists, we can extrapolate this argument out to a sociotechnical environment. That is: Strawson's argument applies to both social *and* technological agents within a sociotechnical world. Using Strawson's reasoning, then, we can firmly conclude that it *does* make sense to create "responsible" computers, because responsibility makes sense as a trait for a computer to have in the same way it might a human actor.

Strawson's insights don't stop there, though. He also produces an interestingly rigorous analysis of how ordinarily fuzzy human factors can be formalised appropriately:

> Indignation, disapprobation, like resentment, tend to inhibit or at least to limit our goodwill towards the object of these attitudes, tend to promote an at least partial and temporary withdrawal of goodwill; they do so in proportion as they are strong; and their strength is in general proportioned to what is felt to be the magnitude of the injury and to the degree to which the agents will is identified with, or indifferent to, it.

[Str62]

Strawson captures an essential component of Marsh's computational trust model: an actor's actions influence the perceived trustworthiness of an agent in proportion of the magnitude of the "trustworthiness" of their actions. This concept, Strawson's elucidation shows, is one which can also extend to responsibility; it influences all of the human factors Strawson seeks to address in his essay (i.e. all human factors).

This gives us an insight into how a realistic computational trust model might function: its perception of responsibility should alter depending on external factors. We are aware, however, that people's actions are not the only things which can affect our feeling of responsibility: all sociotechnical factors might. For example, the "Bystander Effect"[FGPF06] occurs when one feels less responsible for acting in an emergency situation because of the number of people present who could help; in the end, nobody does, because every person's sense of responsibility is weakened. However, it is not weakened by any person's actions: it's influenced by a set of sociotechnical factors, of which the feelings that Strawson discusses are a subset.

A suitable conclusion one might draw, then, would be that a valid computational model of responsibility *must* take into account an analysis of the sociotechnical environment of the responsible agent. It would also be valid to draw the conclusion — as a result of the first of Strawson's points discussed — that it makes sense to talk about "trusting" and "responsible" computational agents as if they were humans. Another important conclusion to draw from this part of Strawson's discussion is that this is one way in which agents' interaction is meaningful: therefore, there is a significant body of work to be undertaken in the combination of trust and responsibility formalisms to the field of HCI. Some of this work is already underway[MBEK+11, MW02].

Strawson also notes what sort of agent one rightly considers responsible. His argument relates to which agents one judges responsibility *in*, rather than judging which actions an agent might consider itself responsible for. However, it is pertinent to this research in that it highlights what assumptions the formalism might make about an agent it models the responsibility of. In other words, Strawson helps us to limit the scope of the formalism in terms of the agents it targets.

> Let us consider, then, occasions for resentment ... To the first group belong [agents of whom we can say] "He didn't mean to", "He hadn't realised", "He didn't know" ... "He couldn't help it" ... None of them invites us to suspend towards the agent, either at the time of his action or in general, our ordinary reactive attitudes.
> The second and more important subgroup of cases allows that the circumstances were normal, but presents the agent as psychologically abnormalor as morally undeveloped. The agent was himself; but he is warped or deranged, neurotic or just a child. When we see someone in such a light as this, all our reactive attitudes tend to be profoundly modified.

[Str62]

Strawson here demonstrates a useful separation of two groups of agents: those who incur resentment through a lack of control but who are fully able to act correctly, and those whose lack of control or basic understanding disqualifies them as agents who can incur resentment at all. A definition of resentment would be useful here; unhelpfully, Strawson doesn't lend one, but helpfully, a reasonable general definition of resent is easy to formulate: one feels resentment toward another agent when a goal entrusted to it is not achieved.

In other words, one feels resentment toward an agent which *fails to fulfil a responsibility*. Using this definition, we can use Strawson's separation of resentment-inducing agents to limit the responsibility formalism's scope: an agent can be considered responsible for their actions when they can reasonably be assigned some goal (and possibly some actions to achieve this goal). Another way of saying this would be that an agent can be considered responsible for an action if we can *trust* the agent with a goal; if a goal is assigned, then the act of assignment makes the agent responsible.

