Investigating Relationship Conflict within the Social Network of Large IS Projects using a SIR Model

1st Yuyang Tang
Department of Management Science
Lancaster University
Lancaster, United Kingdom
Email: y.tang1@aol.com

2nd Richard A. Williams, *Senior Member, IEEE*
Department of Management Science
Lancaster University
Lancaster, United Kingdom
ORCID: 0000-0001-6333-9448

***Abstract*—In large corporate organisations, significant business transformation initiatives are often implemented using large IS and IT projects that involve multiple third-party providers. These are often structured around multi-partner project teams that align to the various functional and technical aspects of the new system, incorporating a diversity of members within each project team, which frequently results in conflicts that spread across the social network of the implementation programme. While several studies on conflict in project teams have been published, to the best of our knowledge, there are no visual simulations that allow observers to examine the underlying social relationships and system dynamics. As a result, we believe exploring the process of conflict propagation through computational modelling will become a critical component of the study of conflict management. A common approach for modelling complex systems is agent-based modelling and simulation, which enables complex emergent behaviours to emerge at the system-level through simple rule-based interactions at the individual agent-level. We developed an agent-based model of conflict that incorporporated the Susceptible-Infectious-Removed model from epidemiology, in order investigate the propagation of conflict in a large IS implementation. Conflict is viewed as an infection, and the process of post-conflict management as patient recovery. We illustrate the model's validity in reality and the results demonstrate that regardless of the magnitude of the conflict, if it is not adequately handled at the start, it may rapidly spread throughout the network.**

***Index Terms*—Agent-Based Model, Relationship Conflict, Information System Projects, Project Social Network, SIR Model**

I. Introduction

Practitioners are well aware that within large organisations, there is often a disparity in business drivers and objectives between the different departments, employees and other stakeholders. Furthermore, for large organisations, the implementation of large information system (IS) projects, such as Enterprise Systems, is increasingly used to deliver significant cost efficiencies and other business transformations by standardising business processes within the organisation. For large organisations, these IS projects are usually implemented through using one or more third-party providers, which runs the risk of significant conflict arising [1]. Indeed, throughout the previous two decades, the three major forms of conflict (task, process, and relationship conflict), have had a detrimental effect on team performance [2]. The potential for conflict is maximum within large corporations that are structured around numerous departments that implement their own projects within a wider programme implementation, and where third-party software vendors and consulting firms are contracted to perform specialist roles that the customer organisation is unable to resource. This combined, multi-organisational programme, will thus consist of team members with diverse cultural, educational and professional backgrounds, which affects their behaviours and individual perspectives [3]. Importantly, and in a way that is similar to an epidemic, if relationship conflict develops within one of the multi-organisational project teams, it has the propensity to spread to other project teams that make up the programme-level social network, resulting in greater harm to the implementation as a whole [3]. Recent research indicates that cybernetics can address and regulate conflicts within an organisation's operations and can help develop interventions to reduce the risk of conflict development [4], with computational modelling presenting a viable tool to help us design these interventions.

Despite this, there is still a lack of visualisation and computer simulation of the processes and dynamics of conflict in project networks [3]. Therefore, in order to have a better understanding of the dynamics of conflict, we have recently proposed using Agent-Based Modelling and Simulation (ABMS) to model conflict propagation and to perform simulation-based experiments to better understand how we can implement effective mitigations to reduce the likelihood of conflict, or contingency plans to dampen conflict once it has developed [3]. ABMS is a computational modelling technique that is capable of modelling complex dynamical systems. In comparison to standard mathematical models, it places a greater emphasis on agent interaction and is more akin to natural thinking, which can aid in the understanding of hitherto unknown issues [5]. Agent-Based Models (ABMs) can be used to identify causal relationships in epidemic simulations by varying various circumstances and have even been used to simulate the effect of policy interventions on health [6], which we believe will also hold true for research around the causes of relationship conflict in the social network of multi-partner IS implementations.

R.A. Williams was in part funded by an Economic and Social Research Council Fellowship with Grant No. ES/L002612/1.

