A PRINCIPAL-AGENT APPROACH FOR THE CROWDING-OUT EFFECT IN INFORMATION TECHNOLOGY OUTSOURCING CONTRACTS UNDER ASYMMETRIC INFORMATION

Thyago Celso Cavalcante Nepomuceno
Universidade Federal de Pernambuco
Av. Prof Moraes Rego, 1235 – Cidade Universitária, Recife – PE
thyago.nepomuceno@yahoo.com
,
Ana Paula Cabral S. Costa
Universidade Federal de Pernambuco
Av. Prof Moraes Rego, 1235 – Cidade Universitária, Recife – PE
apcabral@ufpe.br

ABSTRACT

It has been claimed in the literature the role of punishment as promoting a behavior that can align both objectives the client and the vendor into a set of actions desirable for both. Penalty clauses are commonly used in outsourcing contracts to guarantee a certain level of quality for the good or service or to assure that the delivery date will be respected. However, some grades of penalties instead of avoid the behavior of the vendor from the undesirable by the client works as an incentive to the commitment of more irregularities. The current study derives a general cost structure from the software engineering literature for a typical software vendor and analyzes through a principal-agent model when the imposition of a fine by the client (the principal) whenever the vendor (the agent) delays the delivery of a software can stimulate the vendor to delay more, a behavior named crowding-out effect.

KEYWORDS: Incentive Theory; Information Technology Outsourcing; Software Engineering; Crowding-out Effect

Main area: TEL&SI – Operational Research in Telecommunications and Information Systems, OA – Other application in Operational Research, SE – Operational Research in Services.

1. Introduction

Information Technology Outsourcing (ITO) constitute a typical principal-agent problem in game theory analysis, where a client, firm or a hirer (the principal) wants to maximize their profit by delegate one or a set of activities regarded to the information technology of the business for another person or organization, a vendor or provider (the agent) who will produce what is demanded and charge a certain payment previously signed in a contract in return for the task done (Jensen and Meckling, 1976; Eisenhardt, 1989; Whang, 1992; Keil, 2005; Dey et al. 2010; Singer and Donoso, 2011). The transference of responsibility of create or supply the technological structure that will be critical to the business enables the principal to take more efforts into the core and more profitable activity for the company (Willcocks and Feeny, 2006; Lacity et al., 2009), reduces uncertainty and provides costs saving (Lacity and Willcocks, 1998; Watjatraku, 2005), enables resources for other activities and reduces the amount of risk with the acquirement of ability and expertise by the agent who holds the benefits of the economies of scale and scope (Banker, 1989; Cachon and Haker, 2002; Bustinza et al., 2010).

There are situations which, rather than following the conventional thought, penalty clause in a typical outsourcing contract decreases the agent's intrinsic motivation to perform a task, providing incentives for the principal to commit even more the prohibited behavior. This situation has been called *crowding-out effect* by Frey and Jegen (2001), also known as the *overjustification effect* by the psychological theory of self-perception (Deci, 1971), implying that motivations may be undermine when no-monetary interaction becomes an explicit monetary one. The evidence for this effect when involves penalty is mentioned by Gneezy and Rustichini (2000) who analyzed the imposition of a monetary fine for late-coming parents in a group of daycare centers in Israel. The fine, thought to induce the parents' behavior into a reduction of the late, resulted in an opposite effect increasing the number of late-coming by the parents. Also, Gneezy (2003) applied a proposer-responder game to demonstrate that the proposer gives on average more money to the responder in pure situations (those with no reward or penalty relative to the amount given), than in moments when the responder can penance a fine over the value the proposer decides to give.

Despite the relevance of these evidences a model for this effect of penalty in contracts has never been formalized mathematically and the literature do not draw any conclusions whether this effect is visible in a contractual relationship between companies with different activities as it is an information technology outsourcing scenario. In fact, such an effect has been not receiving any attention in the discussion of the formulation of outsourcing information technology contracts. A pioneering model analysis and a posterior empirical research may add a fundamental value for the discussion when incentives may or may not work in a given type of contractual relationship, moreover, knowing what kind of incentives vendors and clients are subject may grant a great strategic advantage to the possessor of this information both for the formulation of their beliefs and actions, as to strategically influence the beliefs and actions of the other part. Also, the model we are about to draw may allow the decision making analyst to understand the best way to conduct the scenario, can minimize uncertainty and optimize the accuracy of the forecasts into an outcome and possible equilibrium in the game.

