INTERNATIONAL CONFERENCE ON ENGINEERING DESIGN, ICED15
27-30 JULY 2015, POLITECNICO DI MILANO, ITALY

AN AGENT-BASED APPROACH TO SUPPORT PLANNING FOR CHANGE DURING EARLY DESIGN

Fernandes, João; Henriques, Elsa; Silva, Arlindo; Pimentel, César
Instituto Superior Técnico, Portugal

Abstract

Early design is the most dynamic and unpredictable stage of complex design processes, since it involves a great deal of uncertainty, collaborative iteration and adaptive organizational behaviour. This paper argues that current activity-based modelling approaches have limited ability to capture the dynamics of early design and explores novel modelling approaches to support planning during this stage. The development of an Agent Model for Planning and rEsearch of eaRly dEsign (AMPERE) aiming to support early design planning is described. The initial results from agent-based simulations are presented reporting an investigation to the likely effects of requirements change in global design process performance.

Keywords: Early design phases, Process modelling, Planning for change, Agent-based simulation

Contact:
Dr. Joao Fernandes
Instituto Superior Tecnico
Mechanical Engineering
Portugal
joao.ventura.fernandes@tecnico.ulisboa.pt

Please cite this paper as:
Surnames, Initials: *Title of paper*. In: Proceedings of the 20th International Conference on Engineering Design (ICED15), Vol. nn: Title of Volume, Milan, Italy, 27.-30.07.2015

1 INTRODUCTION

Among the different stages of the design and development process of large technical systems, the early design stage, normally referred to as *preliminary design*, exhibits particularly complex dynamics. Previous research performed to aircraft jet engine design, for instance, reported that it is typical that early design phases involve regular interactions between the customer of the complex system and the supplier organization(s) capable of providing a design proposal (Fernandes et al., 2014). Once a request for a proposal is provided, the potential supplier's designer teams work together to synthesize a design solution that can meet the customer's requirements and expectations.

Previous work showed that early jet engine design processes were characterized by rapid design iterations of exploration, convergence and solution refinement, concurrent design activities involving system, sub-system and component designers and large amounts of collaborative work through interactions between design teams and different domain experts intending to resolve conflicting goals and perform solution trade-offs (Fernandes et al., 2014). Furthermore, previous empirical studies showed also that early design typically involves large uncertainty levels arising from frequent high-level requirements change that affect activity realization (Fernandes et al., 2015). Large uncertainties trigger adaptive behaviours in design actors which often need to balance technical risk and the time available to deliver the design proposal to the customer.

Following the previous empirical research, this paper begins by discussing current limitations in modelling approaches to capture these dynamics of early stages of complex design. Aiming to address these limitations, an agent-based approach is subsequently presented to support planning during early design through the estimation of the likely project performance considering the effects of expected levels of change.

2 MODELLING THE EARLY STAGES OF COMPLEX DESIGN

The central purpose of complex design process modelling and simulation is to provide planning support to organizations. Among the existing modelling approaches, *activity-based models* have been the most widely explored approach both by academia and in industrial practice (Browning and Ramasesh, 2007).

Activity-based models view the design process as an "information processing system" (Wynn, 2007) and are based on a network representation of the design process. A set of activity-based models deal with *precedence* relationships between the design tasks, such as PERT (Wiest, 1969), GERT (Pritsker, 1966), Petri Nets (Murata, 1989), Signal Flow Graphs (Eppinger et al., 1997) and Applied Signposting (Wynn et al., 2006). These models typically represent the design process as a pre-determined network of activities, capturing the flow of information according to the chronological order normally followed. Other activity-based models are based on *dependency* relationships between tasks, incorporating the coupling of information but not pre-determining the entire process flow. Dependency-based models such as the Design Structure Matrix (Steward, 1965) and the Domain Mapping Matrix (Danilovic and Sandkull, 2005) rely on storing the interdependencies in a matrix-like form and using algorithms to identify structural patterns (Browning, 2001), searching for improvements (Dong, 2002) or performing process simulation (Cho and Eppinger, 2001).

The key strength of activity-based models is its cost-effectiveness in capturing moderate size and well-structured processes, due to its intuitive graphical notation based on the node-arc or matrix representation of the process network. Because of that, they have been widely explored for design process visualization, planning and execution control (Browning and Ramasesh, 2007), particularly during design stages where the sequence of tasks and patterns of iteration are well defined, such as in the detailed design stages.