Within epidemiology, the Susceptible-Infectious-Removed (SIR) model simulates the transmission of an infectious illness [7], with the potential that an infected person will recover and not get reinfected. ABMS has been used to simulate both the spread of information within social networks [8] and the way in which the SIR model can be used to detect information within networks [9]. We assume in this study that conflict spread in IS implementations is similar to the spread of infection, and that the impact of conflict management strategies is comparable to *recovery* (and not removal) in a modified SIR model.

Our goal is to build an ABM based on a recent case study that used a multi-partner approach to implement a new Enterprise System at a large UK-based organisation [3]. Our ABM simulates the development of conflict within a specific project team, which if not managed properly can propagate throughout the wider programme. This can be used to investigate the factors that contribute to the proliferation of conflict, and importantly how conflict can be managed once it has developed. We believe the ABM can be used to develop a better understanding of how the interrelationships between various departments and individuals can give rise to relationship conflict, and provide inspiration for developing effective project management interventions to reduce the likelihood of conflict development or dampen effects once it has developed.

II. Methodology

Our approach to ABM development followed the process presented by [10], which provided a structured approach to the design and development of computational models for use as *Scientific Instruments* to perform simulation-based experimentation.

A. Data Collection

We used the previously collected data presented in [3], which investigated the development and propagation of conflict within a large UK-based multi-partner Enterprise System implementation. The qualitative data focuses on the development and propagation of conflict, whilst the quantitative data focuses on the programme-level social network, including the list of team members and their project-based demographics, and importantly the formal work-based relationships between team members.

Briefly, there were 159 team members, structured into 7 distinct projects, along with a Programme Management Office (PMO). The individual project teams were aligned to the functional modules within the Enterprise System software or the technical activities that were required to host the software on the client computing infrastructure, these were: Financials, Human Resources (HR), Payroll, Training, Hosting, Technical and Offshore. These project teams were composed of resources from the Customer, the Software Vendor and one of three Consulting Firms who provided subject matter expertise to the Customer, in order to ensure the configuration of the Enterprise System and development of custom extensions was appropriate for their business needs.

B. Developing the Agent-Based Model

We developed a conceptual model by following the approach advocated by [11]. We used the social network map from [3] in order to diagramatically define the location of the 159 team members within their specific project team and the formal work-based relationships between them. We used the *Rich Picture* from [4] to provide a single diagram that presents the work-based interactions between team members and the complex emergent behaviours that result from these interactions, in particular the development and propagation of conflict. Furthermore, we used a number of diagrammatic notations from the Unified Modelling Language [12] to define the detailed interactions that give rise to relationship conflict between team members and its propagation throughout the project team, then wider programme-level social network.

The conceptual model was then translated into an ABM using the NetLogo modelling and simulation tool [13]. Simulation-based experiments were performed and the emergence of relationship conflict and its propagation throughout the social network was analysed using inspiration from epidemiology. Specifically, we used the SIR model [7], which is a basic transmission model for a directly transmitted infectious disease, to analyse the propagation of relationship conflict once it had developed between two or more team members. Importantly, within our model we have used different terminology to represent the three states in the SIR model: we have used *Calm* to represent the susceptible state; *Conflict Infected* to represent the infectious state; and *Recovered* to represent the removed state because within the case study, team members were not usually removed, but either remained in conflict or had recovered through project management interventions.

III. Agent-Based Modeling and Simulation

A. Agent-Based Model Development

Fig. 1 shows the programme-level social network, which was developed in NetLogo and has the individual team members coded by project and employing organisation. Our previous work identified that task and process conflict, if not managed appropriately, has the tendency to evolve into relationship conflict between team members [3]. Furthermore, we identified that the Customer HR Project Manager (CustHRPM) was the predominant resource that initiated relationship conflict [4]. As such, we have simplified the development of conflict in our ABM so that conflict is always initiated by CustHRPM. Relationship conflict can then develop between CustHRPM and either their team members or those of a third-party within the HR Project Team. Once relationship conflict has formed between CustHRPM and other team members, the propagation of conflict is stochastic and follows the SIR model of infection, whereby team members in conflict have their state changed from *Calm* to *Conflict Infected* and potentially to *Recovered* if project management interventions are able to resolve the conflict. To enable stochasticity within simulation-based experimentation, we developed the ABM to ensure pseudo-random number generator seed values can be explicitly