2. First Time Contract Model

According to the conventional software engineering literature (Boehm, 1981; Londeix, 1987; Boehm et al., 2000; Sommerville, 2011) the vendor's cost structure can be expressed as an exponential function of the overall effort provided. The cost structure for the agent do not usually increase linearly with software type (where it represents a high or low quality) due mainly to the side effects observed in the project testing phase: When changes in specifications or errors resolutions leads to another unexpected problems and errors, side effects of the original one. The result is that the more quality the software contract specifies, the more complex the project will be and more anomalies may occur during the development code phase. The sequential repeated game context infers that the agent is also concerning about her reputation and does not focus her strategy only on the currently payment, but also on the long term profit that later outsourcing contracts and re-contracts dealt with the same client may guarantee. The problem the agent faces is summarized by a choice of T, represented by:

$$\underset{T}{Max} U_a = P + P_f + T - C_f - C_\epsilon \frac{Q^b}{P_f T} LK \qquad (1)$$

Where C_ϵ is the vendor's cost sensitivity to the effort assigned by $(Q^b LK)/ P_f T$. The coefficient T is a function of the delivery date assigned in the contract (T_1) and the effective date the agent delivers the product (T_2) so that $T = T_2 - T_1$. A high T means that the agent took more time to deliver the software product, or delayed its delivery (high negative T_2) and a low T (even negative) has the meaning that the agent delivered the software product in less time (low T_2), since the delivery date (T_1) is fixed. The fraction $Q^b/P_f T$ is the ratio between the quality of the software system Q (observed by the vendor through the contract type) and the expected payment P_f weighted by the time the agent spends to develop the software system T. As the vendor notes the quality of the software is greater than the expected gain they might have, the ratio adds to the overall effort cost ($Q/P_f T > 1$), otherwise, when the vendor believes the expected payment will be greater than the endeavor of settle that type of software quality, the ratio reduces the effort cost ($Q/PT < 1$), and when both contributes similarly to the perception of the agent, only the technical parameters LK will matter for the effort-related cost ($Q/PT = 1$).

The exponent b clarifies the non-linear relationship previously described by the side effects of a development code and test phase, and it is usually estimated between 1 and 1,5 (Boehm et al., 2000; Sommerville, 2011). The exponent is sensitive to the novelty and risk of the project, it can be influenced from others parameters such as the bad performance of the development team in understand or use an existing code, and the fact $b \geq 1$ has the interpretation that positive changes in the product quality causes positive variations in the overall effort cost ($\partial C(e)/\partial Q > 0$) and that variation in the effort cost is increasing ($\partial^2 C(e)/\partial Q^2 > 0$). The parameters LK are constants that capture the developer's lack of experience, skills or the immaturity level of the development team, the unproductivity of the labor which we assign by L, and the lack of capabilities of the software tools together with the nonexistence of a reuse code, the capital's unproductivity which we assign by K. The agent still holds a fixed cost C_f over which must bear even without producing anything. The gain the vendor ensures with the outsourcing contract is composed by the payment the current project will guarantee, P, plus an expected profit counted from forthcoming contracts the client might outsource to this vendor, P_f.

The principal derives utility from the quality of the software system so that a contract that specifies a high quality software i. e. with many features or more comfortable interaction for the user is always preferable for the client than a contract which results a low quality product. The utility of the principal decreases as the agent attempts to delay the delivery of the software, but not in the same amount. A High Cost Client (HCC) is assumed to have more difficulties to work without the technology product of the outsourcing contract than a Low Cost Client (LCC). In other words, it is more costly for the High Cost Client to deal the same amount of business in the lacking of the software system requested than it is for the Low Cost Client, and for that reason, the High Cost Client values a shorter delivery time (with no delays) and a high quality software with more intensity than the Low Cost Client. Thus the problem the client faces is summarized by a choice of Q, once she does not have any power over the action of the agent into to delay or not delay, and it could be represented by:

