Structuring Safety Policy Decomposition

Martin Hall-May and Tim Kelly

Department of Computer Science
University of York, York, YO10 5DD, UK
{martin.hall-may, tim.kelly}@cs.york.ac.uk

Abstract. Safety policy is a collection of rules that govern the behaviour of entities such that they do not cause accidents. It has been suggested that policies in general can be expressed at various levels of abstraction and organised as a hierarchy of goals. In developing policy, it is desirable to decompose from top-level objectives down to rules in a structured manner. The Goal Structuring Notation (GSN) allows us to model the policy decomposition in order to scrutinise and better understand the development process. In so doing, a number of issues arise concerning reusable patterns of decomposition and the assumed models of the system whose behaviour the policy is intended to govern. This paper discusses the need to structure a safety policy decomposition and how modelling techniques and patterns can aid in this.

1 Introduction

Complex networks of interacting entities, such as systems of systems [1], multi-agent systems and organisations of individuals, are governed by rules, procedures, laws, codes and conventions. Irrespective of the nomenclature, these regulations can be seen as forming a *policy* that governs the behaviour of the entities interacting as part of a larger network. This policy is *orthogonal* to the immediate aims and objectives of the entities and restricts their actions such that they do not engage in undesired behaviour. The policy is, as such, *persistent*, in that it is relatively invariant over a period of time [2]. Developing such a policy is a significant challenge.

Policy can be formulated according to any of several criteria. For example, a safety policy describes how to protect the physical integrity of a system, a security policy describes how to protect data integrity within a system, while a usage policy describes the rights and privileges of the users of a system. This paper focuses on the concept of a safety policy.

2 Safety Policy

Policy is defined variously in literature, but the most generally applicable system-oriented definition is given in [3]: "A policy is a rule that defines a choice in behaviour of a system." Safety policy is defined as the choice in behaviour that does not contribute to a hazardous situation, or through action actively mitigates a hazard.

Government and organisational policy, among others, are mainly described in lengthy prose, often ambiguous and open to (mis-)interpretation. Wies, in [4], illustrates the problem by noting the confused nature of many (security) policy specifications, which combine in the same document high-level statements concerning network accessibility with low-level statements about the blocking of certain IP addresses.

To take another example, the Rules of the Air [5] describe the policy that a pilot must follow in order to fly safely in UK civil airspace. The resulting document describes a set of rules that are often obvious, but the rationale behind which has been lost. This has implications for *traceability* and *change management*, should the original reason behind the policy rules change. Furthermore, hiding the rationale in such a way affects the ability to scrutinise or analyse the policy and thereby gain confidence in its *completeness* and *consistency*.

2.1 Policy Decomposition

There is an increasing desire to manage complex networks of autonomous systems using policy. Humans may be adept at interpreting informally expressed policies, but their interpretations are rarely consistent and can not be easily automated given the current state of technology [6]. An analogy has been drawn to creating computer code from an abstract set of requirements [7]. Clearly there is a need to organise, or classify, policies expressed at various levels of abstraction into a hierarchy.

Policy decomposition refers to the transformation of high-level policy specifications into more specific policies that are defined in terms of lower-level entities and operations of the system [8]. This decomposition is necessary because policies are more naturally expressed, at least initially, at a high level, since they are typically derived from management — or business — goals. Such a level of abstraction hides system specifics, so that policy-makers are not faced with excessive detail. Moreover, high-level policy is not constrained to any particular underlying resources and will therefore not break in the face of a change in said resources. Indeed, it may not at first be known what target resources are necessary or available.

The Goal Structuring Notation (GSN) [9] — typically used to construct safety cases — is used here to represent the policy decomposition structure. Figure 1 shows an example excerpt of such a policy decomposition modelled using this notation. Rectangular nodes represent goals, while parallelograms indicate the strategy by which a policy goal is decomposed to an increased level of specificity. Goals and strategies are expressed in a context, which is indicated by rounded rectangles. In representing the policy in this manner, we are bringing the process by which it is derived under increased scrutiny, which raises several issues about its development. Since there are often several ways to decompose any policy goal, the policy is not an unambiguous refinement, rather it attempts to provide *qualitative justification* for the decomposition of goals to rules. Two aspects of the decomposition process are important to the confidence in this justification, namely the use of models in system assumptions and common patterns of decomposition. These will be the focus of the remainder of the paper.