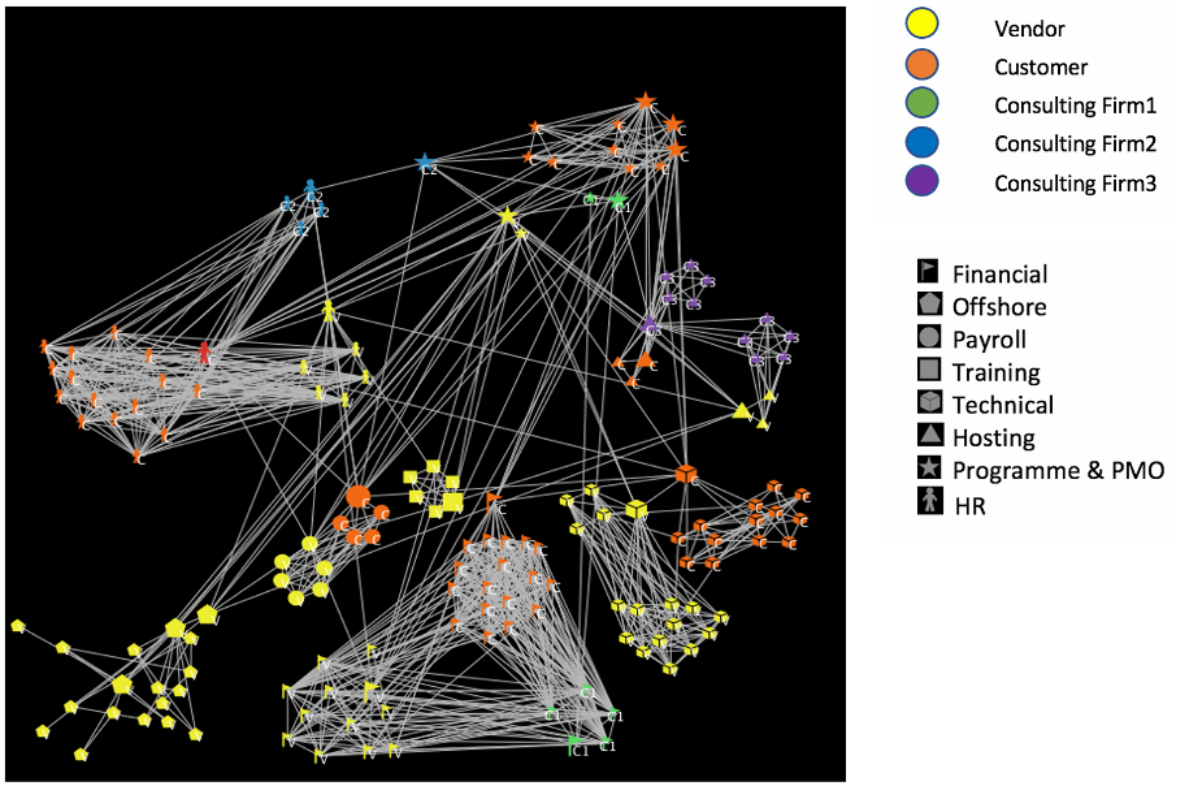

Fig. 1. Programme-level social network (developed in NetLogo), with team members coded by project and employing organisation.

set (to enable multiple replicate simulation runs), and have also used parameters for propensity to *remain calm* and probability to *recover* from conflict at the individual agent level.