$$\underset{Q}{Max} U_p = u_1 Q + u_2 G_f - u_3 T - P - P_f \qquad (2)$$

Where u_1 means how much the principal values a software quality type. A big u_1 for a high quality Q (or a small u_1 for a low Q) tell us that the client cares too much about the software product specifications and quality, what makes us assume we are probably dealing with a High Cost Client, whereas a small u_1 for a high Q (or a big u_1 for a low Q) make us induce that type of principal as a Low Cost Client since the client values fewer the software product quality. The parameter G is the expected gain the client might have with future outsourcing contracts and it increases with the current performance of the agent, i. e., the more quality the software system has in less time of development, more profitable it will be for the client now, and greater it is for the client the probability of been outsourcing the technological issues and infrastructure for a high productivity vendor, and higher is the future profit. The coefficient u_2 captures the level of importance that the principal gives for those expected profits whereas u_3 is the sensitivity of the client unto the delay of the project conclusion. It has the same meaning from the u_1 coefficient described above: A big u_3 for a high time of development T (or a small u_3 for a low T) tell us that the principal cares too much about the software product delivery time, what makes us assume we are probably dealing with a High Cost Client, whereas a small u_3 for a high T (or a big u_3 for a low T) make us induce that type of principal as a Low Cost Client since the valuation is small for software product time to release. In order to interpret the performance of the outsourcing contract we must look for the total social profit, obtained when we join the best response from both sides into a First Best Solution (Dey et al. 2010). Summing (1) with (2), the problem then turn into:

$$\underset{Q,T}{Max\,U} = u_1 Q + u_2 G_f + T - u_3 T - C_f - C_\epsilon \frac{Q^b}{P_f T} LK \qquad (3)$$

The First-Order Conditions (FOC) for the problem above are:

$$\frac{\partial U}{\partial T} = 1 - u_3 - \frac{C_\epsilon Q^b LK}{P_f T^2} = 0 \Rightarrow T = \left(\frac{C_\epsilon Q^b LK}{P_f - u_3 P_f}\right)^{\frac{1}{2}} \qquad (4)$$

$$\frac{\partial U}{\partial Q} = u_1 - b\frac{C_\epsilon Q^{(b-1)} LK}{P_f T} = 0 \Rightarrow Q = \left(\frac{P_f T u_1}{b C_\epsilon LK}\right)^{\frac{1}{(b-1)}} \qquad (5)$$

Those are the optimal choice for both the agent and the principal *a priori*. The First Best Conditions derived above gives us some expected but not uninteresting results. At first, equation (4) tell us that the best choice of T for the agent is positively related to her limitations on capital and labor (K and L), the quality of the software product (Q) and the sensitivity to the variable cost related to the overall effort (C_ϵ), what makes perfect sense, once a high quality software, a large gap in productivity or a high cost structure will persuade the agent to take more time to develop and hence delay the deliver. It is also negatively related to expected payment the agent might have with later contracts (P_f), and this concern adds to the choice of T more than itself because it is summed with itself weighted by a grade of importance the principal considers an increase (or decrease) in the time of development ($u_3 P_f$).

About the best quality choice the Equation (5) predicts that a change in the perception the principal has over the quality of the software (u_1) will affect positively the quality that will be demanded in contract. In other words, if the client cares more about the quality, higher quality software will be ordered. Also, if for some reason the client perceives that the vendor will have much to gain in future contracts (P_f) this will give a great bargain power to the principal who will increase the quality of the current one. We also see a positive relationship between the quality of the product and the time of development (T) because, *a priori*, a greater number of days to perform a service implies in the possibility of obtain a higher quality for the task than it would be if had a little time to do it. Thus, the longer the delay, the higher the quality demanded by the client at first. On the other hand, when the principal feels the agent will have much difficulties to develop the software ordered due to the limitation the vendor has in relation to his capital, techniques or workers (K and L) or due to the high cost of the agent (C_ϵ), she will not be able to order a high quality software system, choosing a low one instead (inversely relationship).