Using Strawson's reasoning regarding resentment, then, we can reasonably assert the implication: a responsible agent is an agent actively trusted with a task: a "causal responsibility", to borrow Sommerville's terminology. Along with the earlier notes on types of agents**??**, we can even further limit the scope of agents the formalism should be targeted toward. We also neatly tie into our formalism the assignment of responsibility to an agent; more detail, along with what the assigned responsibility should represent, is discussed in the proposed approach§ 4.

3.3.2 Thomas M. Scanlon

In his essay, Justice, Responsibility, and the Demands of Equality[Sca06], Thomas Scanlon assesses responsibility as having two factors which bear striking resemblance to Sommerville's "Causal" and "Consequential" Responsibilities: "Attributive" and "Substantive" Responsibilities.

Scanlon discusses "Attributive" responsibilities as a group of duties according to the definition:

> What a person sees as a reason for acting, thinking, or feeling a certain way.

For the purposes of later discussion, utility, and slight simplification, I will generalise Scanlon's definition to be "responsibilities for future actions". He also discusses "Substantive" responsibilities, which are loosely defined as responsibilities for the choices an agent has made, taking into account the effects they have and obligations at the time. We can generalise these to be "responsibilities for past actions". While in doing so, I reduce Scanlon's definitions to a discussion of action as opposed to the wider gamut of human properties, this framing sufficiently limits the scope of the work to apply elegantly to decision theory. Therefore, we can begin to apply Scanlon's work to the actions taken by responsible computational agents.

This limitation of scope also enables us to tie Sommerville's sociotechnical research to work done in the philosophical sphere. A complete responsible computational system, therefore, would be suitable for the same scrutiny that human agents currently undergo in the field of moral philosophy. It also, like Sommerville's work, presents us with a framework for thinking about the scope of responsibilities that an artificially intelligent agent might be subject to.

Beyond philosophical waxing lyrical, we have reason to limit our formalism to a subset of responsibilities Sommerville and Scanlon describe. Specifically, we can use Sommerville's "Consequential" or Scanlon's "Substantive" responsibilities to tailor a responsible agent's judgement of its "Causal" or "Attributive" responsibilities. This structure will act as a scaffolding for the simplified proposed framework in § 5.

3.3.3 Aside: Philosophy's impact on Computing Science

It's worth noting that Philosophy and Computing Science are unusual bedfellows. However, this need not be the case. As we've seen already, and will continue to observe, there is a definite need for computing science to make use of and contribute to research involving human factors — whether those factors be sociological, psychological, ethnographic, or even philosophical. What's more, philosophical research may well have an impact on the ethics and cultural implications that certain advances in Computing Science might entail. When discussing matters of potentially existential levels of risk, it is important to have the foresight to construct solid, informed arguments about our role in the grander scope of things.

To this end, I believe collaboration between Philosophy and Computing Science should be a more routine affair. Some interdisciplinary work of this form is already underway: Nick Bostrom's Future of Humanity Institute at Oxford University and the Machine

Intelligence Research Institute in the USA are two excellent examples of Computing Science and Philosophy coming together to produce vitally important work, which greatly influences the progression of the field of AI Safety.

3.4 Comparing Trust and Responsibility

Trust and responsibility are similar concepts. Seen through the lens of C&F, this is particularly clear:

- They both concern themselves with actions and goals
- They're both naturally framed in terms of task delegation
- They both follow Strawson's note on how agents with human factors change their outlook with respect to these factors through interaction.

Trust and Responsibility's similarity regarding task delegation is particularly interesting: this might allow a responsibility formalism to operate in a similar way to trust, meaning that much of the existing literature on Trust could be re-appropriated (with some research) for the purposes of computational responsibility. Indeed, one definition of responsibility for a task might be the obligation to act on a delegated task. A corollary of C&F argue that task delegation is inherently trust; then, all one need do to extend a trust formalism to a responsibility formalism would be to augment C&F's axioms to include an agent's obligation, and one's formalism would be complete!