B. Simulation-Based Experimentation

CustHRPM is the origin of relationship conflict within our ABM in accordance with [4]. Individual team members within the HR Project Team are then able to become *conflict infected* through stochastic interactions betweem them and CustHRPM, which are modulated through their propensity to *remain calm* parameter value. These initial *conflict infected* team members are usually the PMs from either the Software Vendor or Consulting Firm 2, but were occasionally other team members within the HR Project Team, notably Consultants from the Software Vendor or employees that CustHRPM line-managed. Once team members are infected, the relationship conflict is able to stochastically propagate through the HR Project Team and once a threshold of conflict infected team members has been reached, the team members that act as *bridgers* through formal work-based relationships to other Project Teams, can propagate the *infection* to other Project Teams within the IS implementation. As such, relationship conflict will ultimately propagate throughout the entire programme-level social network unless a project management intervention is implemented to dampen the spread of conflict and potentially *recover*, where the team member in question is now *immune* to further conflict or return back to *calm*, where they are once again susceptible to conflict.

We used the strategy of *integration* for our project management interventions within our experimentation. The strategy is the most desired approach because it completely resolves the relationship conflict by helping team members integrate into the aims of the project and also with their fellow team members, thus ensuring that the individual team members do not have incompatible ideas, personal objectives, or behaviours. Fig. 2 shows the rate of conflict growth among the various projects following conflict development by CustHRPM. As expected, it can be seen that the HR Project has the fastest rate of conflict growth and that once the entire HR Project Team has become *conflict infected*, team members wihtin other Project Teams become *conflict infected*, in particular the Payroll, Financials and PMO teams, who have close links to the HR Project Team due to the integrated nature of an Enterprise System and the project management and administrative oversight applied by the PMO. Conversely, the Technical and Offshore Project Teams have the slowest conflict growth rate and become the last within the programme-wide social network to be *conflict infected*. The propagation of conflict *infection* in the simulation is consistent with the qualitative data and conceptual model of relationship conflict development and propagation from [3] and confirms that if project management interventions are not implemented, relationship conflict will ultimately affect the entire programme-wide network.

Simulation-based experimentation was performed to investigate the types of project management intervention that

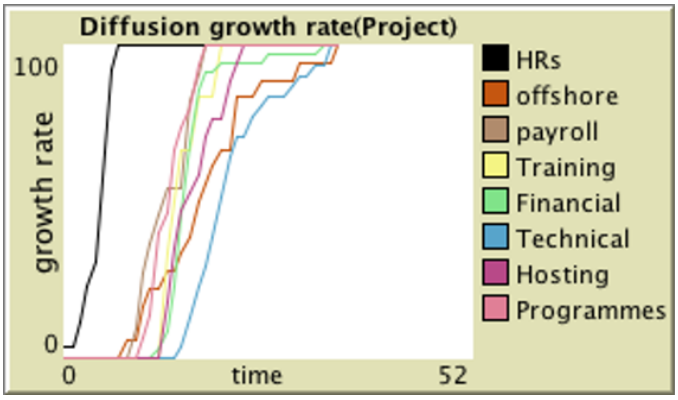

Fig. 2. Propagation rate by Project.

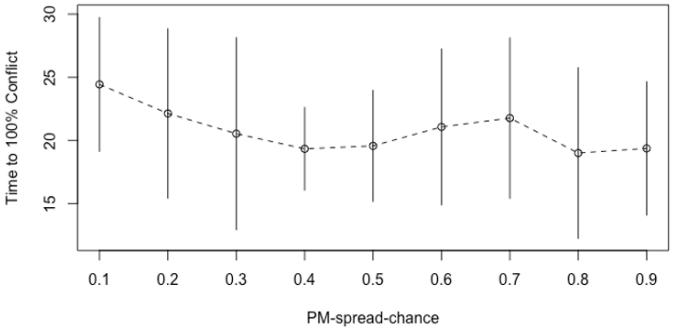

Fig. 3. Impact of PM probability of conflict propagation on the time required for complete programme-wide conflict.