3 Modelling

The informal role of models in policy-making is well established [10]. Theoretical as well as empirical models are used by Government and other decision-makers to set

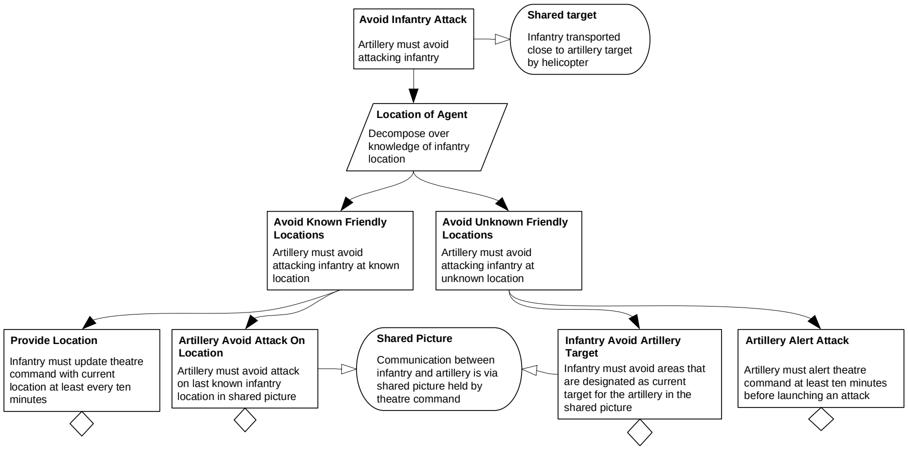
Avoid Infantry Attack
Artillery must avoid attacking infantry
Shared target
Infantry transported close to artillery target by helicopter
Location of Agent
Decompose over knowledge of infantry location
Avoid Known Friendly Locations
Artillery must avoid attacking infantry at known location
Avoid Unknown Friendly Locations
Artillery must avoid attacking infantry at unknown location
Provide Location
Infantry must update theatre command with current location at least every ten minutes
Artillery Avoid Attack On Location
Artillery must avoid attack on last known infantry location in shared picture
Shared Picture
Communication between infantry and artillery is via shared picture held by theatre command
Infantry Avoid Artillery Target
Infantry must avoid areas that are designated as current target for the artillery in the shared picture
Artillery Alert Attack
Artillery must alert theatre command at least ten minutes before launching an attack

Fig. 1. An Excerpt from a Safety Policy Decomposition in GSN

policies on a number of issues, ranging from health and the economy to the environment. Indeed, the defence industry uses models to guide combat decisions based on their knowledge, assumptions and best guesses of enemy capability, the anticipated operational environment as well as the configuration and inter-operation of their own forces.

Models serve two purposes in policy decomposition. Firstly, they aid in decomposing safety goals, together with patterns of decomposition, by providing the policy-maker with factors that should be considered in the achievement of the top-level goal. Secondly, the models provide a vocabulary for the expression of these goals. In this way, templates can be created that are more structured than the "noun-phrase verb-phrase" of traditional safety case goal statements.

In the decomposition of safety policy, models help us to understand the abstract features of hazards and to mitigate them through a general type of policy rather than applying a sticking plaster to every specific instance of the hazard, which quickly becomes unwieldy and inconsistent, leaving *loop holes* and areas of overlap or conflict. We suggest modelling the system under consideration according to three viewpoints, viz. an agent, a domain and a causal viewpoint.