The difficult is increased by the parameter influenced by the novelty and risk of the software (b). We may notice that the client has a dominant strategy since his utility increases with the quality of the product, it will always be better to choose a high quality software than a low one, and since the agent has no power of treat. We use backward induction to verify the best response of the agent to the optimum choice of quality by the principal. Substituting (5) in (4) and assigning b = 1,5 (which is very close to the real estimations being in software engineering literature) we will have after all simplification process:

$$T = [\frac{b^3(C_\epsilon LK)^2 - u_3 b^3 (C_\epsilon LK)^2}{\left(P_f \ u_1^{\frac{3}{2}}\right)}] \quad (6)$$

Equation (6) reveals some important concepts concerning the behavior of the agent in a sequential game when considering, within the decision making, the optimal quality type of software chosen by the principal. First we observe that the best answer as a choice of development time T is negatively related to the expected gains the agent might have in the future and also to the degree of importance that the principal gives about the software quality u_1 This is not a surprising since the expected gains for future contracts grows with a good reputation now, so the best choice to increase these gains would be not to delay the delivery date, and, if it is possible, to further reduce the development time T to satisfy the expected growth in these gains. Likewise, when the utility of the principal increases for a given software the rational choice would be to provide the same software in the shortest time possible in order to keep satisfied the utility of the client and also increase the probability of doing business again (and so increase the expected gains).

About the upper side of the ratio, we have another logical proposition: the time in which the agent takes into development depends positively on their productivity needs and technical constraints, i. e. the deficiency expressed by the unproductivity of the workers, L, their equipment and development tools expressed by K and the variable effort cost of producing a given software, C_ϵ, weighted by the risk that a new project can bring, b. As long as the deficiency increases, the vendor will take a longer time to conclude and deliver the product software.

The more important consideration is the emergence of the parameter u_3 in the equation. Since $C_\epsilon KL$ are constants known by the agent representing the cost of producing that particular software and u_1 and P are known approximately when the principal chooses the type of quality of the software, we have the decision of the agent focused primarily on comparing the effect of parameter u_3, which represents the sensitivity of the client to changes in the development and delivery time of the software. The u_3 parameter reflects the intensity which the principal evaluates an anticipation in the date of delivery or a delay in the same date. For this parameter we assume 3 different situations that could interpret each type of principal we would be dealing with: If the parameter is very high (with value greater than 1), we know that the time for the principal is valuable enough and we can deduct we are dealing with a High Cost Client. If this parameter is too low (less than 1) we can deduct be dealing with a Low Cost Client, since this type of principal appreciates fewer the time the agent has probability because this client is highly productive and perform their tasks with greater ease and speed. Even for this type of principal (Low Cost Client) we have the situation in which this parameter can be equal to 1, which means that the Principal evaluates a delay (or advance) in the delivery date exactly equal to the value the agent has for her time. We discuss each of the three frames pictured above about the parameter u_3 and interpret its effect on the optimal choice of development time by the agent.

First we analyze the most simple situation, when the agent believes that the principal values highly the software development and delivery time (when $u_3 > 1$), which would imply a loss for the agent if she wishes to delay the delivery of the product, since the principal would not renew business with her. When $u_3 > 1$ we see that $(bC_I LK)^2$ - $(u_3 \, bC_\epsilon LK)^2 < 0$ for all nonzero positive $u_3 \, P_f$, which results in a delivery time $T < 0$, meaning that the value of the choice of the variation in the software delivery time for the agent $T = (T_2 - T_1)$ is in fact negative, in other words, the time that the agent effectively deliver product (T_2) should be shorter than the time signed in the contract (T_1) so that T is negative. Thus, if the agent believes that the principal values highly her time, then the agent's choice would be even anticipate the delivery date rather than delay it.

Assuming now the principal evaluation of time is supposed to be the same as the evaluation the agent has on her own time, that is, $u_3(T_1 - T_2) = (T_1 - T_2)$ resulting in $u_3 = 1$. This hypothesis can be quite feasible in our environment of incomplete or asymmetric information where one party knows nothing or almost nothing about the other party. When the agent estimates the principal has the same priority on the time as the agent has for hers the best response by the agent is not to delay the delivery of the software since the closer the estimates are to each other, closer to zero the upper side of the equation will approach and closer to zero is the change in T will be.

