Using Experimental Games to Understand Communication and Trust in Agile Software Teams

Eisha Hasnain, Tracy Hall, Martin Shepperd
Brunel University
Uxbridge,
United Kingdom
Eisha.Hasnain@brunel.ac.uk, Tracy.Hall@brunel.ac.uk, Martin.Shepperd@brunel.ac.uk

Abstract— **Trust plays an important role in enabling software development teams to function effectively. Trust between individual team members has been shown to improve the independence of software teams and reduce the amount of project management effort required by those teams. Our main aims are to investigate (i) the impact communication has on trust between team members in Agile software development and (ii) the usefulness of iterated games as an experimental methodology. We use Game Theory in a simulated Agile development environment. We run 28 iterated games with 56 practitioner and student participants. Stand-up meetings are used as the communication intervention compared to games without such meetings to assess the levels of trust. Our findings are that increased communication has a very large positive effect upon the level of trust between team members in an Agile setting. Our results suggest that communication improves trust in development teams. Opportunities for communication should be built into development processes. Experimental games are a complementary (to qualitative approaches) and effective method for investigating human issues in software engineering.**

Index Terms— **Software engineering, trust, communication, Agile Methods, stand-up meetings, Game Theory, iterated experimental games.**

I. Introduction

The aims of this study are twofold. First, we investigate experimentally the impact that communication has on levels of trust in a simulated Agile software development setting. Second, we demonstrate how a different research methodology based upon iterated experimental games can be used to explore relatively intangible phenomena such as trust within software engineering. We believe this is a complementary method to the more traditional qualitative empirical analysis techniques that are often employed.

Trust is a universal phenomenon that underpins interaction between people. Riegelsberger et al. [1] define trust as the "willingness to be vulnerable based on positive expectations about the actions of others". It enhances the interactions between people and leads to increased effectiveness since effort does not have to be deployed on controlling risk. We would expect trust to be important in Agile projects for a number of reasons. The very first item from the Agile Manifesto [2] is the valuing of "individuals and interactions over processes and tools". Elsewhere, as a typical example of the Agile Methodology literature, Paetsch et al. [3] assert that "Agile approaches create trust by tightly integrating the customer into the development process". Clearly the notion of improving levels of trust is seen as a potential benefit of Agile Methods. An ethnographic study by Robinson and Sharp [4] of Agile software development projects in London and South Africa also suggests that Agile development teams have faith in their own abilities, show respect and responsibility, establish *trust* [our emphasis]...".

Nevertheless, other than the work of Robinson and Sharp [4] there is very little empirical work investigating trust in the context of Agile development. Consequently the effect of trust on development teams is not fully understood, nor do we know all the factors that influence trust and their interactions. In many ways this is understandable. Trust cannot be directly measured and it is a difficult phenomenon for humans to introspect upon. The analysis and interpretation of trust is also value-laden hence trust is a challenging topic to investigate empirically and requires some sensitivity. One might characterise current approaches as being qualitative with a view to building rich pictures and observing phenomena *in vivo*.

We try to overcome some of these difficulties and adopt a contrasting but complementary approach by experimentally investigating trust based upon the use of Game Theory [5]. Game Theory is a mathematical approach to understanding an individual's behaviour when that behaviour is based on interactions with others. As such it is potentially an excellent means by which to investigate trust particularly by means of iterated (infinite) experimental games.

The remainder of the paper is organised as follows. In Section II we describe our experimental method. We present our results in Section III followed by discussion and conclusions in Section IV and ending with a review of the threats to validity in Section V.

II. METHOD

A. Research Questions

Our principal interest is in the impact of different forms of communication upon trust levels in Agile software development. This leads to the following four research questions:

1. What impact does communication have upon trust between team members in Agile methods?
2. Does trust change over time, in other words over the duration of the game?
3. Do trust levels differ between male and female participants?
4. Are there difference in behaviour between student and professional participants?

In all cases the confidence level is set at $\propto = 0{:}05$ for the inferential tests based on two sample t-tests for Q1, 3 and 4 and a correlation test for Q2. However our principal concern is the effect size since from a practical perspective this is what matters.