Unfortunately, C&F's formalism, while technically calculable by a computer, uses logical expressions to evaluate trust. For the purposes of an application such as a decision function of an intelligent agent, unless that agent follows a structure such as a BDI model, this would not be terribly useful — though it's worth noting that granular models of C&F also exist[]. find citation for this

3.5 What work is missing?

4 Proposed Approach

Develop arguments that the responsibility formalism might actually be put to good use, as per § 1

Identify risks in this approach

We can see that the literature available, while not on computational responsibility formalisms, are all fairly related to constructing the formalism proposed. In light of this, we can begin to identify some of the components of a suitable formalism:

- The formalism should answer the research questions laid out in the problem statement§ 2:
  1. How can a computational formalism of responsibility direct the decisions made by an intelligent agent?
  2. How can an intelligent agent assume the consequences of actions it makes, the decisions other agents make, and its general environment, so as to direct its interpretation of responsibility?
- The formalism should suitably limit the scope of the actors it models to be reactive, reflective agents**??**.
  The formalism doesn't apply to all agents; we can imagine some agents which are *not* responsible, such as those shown by Strawson§ 3.3.1, and agents which aren't reactive or reflective.
- The formalism should interpret the obligation an agent is assigned when another agent trusts it with a causal responsibility.
- Ideally, the formalism would utilise as much relevant psychology and sociology literature as possible so as to maximise the formalism's interdisciplinary potential.

4.1 A responsibility formalism's constituent elements

4.1.1 A trust formalism

A useful exercise is to discern what a responsibility formalism would consist of. Worth noting is that this formalism is a proof-of-concept and a starting point for further research to refine. Therefore, a simple model which is easy to implement correctly and reason about should be paramount.

With this in mind, one important constituent part is that of the trust model the formalism works in tandem with. While this could in theory be any model, three notions are worth bearing in mind when selecting one.

1. The trust formalism selected should act as a starting point for the responsibility formalism's design, because of the similarities between trust and responsibility.

2. The trust formalism should express gradations of trust, rather than a boolean logical approach. This is important as the trust an agent has in another agent will allow the former to assess how responsible the latter is; these agents may be required to make judgements of *how* responsible other agents may be, so they might choose one agent to be responsible for a goal over another.

3. The trust formalism should be simple, to act as a basis for the design of a simple responsibility formalism.

To satisfy all of these aims, Marsh's seminal trust formalism seems most appropriate. This is for a number of reasons. For example, Marsh's formalism is very easy to understand and implement — particularly with it being an early formalism uncomplicated by later literature. Another reason is that Marsh's formalism allows one to express gradations of trust; in contrast to another model, such as Castelfranchi & Falcone's model, Marsh's formalism requires no additional complexity to model trust gradation.

Marsh's model is also constructed with other disciplines in mind — psychology and sociology feature prominently in its cited literature. However, Marsh cites no philosophical advancements in his model; therefore, the application of moral responsibility to the formalism being designed cannot be based on similar work used with this formalism.

4.2 Literature influence on the formalism's elements

4.2.1 Formalism designed like Marsh's

The model of responsibility being designed requires the ability to model gradations of trust; therefore, Marsh's Trust formalism will act as the template for the responsibility formalism's design. In particular, Marsh's model's segregation of types of trust are useful for establishing the basic principles behind the responsibility model:

Marsh's Trust	*Responsibility Formalism*
Basic Trust	Basic Responsibility
General Trust	General Responsibility
Specific Trust	Specific Responsibility

Using Marsh's model and adopting the jargon he develops means we can keep lots of the notions he develops, such as variable domains and separation of different "levels" of responsibility, such as how generally responsible an agent is, and how responsible an agent is for a very specific thing. Indeed, a layout of the variables we might use in a formalism of responsibility, compared to Marsh's, might look like this:

Marsh's Trust Model Variables	
Knowledge (of x at time t)	True/False
Importance (of knowing fact x at time t)	$[0, +1]$
Utility (of action αat time t to an agent)	$[-1, +1]$
Basic Trust (of agent A at time t)	$[-1, +1)$
General Trust (of agent A at time t in agent B)	$[-1, +1)$
Situational Trust (of agent A at time t in agent B doing action α)	$[-1, +1)$

Marsh's formalism also makes use of other concepts, such as sets of agents and definitions of what trust might be composed of; similarly, responsibility modelling will require the development of more concrete definitions, as we will see.