However, there are several important limitations in activity-based models when the goal is capturing the dynamics of early stages of complex design. Firstly, their rigid structure limits the representation of loosely defined processes with iterative cycles involving multiple design teams and knowledge disciplines. Early design involves concurrent and frequent interactions between teams, which are difficult to incorporate in activity-based models without adding a large number of decision nodes. This causes that these models become intractable to plan and to communicate the process flow. Secondly, activity-based models normally rely on centralized discrete-event engines which are not designed to handle unscheduled events. This prevents the capture of asynchronous information exchange between teams, which often drives highly distributed and collaborative process flows typical of early design.

Thirdly, adaptive behaviour is difficult to incorporate since these models rely on a "mechanistic" view of the design process (Wynn et al., 2007).

3 AN AGENT MODEL OF EARLY DESIGN

Based on the previous analysis of strengths and limitations of activity-based approaches, this paper explores the potential of *agent-based models* to capture the dynamics of early design and their capabilities for representing key early design characteristics such as uncertainty, iteration, collaboration and adaption. An Agent Model for Planning and rEsearch of eaRly dEsign (AMPERE) was developed to integrate these different facets of complex early design and it was applied in the investigation of the dynamic effects of changes in requirements in the overall project performance. The goal of the current development was also demonstrating the capability of agent-based models to provide early design planning support.

3.1 Architectural design

AMPERE was designed taking advantage of the capabilities embedded in the Smart Phython multi-Agent Development Environment (SPADE) created by Gregori et al. (2006). SPADE provides an agent management system and an agent communication channel which allows the agents to communicate using FIPA-ACL performatives (Gregori et al., 2006). The SPADE agent class supports agents with behaviours, including cyclic and periodic behaviours for repetitive actions, one-shot and time-out behaviours for casual actions, the finite state machine behaviour for internal state transitions, and the event behaviour for actions in response to some event that the agent has perceived.

Building upon this framework, AMPERE includes specialized agents inheriting from the parent SPADE agent. One child is the *Design Agent*, which was designed as a practical reasoning agent (Wooldridge and Jennings, 1995) based on an internal *Belief-Desire-Intention* (BDI) agency structure (Bratman et al., 1998). This agent incorporates a cyclic reasoning algorithm supported on methods for observing the environment, generating and filtering options, planning and executing the chosen action and updating the agent's beliefs.

The Design Agent does planning through searching into an internal task library for a *Task* object with a post-condition matching the selected desire during practical reasoning. Once a Task object is selected, its body method is executed resulting in an action performed on the environment, and the agent remains busy during the time assigned to the task's duration attribute. In addition, the execution of task objects representing actual design activities enables a Design Agent to operate and change the status of a design *Solution* object. The solution's quality is one of the key attributes of the Solution class, enabling a parameterization of the optimization level achieved through the Design Agent's efforts.

3.2 Agent definition and behaviours

Four agent sub-types have been defined in AMPERE inheriting the generic Design Agent architecture: the *Customer* agent; the project *Lead* agent; the *Senior Designer* agent; and the *Junior Designer* agent.

The Customer Agent incorporates a client entity with privileged access to the market environment, and observes how needs evolve over time, reacts to events of change and is motivated to send requests for a design proposal to a supplier organization (Figure 1). The Customer is also prone to trigger an update of requirements when market needs have changed.

Conversely, market information is inaccessible to agents belonging to the supplier organization, but the Lead, Senior Designer and Junior Designer agents are keen to respond to requests and updates arriving from the Customer. Replicating the type of hierarchical unit found in large organizations, the Lead, Senior Designer and Junior Designer agents are the basic building blocks of a *design team* (Figure 1). The Lead agent has the ability to dispatch *directions* to designer agents in the team, which are normally predisposed to accept them. The supplier organization can be composed of one or more design teams working together.

The distinction between Senior and Junior Designers captures the distinct levels of experience often found in design projects, which results in different impact of actions performed. Senior Designers represent elements that have completed many projects and normally work across multiple projects in a collaborative manner during short periods, but with the ability to strongly influence and direct the course of the design solution. Conversely, the Junior Designer represents elements far less experienced, fully committed to one or a few projects and of much lower resource cost.