Finally, we observe the effects on the choice of T when $u_3 < 1$, e.g, when the agent believes that the principal does not matter very much about their time. From this point of view, the agent has no reason to believe that the principal is a Low Cost Client since the only information available to the agent is the quality of software, which it will always be high given the dominant strategy by the client. Thus, from one way or the other, in a sequential game of asymmetric information which the principal may understand about the agent's framework but not the otherwise, the agent has an intrinsic incentive to never delay a delivery of the product. The only reasonable hypothesis for an opposite behavior is when the agent has strategic information provided by the client or obtained in some other way that justifies the fact that both the principal valuation for the delay is small and the gains with futures contracts will not be diminished by the delay. This hypothesis shatters the scenario in which we find ourselves, and should be taken into consideration in the next section.

3. Penalty Contract Model

Let us now assume that the client afraid of a possible delay by the vendor (but not their incentives) wishes to impose a fine for each day the agent delays the delivery of the software system, counted from the date originally agreed. The model we are working with can now be represented as follows:

$$Max\,\underset{Q,T}{U} = u_1 Q + u_2 G_f - u_3 T + f(T) - P - P_f + \left[P + P_f + T - u_4 f(T) - C_f - C_\epsilon \frac{Q^b}{P_f T} LK\right]$$

Where the parameter f represents the fine that the client requires and must be paid by the vendor and u_4 means the vendor's sensitivity to the cost of the penalty. The first expression (outside the brackets) represents the profit to the principal while the second (inside the brackets) represents the profit to the agent. It was put the two expressions together into one so that we may seek the best portrayal for the total performance of this type of outsourcing contract ruled by clauses of penalties. The price of the fine has a constant value equal to f for the principal, the same cannot be said for the evaluation the agent makes to the same penalty. Given the sensitivity of the vendor to the price for the delay u_4, it is reasonable to assume that the degree of importance for the agent increases the lower the endowment of the firm and the greater their costs and the price of the fine. The optimal choice for both the agent and the principal in this contract can be found by the same way we did before. We use backward induction to verify the best response of the agent to the optimum choice of quality by the principal. Once the client signals to the agent her truthful evaluation through the imposition of a fine, there is no reason for us to keep thinking that this signaling would not update the strategy of the agent. A high penalty fine indicates that the principal highly values the time and a low penalty fine signals just the opposite. So we may assume the agent's best estimative of the cost of the time for the principal is the amount of the fine imposed, and the best choice of time by the agent is updated by this amount. Then, making $u_3 = f$, and $b = 1{,}5$ we will have after the simplifications:

$$T = \frac{b^3 (C_\epsilon LK)^2 - u_4 f[b^3 (C_\epsilon LK)^2]}{\left(P_f^2 u_1^{\frac{3}{2}}\right)} \qquad (7)$$

What equation (7) tell us is that the optimal choice for the vendor concerning the delivery time of the software is solely based on a comparison of their costs in relation to the fine imposed, everything else held constant. If the client imposes a fine f too high, and the vendor highly values the money they would have to spend as payment of such fine (a high u_4) then the equation turns to be negative ($- u_4 f [b^3(C_\epsilon L K)^2] > b^3(C_1 L K)^2$) and $T < 0$, meaning that it is likely the vendor will anticipate rather than delay the delivery date. Otherwise, if the client imposes a fine too low, and the vendor appreciates fewer the amount of the money spent as payment of such fine (a low u_4) then the equation turns to be positive ($- u_4 f [b^3(C_1 L K)^2] < b^3(C_1 L K)^2$) and $T > 0$, meaning that the vendor would delay the delivery date (occurrence of the *crowding-out* effect). Any variation between these two results will depend solely on the comparison made between the cost structure of the agent and how much that fine would harm the agent or support her.