B. Participants

There were a total of 56 participants resulting in a total of 28 games, each of which had 10 rounds. Of the 56 participants, 32 were Brunel University undergraduate computing students and 24 were software professionals. The undergraduates were final year students enrolled on a Project Management module and were volunteers. The professional participants were personal contacts (of the first author EH) all of whom were working in the software industry.

C. The Game

The experimental intervention was the stand-up meeting. Half of the games (14) required stand-up meetings to be role-played and the other half (14) did not. Short (approximately two minute) stand-up meetings were role played at the end of each round in games holding stand-ups. In these stand-ups (which followed the standard structure of stand-ups [6]) players discussed the problems they encountered `yesterday' (i.e. what happened in the previous round), the plans they had for their work today (i.e. the next round) and problems which may affect this work. Of course no real development was taking place so players had to imagine work based on the scenario provided and their work or shirk decisions. In the case of the intervention without stand up meetings participants were still in the same room, however, they were not allowed to communicate during the duration of the game. Our research questions are tested using an iterated non-zero sum 2-player game based on a variant of the Work Shirk game and an Agile software development scenario. Participants were unaware of how many rounds would be played, hence the game was infinite, although in fact each game comprised ten rounds. Each round both players were required to independently make `work' or `shirk' choices. They each privately recorded their decisions that were then submitted at the end of the round. Their decisions were based on the payoff matrix with which they were provided a copy (Table II).

TABLE I. PRISONER'S DILEMMA PAYOFF MATRIX

	Cooperate	**Defect**
Cooperate	R,R	S,T
Defect	T,S	P,P

Note: T is the Temptation to defect, R is Reward for joint cooperation, and P is the Punishment for both defecting and S is the Sucker's payoff (i.e. being betrayed).

TABLE II. PLAYER'S PAYOFF

		Developer 1	
		Work	***Shirk***
Developer 2	***Work***	2,2	1,3
	Shirk	3,1	0,0

The canonical structure is similar to the PD game (Table I) and the actual payoffs are given in Table II. This shows that a player gets two points if both choose to work (i.e. cooperate through trust). A player will receive one point if he or she works (trusts) while the other will receive three points if he or she shirks (betrayal). If both players shirk, then both players receive zero points (neither trusts). This payoff matrix benefits a mutual trusting behaviour where both players decide to work, compared to both players shirking hence it is a non-zero sum game. The aim is for each player to maximise their points. In both sets of games we informed each player of the other player's choice after every round. Our matrix slightly deviates from a standard Prisoner's Dilemma or Work Shirk structure in that it does not satisfy the inequalities of $T > R > P > S$ (see Table I), as in our matrix T is 3, R is 2, P is 0 and S is 1, since 0 is not greater than 1.

III. RESULTS

Although the software development scenario is imaginary the participants quickly adopted their assigned roles. So for example, during meetings there were comments on issues such as future intention. Interestingly future intention dominated discussions rather than accounting for the past. Example comments include: "it's a long day and I still haven't finished my work", outcome = work. "I just wanted to finish our work but an emergency happened so I have to go" [justifying the intention to shirk], outcome=shirk. "There is no time for a break we're here to work" reply "I think we don't need to think about breaks", outcome = both work "I need to do some more work", outcome = shirk "let's see if I can make it or not" [noncommittal], outcome = shirk "I can take some time off" reply "ok, agree [meaning we both will]" outcome = both shirk.

In order to test Research Question 1 (that communication increases trust between team members in agile methods) we randomly assigned pairs of participants to one of two different types of games reflecting the communication treatment. One set of games was based upon stand-up meetings and the second set was without stand-up meetings. In all games we interpreted participants choosing the work option as a positive intention to create trust between themselves and their work partner because this choice makes the player vulnerable to the other player choosing to shirk. We

determined the overall trust level (response variable) by counting the number of work (trust) and shirk (betrayal) choices made by each individual over the ten games, hence the trust count ranges from zero to ten[1]. Note that all participants varied their strategies so that no player only chose work or only chose shirk. In practice it ranged from a minimum of 4 to a maximum of 9 across both treatments. Table III shows the summary of the individual trust counts grouped by whether they have stand-up meetings or not.