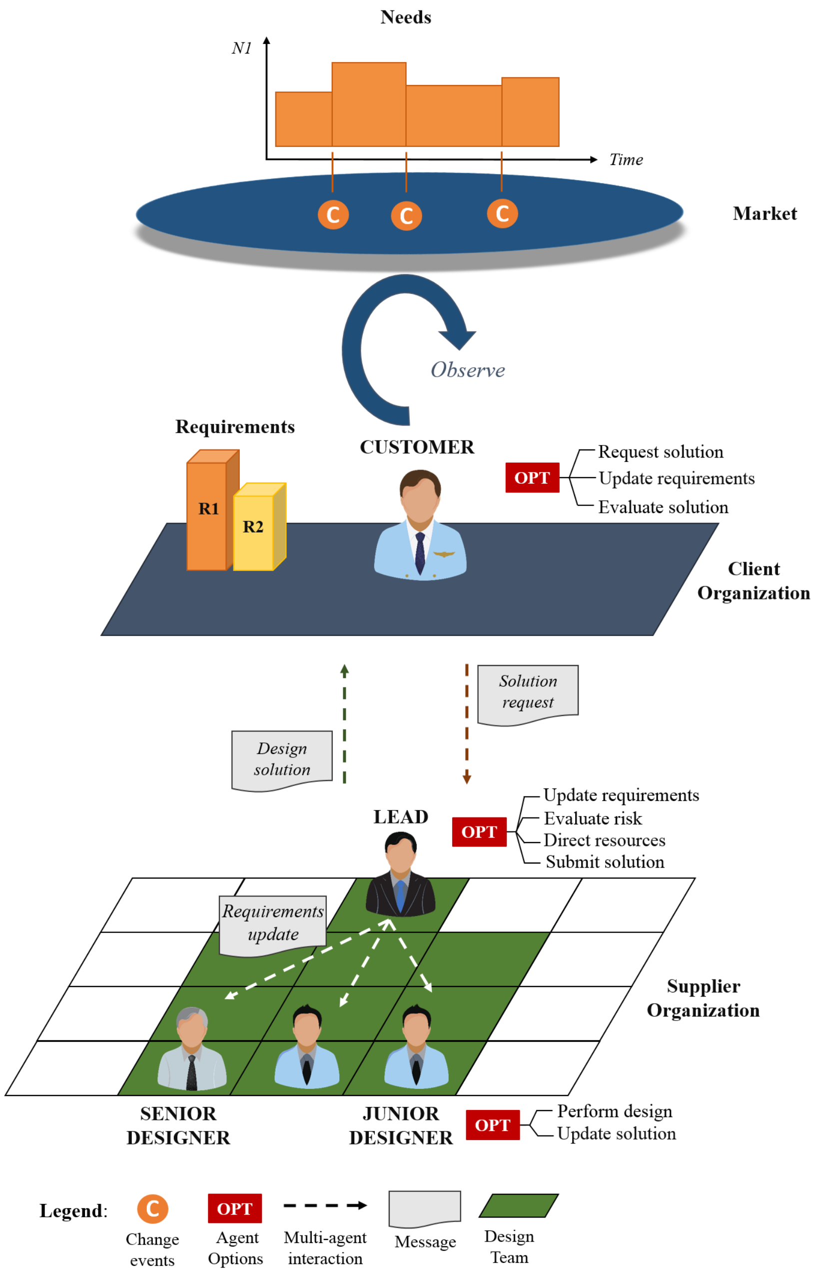

Figure 1. Overview of the agent definition and behaviours in AMPERE.

Each agent is modelled in AMPERE with the ability to perform a finite number of actions on the environment, including interactions with other agents. For instance, Figure 1 illustrates that the Customer may request a design proposal to the supplier, update requirements or evaluate the solution received, while the project Lead is able to request a solution from the team, update arriving requirements, evaluate project risks, direct resources and submit a solution to the Customer. Designer Agents can either perform design activities or update the design solution to its team (Figure 1). Each agent selects the action to perform on the environment according to its beliefs' status. For instance, awareness that the market has changed triggers the Customer's desire to update requirements to the supplier. The Lead's knowledge of the arrival of a new request for a solution causes the agent to implement several actions, namely a solution request to the design team, risk evaluation and adjustment of the team's resources according to the perceived risk level. Designers can either perform design activities or update the design solution to its team. The realization and repetition of the design tasks allows designer agents to generate or improve the status of the design solution's quality until requirements are met, similarly to what is observed in practice. Furthermore, Senior Designers may be chosen to collaborate with Junior Designers belonging to the team, allowing a faster convergence to the requirements (faster design iteration) or a better design solution, or support other projects, in the sense that they are not dedicated to a single project.

3.3 Modelling uncertainty and iteration

When Junior and Senior Designers plan actions representing actual design activities, the execution enables the generation of a design solution instance which becomes available for operation after creation. While the solution's quality is lower than required, meaning that requirements have not yet been achieved, the agents may choose to iterate the design activities. The effect of the iteration on the agent's design Solution instance quality attribute is modelled in AMPERE according to Equation 1:

$$Q(n) = Qs - (Qs - Qi)e^{-\alpha n} \tag{1}$$

where Q is the solution instance quality attribute; n is the number of accumulated iterations; Qs is the standard quality level the agent is able to achieve; Qi is the initial quality level; and α is the quality progress rate coefficient. During simulation, the quality attribute takes real values between [0; 1], where a unitary value represents a fully optimized design solution. Similarly, the effect of iteration on the agent's design task instance duration is modelled according to Equation 2:

$$D(n) = Ds + (Di - Ds)e^{-\beta n} \tag{2}$$

where D is the design task instance duration attribute; n is the number of accumulated iterations; Ds is the standard duration the designer agent is capable; Di is the initial duration of the task; and ß is the duration progress rate coefficient. Durations are defined during simulation as real numbers corresponding to time units.