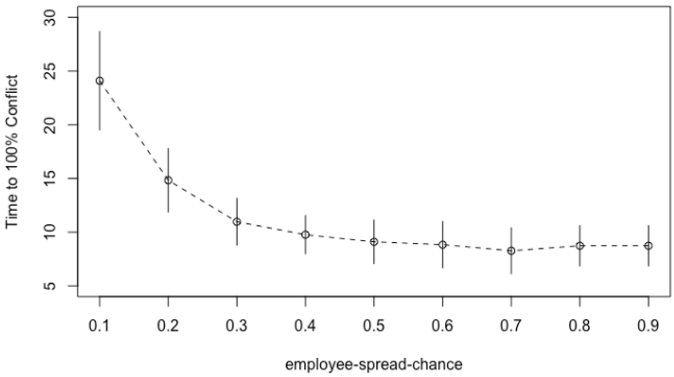

Fig. 4. Impact of team member probability of conflict propagation on the time required for complete programme-wide conflict.

can be implemented in order to control conflict once it has developed. Initial experiments focused on the likelihood of conflict propagating from one Project Team to another, through changing the threshold values, which relates to the percentage of team members being *conflict infected* before the conflict can propagate into another Project Team through the *bridger* team members. Simulation results provide no statistically significant difference, which suggests that once relationship conflict has developed within a Project Team, it is inevitable that it will propagate throughout the programme-wide social network unless specific project management interventions are applied to *conflict infected* team members in order to resolve the conflict and turn them into either a *recovered* or *calm* state. Further experimentation focused on the differing probability of conflict propagation between PMs and normal team members, through varying the *spread-chance* parameter values within the simulation. Fig. 3 shows the effect of varying the probability of PMs propagating conflict on the time required for conflict to *infect* the programme-wide social network, showing a slight decrease in the time required. Similarly, Fig. 4 shows the effect of varying the probability of team members propagating conflict on the time required for conflict to *infect* the programme-wide social network, which has a quicker rate of *infection* of the whole social network (tested through ANOVA and TukeyHSD tests to have a more significant effect than PMs), and also markedly reduced variability across replicate simulation runs. Importantly, the time required for complete conflict propagation can be seen to stabilise once the team member probability of propagation reaches a certain threshold (*spread-chance* parameter value of 0.3 or greater). We conjecture this to mean that once conflict has occurred, it must be contained to a small subset of team members or measures put in place to minimize the propagation to other team members who were not initially involved in the relationship conflict development.

IV. Discussion and Conclusions

Our simulation-based experimentation has found that relationship conflict, once developed, is able to propagate throughout the entire programme-wide social network of a large multi-partner IS implementation if no project management interventions are performed. Specifically, our experimentation around propagation probability for PMs and team members has suggested there to be a greater programme-wide effect from conflict propagating between team members than from PMs. Furthermore, our results indicate a *tipping point* where relationship conflict is guaranteed to propagate throughout the entire network if the probability of spread between team members is not controlled, with Fig. 4 indicating that the probability of spread needs to be kept below 30%.

Although PMs have access to more information as well as acting as *bridgers* in the social network, communication and interactions between team members has a more significant impact on the speed of transmission of conflict, which we conjecture might happen due to four main reasons. Firstly, relationship conflict is due to interpersonal issues and as PMs need to set the project standard for professional behaviours, they have less ability to express their irritations or unhappiness with subordinates, therefore although they might develop relationship conflict due to behaviours of another team member, they proactively try to mitigate the risk of them expressing this within the Project Team (which would cause conflict). Similarly, the second reason might be that PMs, needing to act professionally, are not able to allow their *conflict infected* state to get in the way of them completing tasks/activities associated with their role, unlike other team members, thus

reducing the likelihood of dissatisfaction arising due to the PMs performance within the Project Team. Thirdly, the social networks of individual Project Teams are split into *cliques* that correspond to either their employing organisation or the specific functional/technical activities that they perform within the project, thus when one team member has a grievance, it is likely to propagate to others within the tightly-linked *clique*, which explains why the transmission of relationship conflict between team members is guaranteed once a threshold is reached for the probability of propagation. Finally, since the topolgical characteristic of a large multi-partner IS implementation tends towads a *small world* network, all team members within the programme-wide social network are only a few links away from each other. As a result, when conflict spreads, team members who don't have formal work-based relationships can be *infected* through one or two intermediary members.