We also see that the standard deviation (SD) is greater without the stand-up meeting indicating more variability. In other words not only does the lack of communication lead to reduced trust but also more variability in behaviour. To further confirm our results we used a 2-sample, 1-tailed t-test (alpha = 0:05) to compare the sample means. The null hypothesis is rejected and instead the alternate hypothesis of there being a significant increase is accepted ($p \leq 0{:}0001$). Analysis of the collective trust of a game (i.e. when both players simultaneously co-operate) produces a similar result also with $p \leq 0{:}0001$. Glass's Δ is 2.095 ((7.79-5.64)/1.026) which indicates a very large effect size [7].

Next we examined the second research question to see whether participants changed their behaviour over the course of the game. It is hard to see an overall pattern (see Figure 1 which shows the total count of trust (work) decisions over the course of the 10 rounds of the experiment) and a correlation test of Total Trust (summed for all games) over time (round number) indicates a small non-significant correlation coefficient (r = 0.197). Therefore we conclude that trust levels do not significantly change over the duration of the game.

The third question considers whether any gender differences may be found. The mean value for male trust level is 6.73, whereas mean value of female trust level is 6.70.

TABLE III. SUMMARY OF INDIVIDUAL TRUST GROUPED BY THE STAND-UP INTERVENTION

Stand-Up	Count	Mean	Median	Min	Max	SD
N	28	5.64	6	4	7	1.026
Y	28	7.79	8	6	9	0.787

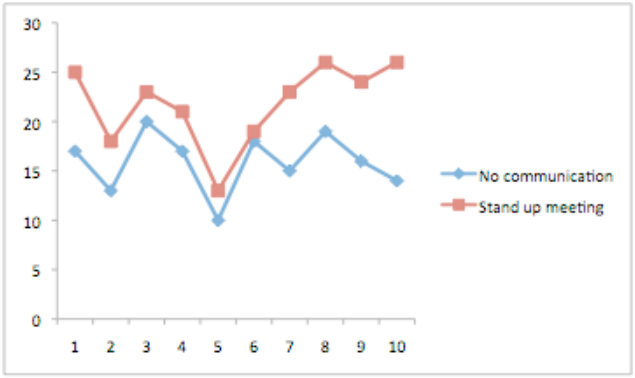

Fig. 1. *Line plot of count of trust decisions (out of 28) per round by treatment*

[1] The betrayal count is simply the number of rounds n minus the trust count t so in our case 10-t

Both values are almost identical. Whilst hardly necessary, a 2-tailed, 2-sample t-test yields (t-Statistic = -0.0785, p = 0.938) and so the null hypothesis stands. This might be slightly surprising as a number of studies have reported evidence from games of women behaving in a more alturistic fashion than men, such as [8].

The use of student participants has often been criticised [9] although at least one other study [10] reported only minor difference between student and practitioner performance. In our experiment we used a mixture of students and software professionals which provided us with the opportunity to investigate if there were differences in behaviour (Research Question 4). No significant difference was found when comparing the mean trust levels between professionals and students which are 6.45 and 6.90 respectively (p=0.24) using a 2-sample t-test. This tends to support the view point of Höst et al [10] that there are fewer distinctions between professional and student participants than might be pre-supposed, though of course this is in a very specific context.

IV. Conclusion

In this study we have used Game Theory to explore the impact of communication on trust. From this study we have two sets of findings. First, our study allowed us to answer Research Question 1 that communication increases trust between team members in Agile methods. Our results strongly suggest that communication plays a very important role in trust. Our experiment revealed a very large difference (>2 SDs) in trust between games where communication between players was required in the form of a stand-up compared to games where no stand-up was required. Therefore, we see clear evidence of the following:

> "It comes as absolutely no surprise that communication improves immediate outcomes in a PD (conventional or impartial). Communication, whether it is face-to-face or electronic, can largely overcome the traditional dilemmas [of PD type games]" [11].

However, we found no particular trend over the duration of the game so no evidence, for instance, supporting the conjecture that trust might grow (or diminish) over time. This is slightly surprising and might be a topic to pursue further. Nor did we find any statistically significant difference between the behaviour of students and professionals or males and females.