Equations 1 and 2 thus model the effect of iteration in the solution's quality and task's duration through exponential *learning curves* (Leibowitz et al., 2010), which is a standard way of describing that further iteration improves the performance achieved by the agent, but yields increasingly lower gains (Hamade et al., 2005). This allows AMPERE to incorporate the empirical notion that the design solution improves faster during early stages of design exploration and convergence and, as the design process progresses to stages of refinement and repetition, it is required an increasingly higher effort to continue design optimization. In addition, it models also the notion that designers learn from experience accumulation and this allows them to perform subsequent repetitions of activities faster than initial executions. AMPERE allows also Junior and Senior Designers to have different improvement rates, according to their level of *experience*.

Moreover, in order to account for the effect of day-to-day *variability* in individual performance of agents, Equations 1 and 2 have been implemented with probability density functions associated to the standard and the initial values and to the improvement rate coefficient. Simple probability density functions, such as triangular functions, are used to capture the effects of variability during simulation. In addition to task variability, the effects of *external uncertainties*, such as events of requirements change transmitted from the Customer, have also been captured in AMPERE through a loss of design solution quality and work efficiency that has been achieved by the agent until the event occurred. This deterioration captures the concept that change events transport designers to a state of lower knowledge, since goals have been modified. The deterioration in quality and work efficiency has been modelled in AMPERE proportionally to the *magnitude of change* perceived by the agent.

3.4 Modelling collaboration and adaption

In addition, facets of collaborative behaviour often observed during early complex design have been also incorporated. One fundamental dimension of collaboration is accounting for the traditional breakdown of large pieces of work into separate and smaller parts, which are delivered by different actors. AMPERE allows the definition through the agent's task library of specific responsibilities to individual Senior or Junior Designers, such as component or discipline-related design responsibilities. For instance, the model allows one agent to be responsible for the aerodynamic design of one component and to pass information to another agent part of the team which may be responsible for the mechanical design of the same component. Both work until they are satisfied with their individual solutions, reacting to changes that are communicated during that process.

Since this facet of collaboration requires partial solutions to be integrated into a more global solution, AMPERE includes the Collaborative Solution class which allows agents to track and store the results from collaborative and distributed design processes. The quality of a collaborative design solution instance is defined according to Equation 3:

$$CSQ = \sum wjQj \tag{3}$$

where *CSQ* is the collaborative solution instance quality attribute; *wj* is the contribution weight of each part or discipline aspect for the global design solution quality and *Qj* is the design solution instance quality attribute of each part or aspect, being *wj* subject to the constraint:

$$\sum wj = 1 \tag{4}$$

During simulation, *CSQ, Qj* and *wj* take real values between [0; 1] and Equations 3 and 4 capture the effects of collaboration based in work decomposition. Taking advantage of the ease of implementation of concurrency in agent-based models, AMPERE allows different Designer agents to be working on alternative solutions for the same component or disciplinary aspect. This intends to capture the dynamics of types of iteration, such as exploration and convergence (Wynn et al., 2007), where designers work concurrently on several candidates for the solution. The Collaborative Solution class has been developed with methods allowing agents to screen among alternative solutions developed for the same goal and select the one that achieved the most promising quality level.

Facets of adaptive decision-making behaviour encountered during complex design have been also captured in the agent model. The practical reasoning behaviour used by agents to plan their actions according to the perceived environment state is, in essence, a way of capturing *in-situ* adaptive decision-making. An example of adaptation is the Lead's capability of evaluating risks and deciding either to direct the resources available in the design team to work on project activities or allow them to remain available to work in other projects. Risk evaluation made by the Lead agent includes *in-situ* estimation of the probability and impact of not meeting the Customer's expectations and risk computation according to the standard approach: Risk = Probability x Impact. The probability of not meeting the Customer's expectations has been modelled in AMPERE as a function of the time available until the deadline for design proposal submission and the gap in the solution quality relative to the Customer's expected level. The impact of failing to deliver has been modelled as a linear function of the gap in quality relative to the expectation. Based on this risk of failure, the Lead agent continuously adjusts his design team's resources, which essentially models the behaviour of *adaptive planning* to the environment's changes often observed in design organizations.