4.2.2 Outlining the proposed formalism

An example breakdown of responsibility and what it is might be the following:

Responsibility Term	*Definition of Term*
Agent / Actor	Combination of Constraints, Environment, and Beliefs. Acts on a decision function.
Obligation	Combination of Authority, Goal, Set of Appropriate Actions, Responsibility Score
Responsibility Score	Number in $(0, 1]$ assigned by authority denoting degree of obligation
Action	Combination of Activity, Resource requirements, Possible Issues and Effect
Authority	Agent which assigns an obligation to another agent

Not all terms above are completely defined; however, the table is provided as an example of how a responsibility formalism might look. The formalism proposed here is by no means final, but it can be seen that it would produce gradations of responsibility, that authorities assign an obligation with a certain score, and that agents choose actions based on a decision function — a decision function which takes into account assigned obligations when it produces a next action.

One difference between the above table and a final formalism of responsibility is that the final formalism would also account for processes. For example, a number assigned by an Authority represents the degree to which the responsible agent is obliged to act toward a goal, but the agent itself needs to weigh this up against other factors. For example: might the goal be time sensitive? It's imperative that I pay my rent on time, but a responsibility to keep windows closed doesn't depend on any one point in time. In other words, regardless of the initial score assigned to paying rent, I (as an agent) must interpret that score while taking into account other features of the goal — these features change from goal to goal, and theoretically from agent to agent, too. This *interpretation function* is a necessary part of the formalism, too — but it is defined by process, and not semantically.

4.3 In answering research questions

In stating the problem proposed, two research questions were devised:

1. How can a computational formalism of responsibility direct the decisions made by an intelligent agent?
2. How can an intelligent agent assume the consequences of actions it makes, the decisions other agents make, and its general environment, so as to direct its interpretation of responsibility?

The first research question can be answered in part by the definition of the Agent / Actor's decision function making use of the responsibility formalism in choosing the agent's next actions. To show this, artificial agents will be constructed with decision functions which are guided by the agent's assigned responsibilities.

The second research question can be answered with similar methods. To construct artificial agents with responsibility-guided decision functions, a full responsibility formalism will be devised and implemented in these artificial agents. In doing this, the formalism will work in tandem with the interpretation functions of the agents developed. In doing so, the interpretation function developed will show that a complete implementation of the responsibility formalism will answer the second research question.

It is worth noting that the interpretation function which answers the second research question is a necessary component of the formalism, but no one interpretation function belongs in the responsibility formalism itself. This is because, while a function can be imagined which has certain properties, two different agents may require different interpretation functions. In other words, the interpretation is a separate concern from the formalism itself: the formalism might specify *domains* for the interpretation function to map from and to, but cannot specify specific processes and parameters. Therefore, no canonical version of the interpretation function may be provided, though some properties of the function will be noted and incorporated into the formalism's definition.

5 Work Plan

My strategy for completing the formalism outlined is composed of four milestones, with concrete and well-defined deliverables:

1. Complete the formalism
   This will see a full formalism developed, complete with definitions.
2. Design a test case
   The product of this stage will be a scenario where the formalism can be fairly tested, so as to answer the research questions as laid out in § 4.3.
3. Test the formalism
   This will see intelligent agents developed and tested in the example scenario. It will produce experimental data, and may result in minor modifications to the formalism so as to properly simulate responsibility.
4. Write-up
   With experimental data collected and evaluated, the final report will be written up.

5.1 Completing the Formalism

While some detail regarding the formalism and its structure has already been surmised, the full extent of the formalism is incomplete.

One aspect of the formalism which has not been developed is the mathematical reasoning which turns the semantic descriptions of the formalism into one which is fully algorithmic. The mathematical expressions to be devised, while being vital for the completion of the formalism, allow the artificial agents to assess the impact of their actions and reason about the obligations they have been given, weighing them up and having their assessment of their obligation change over time. This is necessary for answering both the first and second research questions.

Another aspect of the formalism yet to be completed is that some definitions of components of the formalism are yet incomplete. Some of these definitions are somewhat trivial, such as what precisely a goal is composed of — these definitions are a necessary component of the formalism, but are unlikely to have a major impact on its structure. Other components, such as the domains the interpretation function maps to and from, may require a greater degree of insight to finalise.

Some of the formalism still to be defined, such as the interaction between an agent and an obliging authority, may require philosophical reasoning and the creation of arguments of moral responsibility to fully develop. Philosophical work has already impacted

the development of this formalism, and the Philosophy department at Glasgow University has kindly helped in pointing these components of the formalism in the correct direction. Further collaboration and ethical reasoning will continue in order to finish this part of the formalism.