3.1 Agent Viewpoint

Taking inspiration from multi-agent systems development, it is important for the development of safety policy to have a good understanding of the capabilities of, and communications between, the entities in the system. Using a suitable methodology, such as the Process for Agent Societies Specification and Implementation (PASSI) [11], several models of the agent society can be generated. These include describing how the entire required system functionality is apportioned to individual agents according to their specialisations and capabilities, and the modelling of envisaged scenarios of interactions. This is important in order to consider the types of communications that will occur as well as which agent relies on the services or knowledge of another. Hazards can arise as a result of the incorrect provision of such services, provision when not expected, or absence of provision.

3.2 Domain Viewpoint

In understanding how an agent might misinterpret something we need to know additional properties about the domain in which it operates. This includes the ontology it uses, i.e. how it represents knowledge of what exists, and assumptions about the properties of these concepts. In ontologies, e.g. Cyc [12], predicates define how concepts are related and which actions can be performed on certain concepts. For example, an agent might know that a *pilot* flies an *aeroplane*, which is a kind of *fixed-wing aircraft* and which is capable of being *airborne*. Similarly, the absence of such a predicate would indicate that a pilot cannot fly a tank. The unambiguous representation of such knowledge is critical in safety-related applications, since any misinterpretation in communications from one agent to another can lead to hazards.

3.3 Causal Viewpoint

The causal viewpoint recognises that accidents in complex systems arise out of, as Perrow described, dysfunctional interactions [13]. The accident model, STAMP [14], recognises the importance of this type of interaction in safety-critical applications. Similarly, we must take a more systems-theoretic approach to describing the relationships between causal factors in the lead up to an accident, rather than traditional chain-of-event failure models such as Fault Trees. Behaviour typical of complex systems results from multiple interacting feedback loops. This means that it is not possible to take a mechanical approach to working through the causal chain, because many factors influence each other as well as, indirectly, influencing themselves.

Multi-agent Influence Diagrams (MAIDs) [15] are an extension to Bayesian belief networks and decision networks, which are suited to describing processes composed of locally interacting components. Using them it is possible to represent how agents' decisions are influenced by various factors. These factors include probabilistic variables (represented by circular 'chance' nodes) that are, in effect, 'decided' by the environment, as well as the results of other agents' decisions (rectangular nodes). The 'utility' of the decisions to the agents, i.e. their preference for the result of a particular decision in terms of the cost or benefit to them, is represented by diamond-shaped nodes.

Figure 2 gives a simplified example of two agents' decisions. The artillery's decision to launch an attack is based on the decision of an unmanned air vehicle (UAV) spotting an enemy target. In reality, the artillery cannot directly observe the UAV's decision, hence its own action is influenced by many other chance variables, such as the accuracy of the UAV's sensor and the state of the communications network. Models of this type are important in order to consider which variables in the system influence which other variables and whether these variables are determined by chance or are under the control of an intelligent agent.

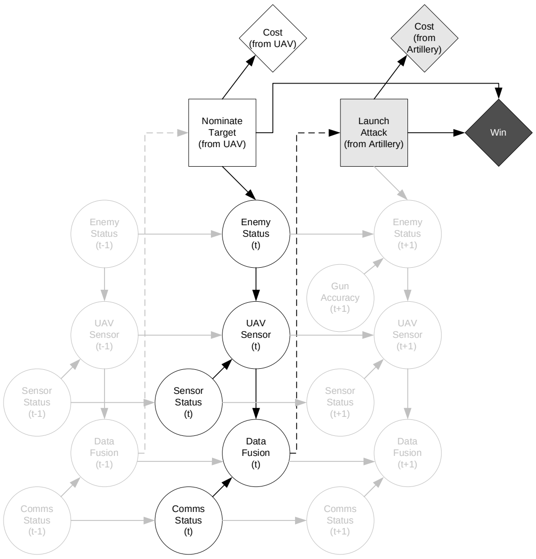

Fig. 2. A Multi-agent Influence Diagram Representing Two Agents' Decisions

4 Patterns

Often in the development of policy, the same type of decomposition is used repeatedly. This leads us naturally to think about structuring policy around patterns of decomposition. Patterns can be based on considering agent capabilities, cooperation between agents, milestones in a sequence of tasks that an agent must carry out to achieve its objective, as well decomposing according to the types of case in which the policy must apply. Unfortunately, space constraints do not allow a more detailed treatment of this issue, however we would direct the interested reader to a previous paper [16].

