PM and network

Matteo Boidi[1,2]; *Paolo Demagistris*[2]; *Enrico Maria Fenoaltea*[3]; *Fabrizio Monge*[4]
1 ESPC Business School
2 Politecnico di Torino, Department of Management and PRoduction engineering, Torino
3 University of Fribourg, Physics Department, Chemin de Muse 3.
4 Associazione Erminio Bullio, Torino

In the last three decades the project delivery methodology has been broadened to consider alternatives to the traditional predictive delivery framework as embodied in PMI's PMBoK. And as alternative processes, techniques and tools are available, so the project practices of designing and organizing the project have emerged more important, as in IPMA ICB. However no systemic quantitative approach has been proposed to correlate by models those subprocesses output with project success indexes. PMI suggests that project success correlate positively with the so called PM3, project management maturity model. So the higher the maturity the higher the frequency of projects delivering goals on time and budget, and the lower the frequency of project failures, where budget is lost. In the *project management maturity model*, however, there is nothing quantitative and the maturity is reached with "failure cycles" as the system is not proactive but reactive. The purpose of our research is to establish a quantitative method that can determine the evolution of the project.

An approach that seems promising is to exploit the *actor network theory* (ANT): the agents of a socio-technical system (people, institutions, artifacts etc.) live within an interacting and interconnected system that can be represented through a network, therefore they are considered as the nodes of this network which are connected to each other through edges and in turn these edges represent the relationships between the agents of the system.

What kind of relationships between the agents of the governance of a project can be mapped as edges in our network? One way to proceed, since we are in the economy field, is to consider any **transactional relationship** as edges:

- **Capital transactions** (ex. investor or bank and dcf)
- **Standard economic transaction** (ex. buy and sale)
- **Knowledge transaction** (ex. any information exchange, such as the provision of permits etc.)
- **Economic reports** (ex. input and output)

Within a socio-technical network, however, there are also **behavioural relationships** between the agents, but since these are related by human actors' bounded rationality, they are not easily modeled. These relationships can be treated at a later time as a second order effects.

Another fundamental characteristic of this type of network is that it is not static but evolves over time ($G = G(t)$), that is, if photographed at each instant of time it will have a different configuration. It is more convenient to consider the time istantes discrete: the initial network $G(t = 0)$ represents the system in the *initiate* phase, that is, in the phase in which the project resolution is being prepared; at the next instant the network $G(1)$ represents the *planning* phase and so on.
At this point the crucial questions that this research asks are the following:

1. What is the most convenient way to map the project management process in a network?

2. Once the first question is resolved, how important is what happens in the transition from one network to another $G(t) \to G(t + 1)$?

Ideas and motivations for question 1

Mapping a network for project management processes is fundamental both for viewing and for managing information within the process itself. Such socio-technical processes are often characterized by a high degree of complexity and it is no longer conceivable to attempt to manage them through linear approaches.
Realizing a protocol for the construction of such a network in a specific case could become a useful resource for the construction of a standard protocol in the construction of any managerial network.
The mapping is not unique: there are several ways to deal with the characteristics explained above. For example, the various relationships between agents can be treated in an equivalent manner and be positioned within the same network, or a network can be built for each type of relationship (one for knowledge transactions, one for economic transactions and so foth), in this case it would be a *multi-layer network* and would have completely different structural characteristics (this is the version that attracts me most, about this I suggest reading https://arxiv.org /abs/1308.3182); furthermore, it is necessary to decide whether to treat the edges so that they are directional or not, and here too the network would assume very different characteristics; still, nodes can be treated differently depending on their function and so on. So as one can notes, the construction of the network is arbitrary but must be done in such a way as to minimize the amount of information dispersed and therefore needs some attention.