As a result of the multi-faceted and sometimes interdisciplinary nature of the work yet to be completed with regards finishing the formalism's finer details, a large portion of time has been allocated to ensuring that the details are correct. While testing the formalism may alter some of the definitions produced in minor ways, the intention is to create as complete and well-reasoned a formalism at these early stages as possible, so as to ensure the development of an appropriate test case. A risk in the research planned is that the test case developed does not properly test the responsibility formalism; to counter this concern, the formalism will be engineered to be as complete as possible in as early a stage of the work as possible.

It is intended that this be completed by mid-January.

5.2 Designing the Use Case

The use case required to test the formalism is particularly important, as the somewhat semantic nature of the research may make data collection and analysis difficult. With prior work developing similar formalisms for fields such as trust formalisms, however, this component of the research is not expected to become a time-consuming aspect of the work. Unlike the development of the formalism itself, there are examples which can fairly heavily inspire the experimental design of the project. However, this does not mean that there is no work to be done!

An example of the work to be completed here is that a scenario must be chosen to be modelled. For example, were one performing research on workflow modelling, one would select a suitable real-world situation which can be broken down into a workflow and modelled appropriately — an example might be the workflow a developer goes through in producing a new commit for a programming project in a VCS. For this responsibility formalism, a real-world scenario where responsibility affects one's decision making is required; this scenario should also act as an example of how an agent's feeling of responsibility regarding an obligation they have been assigned changes over time, so that this too might be modelled so as to test Research Question 2 (§ 4.3). The scenario chosen should also be tailored to exhibit definitions and other aspects of the formalism which were determined in the previous segment of work.

An AI strategy should also be devised during this stage. While this work is not strictly artificial intelligence-specific, testing it will require artificial agents to be developed. In keeping with the research performed by Marsh[Mar94], these agents will likely be reinforcement learning agents, as this permits learning behaviour in the artificial agents without a particularly complex implementation., these agents will likely be reinforcement learning agents, as these can prove to be simple to implement while still exhibit-

ing suitable learning behaviour. Decisions as to algorithms to implement for testing purposes will be determined in this stage.

Along similar lines to the design of the intelligent agents' learning algorithms, the agents themselves will need to be designed to appropriately simulate responsibility using the formalism. This will require design of the agents in tandem with the example scenario, as well as determining the nature of the interpretation function used by the agents in assessing their obligations.

A nuance of this segment of work is that the experiment being developed must ultimately produce some data to analyse. Therefore, metrics which are appropriate for the analysis of an artificial agent's behaviour — and how responsibly that agent is behaving — should be produced. This will be used to assess the formalism's efficacy in the next part of the work plan.

This section of work, being smaller than the previous segment and building on the advancements made in the finalising of the definitions, should not be as time consuming as previous segment. Therefore, this is expected to be completed around the beginning of February.

5.3 Testing the Formalism

Once the test case has been identified and necessary aspects of the experiment are established, the example scenario will be developed using agent modelling tools appropriate for the task at hand. The agents developed must be designed to make use of the responsibility formalism, and should exhibit some characteristics:

- Agents should be able to effectively discard actions they no longer assess themselves as "responsible" for. A real-world example of this behaviour might be if a person asks for coffee to be ordered for them, but then immediately order their own coffee: the sociotechnical environment is such that a human agent in this scenario, while aware that they have still been asked to order coffee, can see that the goal of ordering coffee has been fulfilled by the authority of the obligation already.
- Agents should be able to act in increasingly more responsible ways as they learn their effects on the environment and the world around them. This is necessary for answering Research Question 2 (§ 4.3).

Experimental data will be produced by assessing the actions chosen by the agents according to metrics specified in the previous segment of work. This data will then be analysed, so as to ascertain whether the formalism must be tweaked at all: any changes to the formalism which become apparent upon implementation will be made here, and the experiments will be carried out a second time should this be required. Another noteworthy component of this segment of work is that the analysis done on the data may

take some time, given that the research does not lend itself as easily to experimental analysis as some other fields (such as algorithmic complexity analysis) might.