Because now the agent knows for sure what type of principal she is negotiating, if the principal imposes a lower penalty fine (in the perception of the agent), the agent may delay and still maintain an expectation of future earnings from future outsourcing contracts with the same client. We have found an interesting approach of the consequence that the crowding-out effect may outcome over an asymmetric information sequential game: The fact that to impose a penalty over some irregularity such as a delay, as we have analyzed, may cause further emergence of such irregularity instead of make it disappear. In the outsource contract scenario it leads to a distortion of the previously observed balance of the game which the client chose a high-quality software as the dominant strategy and the vendor responded by not delaying as the dominant strategy. When the client attempts to guarantee that the vendor will not delay the delivery imposing a fine and thinking that, if before the fine the vendor was not likely to delay, now with the imposition of the fine the vendor will not delay for sure, this conclusion does not take into account the incentives arising from the information asymmetry in this kind of game. The client thoughts of to depart the chances of delay by the vendor is instead giving the agent the opportunity to delay by signaling her cost structure to the vendor, who now updates her optimal strategy which may be to delay the delivery date of the software product, when sooner she would not.

4. Conclusions

The model we just built deduced that an outsourcing of information technology characterized as a sequential game of incomplete information, with potential future gains for both parties, culminates in the principal, i. e. the client, having a dominant strategy which is to choose a contract of high quality software, since the agent, i. e. the vendor, would always strive to ensure that quality for the client in order to maintain a good reputation to future contracts and because the absence of adequate short-term incentives to delay the date of delivery of the software. Given the uncertainty of the vendor to know what kind of client they are dealing with (whether high or low cost), the expected payoff made by the current payment plus a future value that are expected according current performance of the vendor will be higher when the agent do not delay the delivery of the software product. So, unless the agent has strategic information provided by the client or obtained in some other way that justifies the fact that both the principal has a lower value for her time and the gains to the agent from futures contracts would not be diminished with a delay, the vendor will never delay the software delivery in this sequential game. If the agent has such information about the type of principal, outsourcing will no longer be characterized as a sequential game of incomplete information, and the agent delay strategy may be drastically modified by the reason of it being no longer dealing with expected payoffs, but now with certain payoffs.

We have demonstrate that the imposition of a fine works as a sign to the agent about the cost structure of the principal and thereby her type (whether high or low cost), which can often result in a crowding-out effect on the incentive structures where this grade of penalty may increase the delay, rather than eliminate it, because now the agent updates her estimations about the type of the client according to the price of the fine and the value of the fine for herself (that not necessarily must be equal to the price of the fine for the client). If the fine is too high in the perception of the client but is not that expensive for the vendor, or if both has a low valuation about the price of the fine, the agent would delay the delivery, otherwise, if this fine is too high in the perception of both (but not high enough to be considered as unfair and breaks the contract) or if it is expensive for the vendor but not that much for the client, the vendor agent would not delay the delivery of the software product. The conclusion our formal model derives comes from the same conclusion Greiff et al. (2013) took about the strategic value the information asymmetry can generate. In their study the authors conclude that the price that consumers are willing to pay for a good or service on a Pay What You Want pricing scenario depends on how much information these consumers possess regarding the costs of producers. If the supplier of a good or service has low costs, the best strategy for her is to not reveal any anchor price so that consumers have no idea how much they must pay. If the cost structure of the producer is high, this producer will raise more money revealing a reference price for the consumers. The same analogy could form our independent outsourcing scenario: If the client observes her nature as a Low Cost Client, the best optimum choice would be to choose a contract with no penalty clauses in order to avoid a possible delay by the vendor. If the client is a High Cost Client, the imposition of a fine may be the best strategy. The same analysis can be made into any of these factors: a monetary value, number of days of delay or quality of the product and the conclusion shall not be affected. However, an empirical approach to this effect in real IT outsourcing contracts and services outsourcing contracts should add sufficient value to the theoretical gap our model leaves open. Camlas Consulting, a firm specialized in management and contracts review for IT outsourcing in Europe, found IT services contracts where it was cheaper for the supplier to pay a fine than putting time, effort and money to meet the agreed service levels. In fact, a more practical framework for this effect in outsourcing contracts must be a feasible suggestion for coming works.