Ideas and motivations for question 2

At this point, once the characteristics of the network are known, its structural metrics must be measured: in its simplest form, a network is characterized by various metrics such as *density*, *degree distribution*, *diameter* etc.
Now, the value of these metrics is not a judgment of the goodness of the network but is simply a description of it. What matters is the dynamics of these metrics, in fact, just as the network varies its shape, they will change value at every moment of time. Question 2 can therefore be formalized as follows: given a metric $d = d(t)$, what does its change in time tell us $\Delta d = d(t+1) - d(t)$? Or also, what does the variation of its variance tell us $\Delta\sigma_d^2 = \sigma_{d(t+1)}^2 - \sigma_{d(t)}^2$?
Normally, change is associated with risk. Therefore, if a certain acceptability threshold S is appropriately defined, then if $\Delta d >> S$ then the probability that the project fails is very high.
A possible development of such an approach could be to decide, given a certain Δd, which is the best strategy to transform the network at the next time in such a way that the risk of failure (or the loss of an investment) is the minimum possible. For example, given a certain Δd, it is convenient to adopt a strategy of *avoiding*, *transfering*, *mitigating* or *accepting* ?.
In addition, the next step could be to make forecasting: that is, to establish what is the probability of failure given the dynamics or correlation of certain metrics, for example one could see if there is a "point of no return" beyond which the failure or the success are unavoidable.

Methodology

Obviously the goal is that the model is built as close as possible to the real world outcome and once the model is made fairly reliable (with an acceptable ratio (model results) / (real results)), one can try to refine it to make it really predictive.
To verify the reliability of the model it is necessary to clean it through "control parameters" in the real world on which to check its reliability (through the classic "statistical tests"). An attempt is therefore made to make the model dataset "controllable" in such a way that its outcomes are independent of exogenous factors or externalities (such as the second-order effects discussed above). In more familiar terms to the scientific world, we try to make the model normalizable to external conditions. The most important control parameters on which to normalize the model are:

- **Industry**
- **project size**
- **project lifecycle duration**

- and others to be added later

The approach of this research is purely **empirical** in that the $risk = change$ conjecture is taken for granted and experiments are performed to solidify this conjecture. The study is performed on a **population of projects at least 50 unities**.

As already mentioned, the model consists in the construction of a network, therefore the inputs in this network are necessarily the nodes (the actors within the management process), the edges between these nodes (the transactional relationships mentioned in the previous sections) and, in the event that one wants to build a *weighted network*, the quantity of these edges (in terms of capital, price, hours etc.). In particular, the basic inputs are:

- **capex transaction**, quantified by capital transacted
- **opex transaction**, quantified by price transacted
- **labour transaction**, quantified by men/hours
- **information transaction**, semi-quantified by ranking in criticality

These inputs must be given in discrete time stages, in particular we want to build a network for each canonical stage of management process: *casing, initiating, planning, execution* and *project*. For convenience, let's call these stages with the following notation: $T = \{t_1, t_2, t_3, t_4, t_5\}$. Obviously each stage t_i is not totally static, but we make the reasonable approximation that, if d is any metric and δ is the difference of the metric value between two networks belonging to the same stage, then it is valid

$$\delta d << \Delta d \, ,$$

and therefore we only consider the phase transitions between one canonical stage and another.
With this type of data one can build two types of networks and therefore you will have two types of outcomes

- one can build a *temporal multilayer network*, i.e. a five-layer network in which these layers are precisely the canonical stages (it is not excluded that the network of each of these stages is itself a layered network) and in this case the output of the model is a vector representing the static metrics d analyzed.
- or, 5 independent *networks* can be built, each representing a different canonical stage and in this case the model outputs will be vectors representing the Δd analyzed.

In both cases the outputs must be normalized according to the control parameters described above. The ultimate goal is to find a causal link between the model's outputs and real outcomes through, for example, a regression (appropriately considering the control parameters). The usefulness of the results is to see, as already said, the probability of failure/success $P(f/s|\{d_1, d_2, ..., d_n\})$ conditional on the network metric currents. In other words, a curve is sought that describes the expected outcomes as a function of the structure of the network.

Furthermore, this curve could be used to decide which management strategy to adopt once aware of the structure of the underlying socio-technical network and so the future of this research is likely to fall into machine learning.