We believe that the PMs role is paramount in situations where conflict has developed and spread within their Project Team. This is not only due to their professional role of acting as PM, but also due to their network characteristic of being a *bridger*. As such, PMs need to develop interventions that restrict the likelihood of propagation from their Project Team to another, which we conjecture could potentially be through implementing a kind of *quarantine* intervention by temporarily severing the link to other Project Teams, or through arbitration between *conflict infected* team members, which could act in an analogous way to *antibiotics* that fight off bacterial infections in humans (and animals).

To conclude, we believe that our ABM has confirmed that simulation-based experimentation can act as a powerful tool to develop hypotheses on how PMs can intervene to control relationship conflict that can develop within large multi-partner IS programmes. Our simulations have also confirmed that relationship conflict can propagate throughout the programme-wide social network in a similar way to infections and disease propagating through social networks in the field of epidemiology. Finally, we believe that inspiration can be taken from the way the human immune system fights infection, and that analogous approaches can be developed by PMs in order to intervene when relationship conflict has *infected* their Project Team.

ACKNOWLEDGMENTS

This conference paper has been developed from the MSc dissertation of Y. Tang [14], which was submitted for the degree of MSc Business Analytics at Lancaster University following his summer research project under the supervision of R.A. Williams.

REFERENCES

[1] C.C. Chen, C.C.H. Law and S.C. Yang, "Managing ERP implementation failure: a project management perspective," IEEE T. Eng. Manage., vol. 56, pp. 157-170, 2009

[2] C.K.W. De Dreu and L.R. Weingart, "Task versus relationship conflict, team performance, and team member satisfaction: a meta-analysis," J. Appl. Psychol., vol. 88, pp. 741-749, 2003

[3] R.A. Williams, "Conflict propagation within large technology and software engineering programmes: a multi-partner enterprise system implementation as case study," IEEE Access, vol. 7, pp. 167696-167713, 2019

[4] R.A. Williams, "Cybernetics of conflict within multi-partner technology and software engineering programmes," IEEE Access, vol. 8, pp. 94994-95018, 2020

[5] U. Wilensky and W. Rand, An Introduction to Agent-Based Modeling: Modeling Natural, Social, and Engineered Complex Systems with NetLogo, Boston: MIT Press, 2015

[6] A.M. El-Sayed, P. Scarborough, L. Seemann and S. Galea, "Social network analysis and agent-based modeling in social epidemiology," Epidemiol. Perspect. Innov., vol. 9, pp. 1-9, 2012

[7] R.W. Anderson, "Discussion: the Kermack-McKendrick epidemic threshold theorem," Bull. Math. Biol., vol. 53, pp. 1-32, 1991

[8] S. Kumar, M. Saini, M. Goel and B.S. Panda, "Modeling information diffusion in online social networks using a modified forest-fire model," J. Intell. Inf. Syst., vol. 56, pp. 355-377, 2021

[9] K. Zhu and L. Ying, "Information source detection in the SIR model: A sample-path-based approach," IEEE ACM Trans. Netw., vol. 24, pp. 408-421, 2014

[10] S. Stepney and F.A.C. Polack, Engineering Simulations as Scientific Instruments: A Pattern Language, Cham: Springer, 2018

[11] S. Robinson, "Conceptual modelling for simulation part ii: a framework for conceptual modelling," J. Op. Res. Soc., vol. 59, pp. 291-304

[12] The Object Management Group, OMG Unified Modeling Language version 2.5.1, URL:https://www.org/spec/UML/2.5.1/PDF, 2017

[13] U. Wilensky, NetLogo, The Center for Connected Learning and Computer-Based Modeling, Northwestern University, Evanston, IL, http://ccl.northwestern.edu/netlogo/, 1999

[14] Y. Tang, Conflict simulation in large organisational social networks using an agent-based model. MSc Dissertation. Lancaster University, 2021.