4 EXPLORATORY SIMULATIONS

Based in previous empirical research of early stages of aircraft jet engine design (Fernandes et al., 2014), this section presents and explores results arising from simulation of an initial AMPERE model conceptualizing a simple scenario: a single Customer agent that regularly sends a request for a solution proposal to a single supplier. Within this organization, a single design team composed of a Lead agent and several Senior and Junior Designer agents is in charge of generating a design solution proposal that meets the Customer's expectations and respond to the Customer before the specified deadline. In this simple model, the Lead agent has an available design team composed of five designer agents: two aerodynamic Designers, one Junior and one Senior; and three mechanical Designers, where two are Junior level and one is Senior level. In addition, change events occur stochastically in the market environment that the Customer is continuously observing, which generates updates of requirements to the design team. The team is then capable of adapting resources and execute actions, including design activities, to tackle the arrival of changes and deliver a solution proposal to the Customer.

The setup of this simple AMPERE model for simulation essentially included the definition of internal parameters adjusting the behaviour of the environment and the general behaviour of the agents previously described in this paper. Results from these exploratory simulations provide an assessment of various model outputs and an investigation to the effects of expectable levels of change in project performance.

4.1 Visualization and performance evaluation

Process visualization is a key purpose of product development process modelling since it provides support for group discussions within design organizations (Browning and Ramasesh, 2007). Because of that, exploratory simulations investigated ways to picture the process output resulting from agent-based simulation to provide the generation of a design process chart as a standard post-processing feature. Figure 2 presents an annotated and detailed view over part of a complete design process chart resulting from one simulation run with the simple setup previously described. The run resulted in a design process duration of 12 weeks comprising three main interaction cycles between customer and supplier.

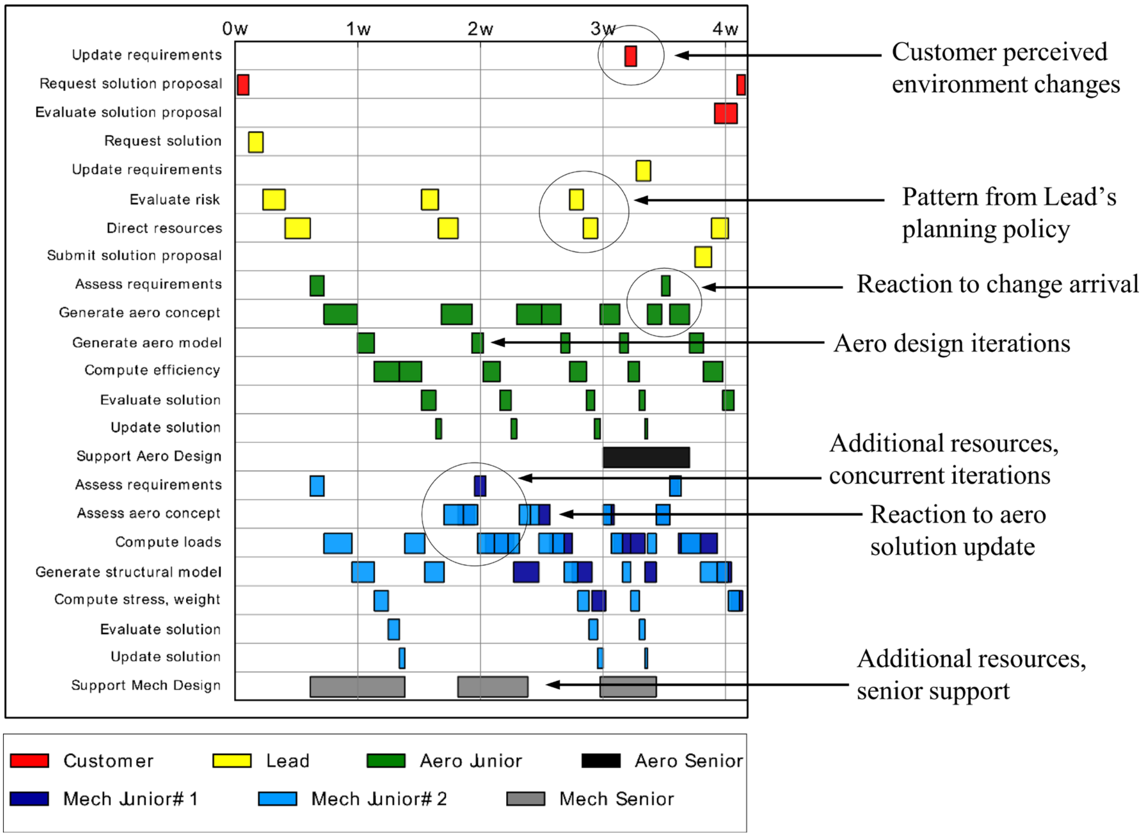

Figure 2. View of 4 weeks of the design process chart arising from a single simulation.