Due to the need to write code and assess that the model produced is as bug-free as can be ensured, this segment is expected to take a significant amount of time. Should the previous block of planned work be completed around the beginning of February, this implementation is expected to be completed toward the end of February. If so, the analysis of the data produced should end this block of work around the middle of March.

5.4 Write-up

Once experimental data has been collected, verified, and analysed, the final block of work is to finish the write-up of the project. While this report will be developed in small parts as the work goes on, the bulk of the writing — as well as insight and introspection as to the conclusions of the work, and the next steps which it offers — will be developed and produced. It is expected that, should the rest of the work be properly completed at this point, that the report be finished around the beginning of April.

5.5 Review of time allocation

The time allocated to the various components of the proposed work is as follows:

Make a Gantt Chart!

Section	*Intended completion time*
Finalise Formalism	Mid-January
Design Use Case	Beginning of February
Run Tests and Collect Data	Early/Mid-March
Write-up Completed	End March

Note that the completion of the work ends three weeks before the project is due. To avoid complications and rushed research, the work has been planned with a three-week margin of time should any part overrun. This is particularly handy in the case that testing the formalism introduces minor changes, or other unexpected delays. It is hoped that this margin results in the entirety of the research having the amount of time available to complete the work *properly*, rather than rushing later stages such as experimentation as a result of some unintended issue.

Todo Notes

References

[Bir] G D Birkhoff. AESTHETIC MEASURE.

[CE11] Partheeban Chandrasekaran and Babak Esfandiari. A model for a testbed for evaluating reputation systems. *Proc. 10th IEEE Int. Conf. on Trust, Security and Privacy in Computing and Communications, TrustCom 2011, 8th IEEE Int. Conf. on Embedded Software and Systems, ICESS 2011, 6th Int. Conf. on FCST 2011*, pages 296–303, 2011.

[CF] Cristiano Castelfranchi and Rino Falcone. Social Trust: A Cognitive Approach.

[FGPF06] Peter Fischer, Tobias Greitemeyer, Fabian Pollozek, and Dieter Frey. The unresponsive bystander: are bystanders more responsive in dangerous emergencies? *European Journal of Social Psychology*, 36(2):267–278, 2006.

[HJS06] Trung Dong Huynh, Nicholas R. Jennings, and Nigel R. Shadbolt. An integrated trust and reputation model for open multi-agent systems. *Autonomous Agents and Multi-Agent Systems*, 2006.

[Hon] Ted Honderich. Free will, determinism and moral responsibility – the whole thing in brief.

[LSSB10] Russell Lock, Tim Storer, Ian Sommerville, and Gordon Baxter. Responsibility modelling for risk analysis. 2010.

[Mar94] Stephen Paul Marsh. Formalising Trust as a Computational Concept. *Computing*, Doctor of(April):184, 1994.

[MBEK+11] Stephen Marsh, Pamela Briggs, Khalil El-Khatib, Babak Esfandiari, and John A. Stewart. Defining and Investigating Device Comfort. *Journal of Information Processing*, 19(7):231–252, 2011.

[MW02] Michael W. Macy and Robert Willer. From Factors to Factors: Computational Sociology and Agent-Based Modeling. *Annual Review of Sociology*, 28(1):143–166, 2002.

[PMB05] Andrew Patrick, Stephen Marsh, and P Briggs. Designing systems that people will trust. *Security and Usability: Designing Secure Systems That People Can Use*, pages 75–100, 2005.

[Sca06] Thomas M Scanlon. Justice, responsibility, and the demands of equality. 2006.

[SFYA15] Nate Soares, Benja Fallenstein, Eliezer Yudkowsky, and Stuart Armstrong. Corrigibility. *AAAI Workshop on AI and Ethics*, (2014):74–82, 2015.

[Slo84] Aaron Sloman. The Structure of the Space of Possible Minds. pages 35–42, 1984.

[Som07] Ian Sommerville. Models for responsibility assignment. In *Responsibility and dependable systems*, pages 165–186. Springer, 2007.

[SS15] Robbie Simpson and Tim Storer. Formalising responsibility modelling for automatic analysis. In *Lecture Notes in Business Information Processing*, 2015.

[Str62] Peter F. Strawson. Freedom and resentment. *Proceedings of the British Academy*, 48:1–25, 1962.