5 Summary

A safety policy defines the set of rules that governs the safe interaction of a society of entities. In practice, policies are expressed at various levels of abstraction and can be modelled using GSN as a structure of increasingly specific goals. These goals are decomposed according to strategies. However, the decomposition process is not obvious and relies on the repeated application of patterns and the use of system modelling.

6 Acknowledgement

This work is carried out under the High Integrity Real Time Systems Defence and Aerospace Research Partnership (HIRTS DARP), funded by the MoD, DTI and EPSRC. The current members of the HIRTS DARP are BAE SYSTEMS, Rolls-Royce plc, QinetiQ and the University of York.

References

1. Alexander, R., Hall-May, M., Kelly, T.: Characteristic failure modes in systems of systems. In: Proceedings of the 22nd International System Safety Conference, Providence, Rhode Island, System Safety Society (2004) 499–508
2. Moffett, J.D., Sloman, M.S.: The representation of policies as system objects. In: Proceedings of the Conference on Organizational Computing Systems, Atlanta, Georgia, USA, ACM Press (1991) 171–184
3. Damianou, N., Dulay, N., Lupu, E., Sloman, M.: Managing security in object-based distributed systems using Ponder. In: Proceedings of the 6th Open European Summer School (Eunice 2000), Twente University Press (2000)
4. Wies, R.: Using a classification of management policies for policy specification and policy transformation. In Sethi, A.S., Raynaud, Y., Fure-Vincent, F., eds.: Proceedings of the IFIP/IEEE International Symposium on Integrated Network Management. Volume 4., Santa Barbara, California, USA, Chapman & Hall (1995) 44–56
5. Allan, R., ed.: Air Navigation: The Order and the Regulations. third edn. Civil Aviation Authority (2003)
6. Moffett, J.D., Sloman, M.S.: Policy hierarchies for distributed systems management. IEEE Journal on Selected Areas in Communication **11**(9) (1993) 1404–1414
7. Sloman, M.: Policy driven management for distributed systems. Journal of Network and Systems Management **2**(4) (1994) 333–360
8. Bandara, A., Lupu, E., Russo, A.: Using event calculus to formalise policy specification and analysis. In: Proceedings of the 4th IEEE Workshop on Policies for Distributed Systems and Networks, Lake Como, Italy (2003) 26–39
9. Kelly, T.P.: Arguing Safety—A Systematic Approach to Managing Safety Cases. DPhil thesis, University of York, Heslington, York, YO10 5DD, UK (1998)
10. Weinstein, M.C., Toy, E.L., Sandberg, E.A., Neumann, P.J., Evans, J.S., Kuntz, K.M., Graham, J.D., Hammitt, J.K.: Modeling for health care and other policy decisions: Uses, roles, and validity. Value Health **4**(5) (2001) 348–61
11. Cossentino, M., Potts, C.: PASSI: A process for specifying and implementing multi-agent systems using UML. (2002)
12. Guha, R.V., Lenat, D.B.: Cyc: A midterm report. AI Magazine **11** (1990) 32–59
13. Perrow, C.: Normal Accidents: Living with High-Risk Technologies. Princeton University Press (1999)
14. Leveson, N.G.: A new accident model for engineering safer systems. Safety Science **42**(4) (2004)
15. Koller, D., Milch, B.: Structured models for multi-agent interactions. In: Proceedings of the 8th conference on Theoretical Aspects of Rationality and Knowledge, Siena, Italy, Morgan Kaufmann Publishers Inc. (2001) 233–248
16. Hall-May, M., Kelly, T.P.: Defining and decomposing safety policy for systems of systems. In: Proceedings of the 24th International Conference on Computer Safety, Reliability and Security (SAFECOMP ’05). Volume 3688 of LNCS., Fredrikstad, Norway (2005) 37–51