Several changes occurred in the market environment during the simulation, which are visible through the requirements updates sent from the Customer agent to the supplier organization. Figure 2 shows also the patterns of risk evaluation and adjustments in the number of resources allocated to the design process resulting from the Lead agent's planning activities. Directions arising from the Lead motivate Designers to start their design work. Figure 2 shows how the request for additional resources arising from the Lead's risk evaluation triggers various Designers to engage in concurrent design iterations.
Design activity repetition patterns appear in Figure 2 as a result of the need for iteration to improve the solution's quality. For instance, Figure 2 reveals aerodynamic design iterations, consisting of concept generation, calculations, solution evaluation and update to other team members. Interruptions of the natural activity cycle are also depicted, as a result of agent reaction to changes arriving from the environment. For instance, Figure 2 shows the in-situ reassessment of requirements upon the perception of a requirements change event and the repetition of previous activities as a result of design updates arriving from other design disciplines. The design process visualized in Figure 2 thus emerges from concurrent streams of design activities shaped by frequent social interactions and adaptive behaviours. The ability of capturing these dynamics typical of early design is a major strength of the agent-based modelling approach.
As a result of the resource utilization made by the design team, Figure 3 sheds light to the cost evolution during the course of the simulation time, showing how both the hourly and cumulative cost evolve as a result of the project use of the design team resources. Peak values of hourly cost consumed

by the project occurred between weeks 3 and 4 and also between weeks 6 and 7, as a consequence of time periods where all resources available in the supplier organization were performing design work in a concurrent manner (visible in Figure 2). The accumulated cost incurred by the organization during the course of 12 weeks to execute the project reached approximately 67 thousand cost units. These exploratory simulations refer to arbitrary project cost units and the reader can relate to its own monetary units.

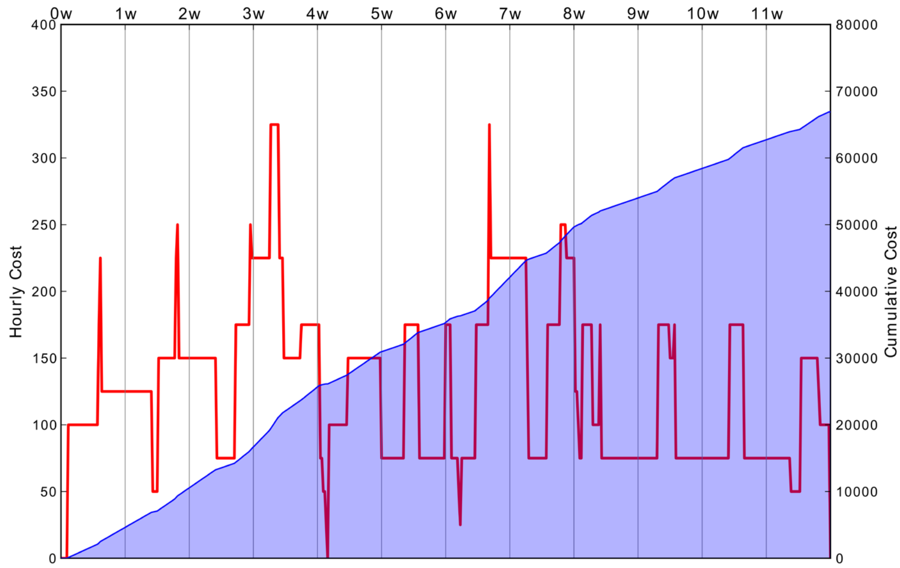

Figure 3. Cost evolution during the course of a single simulation run. Cost values are presented in generic project cost units.

4.2 Effects of change in performance

One of the strengths of agent-based simulation is the possibility of investigating complex cause-effect relationships in a cost effective manner. Because of that, exploratory simulations have also addressed the need for investigation of the relationship between requirements change and its effects in the global project performance.

The study consisted essentially in varying the external environment's time between changes probability distribution - keeping all other setup variables constant - and observe the response behaviour. The uniform distribution defining the environment's time between changes was varied in steps of one working week. Fifty simulation runs of the model were performed with each modified setup. The project's performance was measured in terms of two key metrics: the solution quality achieved in the last proposal delivered by the design team within the due date; and the cumulative cost incurred by the project. Both were characterized statistically for each set of 50 simulation runs using the median of quality and cost values.