Second, we have shown that Game Theory is an effective method by which to investigate trust as it allowed us to simulate behaviour in relation to trust and to directly observe that behaviour and analyse it objectively. Our experiments have been very simple but clearly there is considerable potential to explore more complex scenarios such as multi- player games (n > 2), multi-role games where the payoff matrix is not symmetric, dynamic payoff matrices and games with more sophisticated interventions (e.g. email, Skype meetings, etc.). This is in contrast to most previous studies where human issues like trust, are

explored only indirectly (self-reported via methods such as questionnaires or interviews). However, we do stress that we see experimental games as a complementary technique and one that might pinpoint problems or raise questions that might better be resolved by more traditional qualitative methods. A good example of this is why did time seem to have so little impact upon trust behaviour. Interviewing the participants could yield richer insights.

Previous studies have shown that trust plays an important role in developing effective software teams. Our results suggest that introducing specific communication points in the form of stand-ups is a very powerful way to improve trust. This is an important finding for project managers and suggests that much attention should be given to the frequency and nature of communication points in any software development process.

V. Threats to Validity

A. Representativeness

The student participants volunteered so it is possible that they differ in some sense from non-volunteers perhaps by being naturally more cooperative. As a result our student sample may contain some bias. The professionals were also not selected randomly. This means that we cannot claim that our sample of participants is fully representative of the population of Agile software developers.

B. Communication

The players in games without stand-up meetings were located in the same room as each other. Although players did not verbally communicate there is a possibility that non-verbal communication took place, e.g. facial expressions were interpreted by players. If this had an impact it would be to reduce the extreme nature of the control treatment and so reduce any observed effect size.

C. *Payoff matrix*

As previously discussed, to have the form of a PD, a game needs to satisfy certain conditions. One of the conditions is that it should satisfy the inequalities $T > R > P > S$ (see Table I). Our matrix does not fully satisfy this condition, however this should not substantially impact the results we report here as we do not report the pay-off scores achieved by players. Instead we report only the number of times players choose the option work or shirk, i.e. trust or not trust.

D. Role Playing

All participants role played as software developers in an Agile team. Some participants also role played stand-up meetings. Although we asked participants to base their role playing on their previous work experiences, it is difficult to know how participants would actually have behaved in a real Agile team. Therefore it is difficult to know how well our simulated results reflect real-world behaviour.

Acknowledgments

The authors would like to thank all the participants for giving their time to this experiment. We would also like to thank Steve MacDonell and Carolyn Seaman for their perceptive and constructive criticism of an earlier version of this paper.

References

[1] J. Riegelsberger, M. Sasse, J. McCarthy, The researcher's dilemma: evaluating trust in computer-mediated communication, International Journal of Human-Computer Studies 58 (2003) 759-781.

[2] K. Beck, Manifesto for agile software development, 2001. http://agilemanifesto.org/ (Retrieved April 5, 2011).

[3] F. Paetsch, A. Eberlein, F. Maurer, Requirements engineering and agile software development, in: 12th IEEE International Workshop on Enabling Technologies: Infrastructure for Collaborative Enterprises, Linz, Austria, pp. 308-313.

[4] H. Robinson, H. Sharp, The characteristics of XP teams, in: Extreme Programming and Agile Processes in Software Springer Verlag, Berlin, 2004, pp. 139-147.

[5] E. Rasmusen, Games and information: An introduction to game theory, Blackwell Publishing, Oxford, 2007.

[6] J. Yip, Its Not Just Standing Up: Patterns for Daily Stand-up Meetings, Technical Report, Thought Works, Inc, 2003.

[7] J. Cohen, A power primer, Psychological Bulletin 112 (1992) 155-159.

[8] C. Eckel, P. Grossman, Are women less selfish than men? The Economic Journal 108 (1998) 726–735,

[9] D.I.K. Sjøberg, B. Anda, E. Arisholm, T. Dybå, M. Jørgensen, A. Karahasanovic, E. Koren, M. Vokåc, Conducting realistic experiments in software engineering, in: 1st IEEE International Symposium on Empirical Software Engineering (ISESE02), IEEE Computer Society, 2002

[10] M. Höst, B. Regnell, C. Wohlin, Using students as subjects - a comparative study of students and professionals in lead-time impact assessment, Empirical Software Engineering 5 (2000) 201-214.

[11] N. Frohlich, J. Oppenheimer, Some consequences of e-mail vs. face-to-face communication in an experiment, Journal of Economic Behavior and Organization 35 (1998) 389-403