References

Banker, R. D., and Kemerer, C. F. Scale Economics in New Software Development. *IEE Transaction on Software Engineering*, 15, No. 10, 1199 – 1205, 1989.

Boehm, B. W. Software Engineering Economics. *Englewood Cliffs*, NJ: Prentice Hall. 1981.

Boehm, B. W., Abts, C., Brown, A. W., Chulani, S., Clark, B. K., Horowitz, E., Madachy, R., Reifer, D. and Steece, B. Software Cost Estimation with COCOMO II. *Upper Saddle River*, NJ: Prentice Hall. 2000.

Bustinza, O.F., Arias-Aranda Gutierrez-Gutierrez D., L. Outsourcing, competitive capabilities and performance: an empirical study in service firms. *International Journal of Production Economics* 126 (2): 276-288, 2010.

Cachon, G., and Haker, P. Competition and Outsourcing with Scale Economies. *Management Science* 48(10): 1314 – 1333. 2002.

Debabrata Dey, Ming Fan, Conglei Zhang. Design and Analysis of Contracts for Software Outsourcing. *Information Systems Research* 21(1):93-114. 2010.

Deci, E. L. Effects of externally mediated rewards on intrinsic motivation. *Journal of Personality and Social Psychology*, 1971, 18, 105-115.

Eisenhardt, K. M. Agency theory: An assessment and review. *Academy of Management Review*, 14(1), 57. 1989.

Frey, Bruno S., and Reto Jegen. Motivation Crowding Theory. *Journal of Economic Surveys*, 15: 589-611. 2001.

Gneezy, U. and A. Rustichini. A fine is a price, *Journal of Legal Studies*, 29, 1-17. 2000.

Gneezy, Uri.. The W Effect of Incentives. *Mimeo. The University of Chicago Graduate School of Business*. 2003.

Greiff, Matthias and Egbert, Henrik and Xhangolli, Kreshnik. Pay What You Want – But Pay Enough! Information Asymmetries and PWYW Pricing. *Joint Discussion Paper Series in Economics by the Universities of Aachen*. 2013.

Jensen, Michael C. and Meckling, William H. Theory of the Firm: Managerial Behavior, Agency Costs and Ownership Structure. *Journal of Financial Economics (JFE*), Vol. 3, No. 4: 305 - 360,1976.

Keil, P. Principal Agent Theory and its Application to Analyze Outsourcing of Software Development. *Proc. of the Int. Workshop on Economics-Driven Software Engineering Research (EDSER)*, St. Louis, USA, IEEE Computer Society. May 2005.

Lacity, M.C., Willcocks, L.P. An empirical investigation of information technology sourcing practices: lessons from experience. *MIS Quarterly* 22 (3), 363–408, 1998.

Lacity, Mary C., Khan S. A., Willcocks, L. P. A review of the IT outsourcing literature: Insights for practice *Journal of Strategic Information Systems* 18 (2009) 130–146

Londeix, B. Cost Estimation for Software Development. *Wokingham: Addison-Wesley*. 1987.

Mookherjee D. and Ivan Png. Optimal Auditing, Insurance, and Redistribution.*The Quarterly Journal of Economics*, vol. 104(2), pp. 399-415, 1989.

Nosenzo, Daniele; Offerman, Theo, Sefton, Martin and van der Veen, Ailko. Discretionary Sanctions and Rewards in the Repeated Inspection Game. *CeDEx Discussion Paper* No. 2012-10.

Singer, Marcos and Donoso, Patricio. *Contracting contractors. Journal of Business Research,* Vol. 64, Issue 3,Pp. 338-343, 2011.

Sommerville, Ian. Software engineering 9th Ed. *Pearson Education, Inc*. 2011.

Watjatrakul, B. Determinants of IS sourcing decisions: a comparative study of transaction cost theory versus the resource-based view. *Journal of Strategic Information Systems* 14 (4), 389–415, 2005.

Willcocks, L., Feeny, D. IT outsourcing and core IS Capabilities: challenges at lessons at DuPont. *Information Systems Management* 23 (1), 49–56, 2006.

Whang, S. Contracting for Software Development, *Management. Science*, 38, 2, 307-324. 1992.