Results are presented in Figure 4. The design solution quality response behaviour reveals that there is an interval or plateau - a change in requirements triggered each 3 to 5 weeks - where variations in the time between changes in the environment have little effect in the solution quality delivered to the Customer. In addition, Figure 4 shows that further reductions in the change frequency beyond 3 weeks produces a reduction of the last proposal's solution quality with increasing rates. On the other hand, project cost appears to behave rather linearly and inversely to the increase in the time between changes in requirements arriving from the environment. This behaviour arises from the fact that Designer agents spent less time looking for a new design solution due to updated requirements and thus progress faster as the rate of arrival of changes decreases.

Looking to the design process as a system, Figure 4 suggests that there is a *stability region* relative to the arrival rate of changes and an *instability point*, which determines the transition to a state where changes arrive faster than what the system can cope with. Such state should naturally be avoided since project performance is significantly affected. The development of more complex models of early design with a higher number of hierarchically organized design teams, each responsible for sub-system

and component design and capable of communicating changes independently at different frequencies, is however required for further investigating this hypothesis.
Considering that these results arise from a very simple model, this paper argues that agent-based models such as AMPERE are a promising approach to investigate the behaviour of complex design systems and can support industrial decision-makers planning projects during early stages. One example of such planning support is the estimation of the most likely performance of projects during early design, for a given likely level of requirements change.

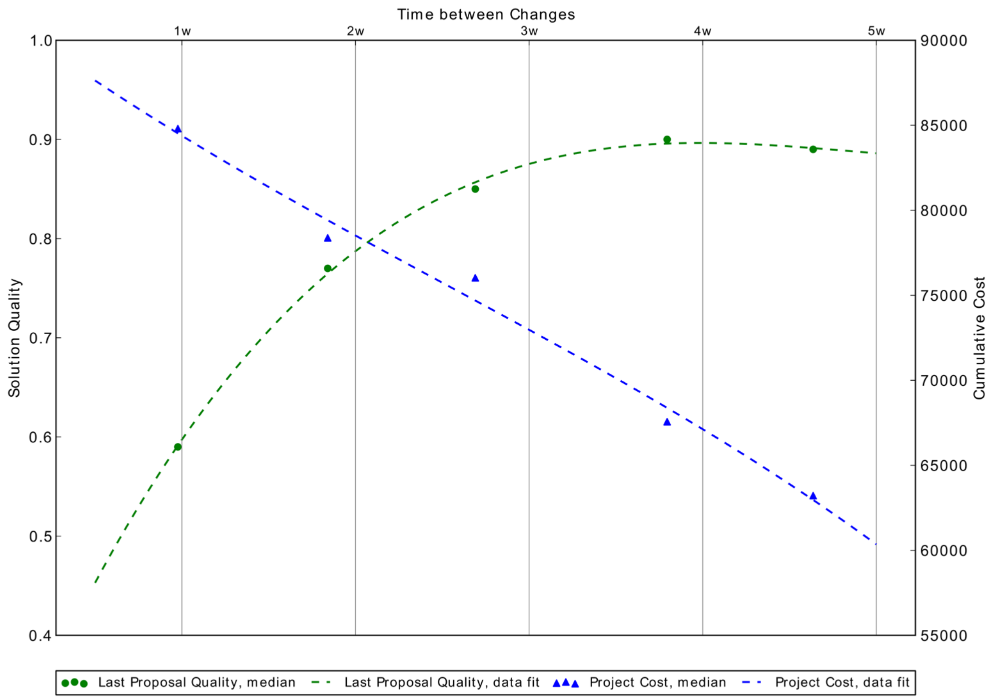

Figure 4. Effects of change in project performance. Each performance data point arises from 50 simulation runs of a particular environment setup and refers to computed median values.

5 CONCLUSIONS

Based on a general-purpose agent development platform, we have synthesised an Agent Model for Planning and rEsearch of eaRly dEsign (AMPERE) aiming to understand the behaviour of design teams and support early stage planning activities during complex product development, such as planning for likely effects of changes in requirements.
This agent model conceptualized key actors of early design stages, such as the external Customer, a design Lead and Junior and Senior Designer agents belonging to a design team. Agents were realised based on a practical reasoning algorithm constructed with internal Belief-Desire-Intent data structures and with a possible set of agent actions on the environment. During simulation, the design agent's actions are influenced by the stakeholder requests and changes arising from the environment. Results from AMPERE simulations include the level of solution quality achieved by the design team and the project cost associated to the use of the direct resources, which are key performance metrics supporting planning activities.
This paper showed that AMPERE simulations allow the estimation of the likely level of solution quality achieved and the project cost that will be induced to accomplish a certain proposal quality level for a given level of expectable change arising from the agent's environment. Focusing on externally-driven requirements change, AMPERE supports also the realization of sensitivity studies to understand and characterize the project's response to varying levels of incoming change.
In addition, considering previous product development modelling approaches such as activity-based models, this paper argues that agent-based models offer significant advantages to capture the dynamics of early design, such as uncertainty, iteration, collaboration and adaptation. AMPERE simulations showed that agent-based models facilitate the capture of concurrency in activity streams, frequent social interactions and asynchronous information exchange between design teams, collaborative

efforts and distributed and adaptive decision-making of design actors. Capturing such dynamics is critical to model the early stages of complex design processes. Further development and application of agent-models using the AMPERE framework is a promising avenue for future research.

REFERENCES

Bratman, M., Isreal, D. and Pollack, M. (1998) Plans and resource-bounded practical reasoning, Computational Intelligence, Vol. 4, No. 4, pp. 349-355

Browning, T.R. and Ramasesh, R. V. (2007) A survey of activity network-based process models for managing product development projects. Production and Operations Management, Vol. 16, No. 2, pp. 217-240.

Browning, T. (2001) Applying the design structure matrix to system decomposition and integration problems: a review and new directions. IEEE Transactions on Engineering Management, Vol. 48, No. 3, pp. 292-306.

Cho, S. and Eppinger, S. (2001) Product development process modeling using advanced simulation, ASME Design Engineering Technical Conference Proceedings, Pittsburgh, Pennsylvania, pp. 110

Danilovic, M. and Sandkull, B. (2005) The use of dependence structure matrix and domain mapping matrix in managing uncertainty in multiple project situations, International Journal of Project Management, Vol. 23, No. 3, pp. 193-203

Dong, Q. (2002) Predicting and managing system interactions at early phase of the product development process. PhD thesis, Department of Mechanical Engineering, Massachusetts Institute of Technology, Boston, MA, USA

Eppinger, S., Nukala, M., and Whitney, D. (1997) Generalised models of design interaction using signal flow graphs, Research in Engineering Design, Vol. 9, No. 2, pp. 112-123

Fernandes, J., Henriques, E., Silva, A. and Moss, M. (2014) A method for imprecision management in complex product development, Research in Engineering Design, Vol. 25, No. 4, pp. 309-324

Fernandes, J., Henriques, E., Silva, A., Moss, M. (2015) Requirements change in complex technical systems: an empirical study of root causes, Research in Engineering Design, Vol. 26, No. 1, pp. 37-55

Gregori, M.E., Cámara, J.P. and Bada, G.A. (2006) A jabber-based multi-agent system platform, Proceedings of the Fifth International Joint Conference on Autonomous Agents and Multiagent Systems, New York, USA, pp. 1282-1284

Hamade, R.F., Artail, H. A. and Jaber, M. Y. (2005) Learning theory as applied to mechanical CAD training of novices, International Journal of Human-Computer Interaction, Vol. 19, No. 3, pp. 305-322

Leibowitz, N., Baum, B., Enden, G. and Karniel, A. (2010) The exponential learning equation as a function of successful trials results in sigmoid performance. Journal of Mathematical Psychology, Vol. 54, No. 3, pp. 338-340

Murata., T. (1989) Petri nets: Properties, analysis and applications, Proceedings of the IEEE, Vol. 77, No. 4, pp. 541-580

Pritsker, A. (1966) Gert: graphical evaluation and review technique, Technical Report RM-4973-NASA, The RAND Corporation, Pittsburgh, Pennsylvania

Wiest., J.D. (1995) A management guide to PERT/CPM. Prentice-Hall, 1969

Wooldridge, M. and Jennings, N. (1995) Intelligent agents: theory and practice, Knowledge Engineering Review, Vol. 10, No. 2, pp. 115-162

Wynn, D. (2007) Model-based approaches to support process improvement in complex product development. PhD thesis, University of Cambridge, Cambridge, UK

Wynn, D., Eckert, C. and Clarkson, P.J. (2006) Applied signposting: a modelling framework to support design process improvement, In Proceedings of the ASME International Design Engineering Technical Conferences and Computers and Information in Engineering Conference, Philadelphia, Pennsylvania, USA

Wynn, D., Eckert, C. and Clarkson, P.J. (2007) Modelling iteration in engineering design, In Proceedings of the 16th International Conference on Engineering Design, Paris, France

ACKNOWLEDGMENTS

Funding from the Portuguese Foundation of Science and Technology under the doctoral grant SFRH/BD/51107/ 2010 is gratefully acknowledged.