



OPEN

Activity networks determine project performance

Alexei Vazquez[✉], Iacopo Pozzana, Georgios Kalogridis & Christos Ellinas[✉]

Projects are characterised by activity networks with a critical path, a sequence of activities from start to end, that must be finished on time to complete the project on time. Watching over the critical path is the project manager's strategy to ensure timely project completion. This intense focus on a single path contrasts the broader complex structure of the activity network, and is due to our poor understanding on how that structure influences this critical path. Here, we use a generative model and detailed data from 77 real world projects (+ \$10 bn total budget) to demonstrate how this network structure forces us to look beyond the critical path. We introduce a duplication-split model of project schedules that yields (i) identical power-law in- and-out degree distributions and (ii) a vanishing fraction of critical path activities with schedule size. These predictions are corroborated in real projects. We demonstrate that the incidence of delayed activities in real projects is consistent with the expectation from percolation theory in complex networks. We conclude that delay propagation in project schedules is a network property and it is not confined to the critical path.

Delivering projects on time and on budget is necessary to improve human prospect¹, with the World Bank stating that 22% of the world's gross domestic product—about \$48 trillion—relies exclusively on project-based delivery mechanisms². Yet the majority of public and private large-capital projects are completed late and over budget³. An industry survey captures the scale of the problem—reviewing 10,624 projects from 200 companies in 30 countries and across a variety of industries, it concludes that only 2.5% of projects were completed on time and budget⁴. A recent review reaffirms the stubbornness of the challenge, with delays remaining at comparable levels even after 15 years of project management advancements (comparing projects started between 1998 and 2003 vs. 2013–2018)⁴.

This consistency in poor performance suggests that the core method of evaluating delay risk is inadequate for the complex nature of modern projects. Known risk events can be identified, analyzed and responded using risk management plans during the planning phase. However, unknown risk events deteriorate the project performance.

Since the 1960s project managers have almost exclusively relied on monitoring the critical path as the means to manage delay risk. This path is essentially a sequence of activities from start to end that are executed without any slack time in between^{5,6}. The critical path activities play a key role in the scheduling of limited resources and the delay risk analysis.

An increase in the duration of any activity in the critical path causes project end overrun. It is a simple concept and it provides a simple solution: the critical path must be executed as planned at all costs. Yet, modern projects are more complex, with schedules that look like complex networks of activity dependencies^{7,8}. Delays in activities outside the critical path can similarly cause project end overruns through domino-like cascades, similar to how viruses spread⁹.

Given the consistency in project delays over the past decades, we examine the limit of applicability for the critical path using both synthetic and real data. We find that, beyond a certain level of complexity, the critical path becomes irrelevant and project end overruns are primarily driven by activities that are outside of that path.

Results

Generative model of project schedules. A project schedule is generated using a standardised procedure. In that process planners take into account the state of the art of contractors operations. If specialization occurs and the work of a former contractor doing activity A is now carried on by two contractors doing activities A1 and A2, then we would experience a change of A to A1 and A2 when looking at schedules before and after this specialization.

The evolution of project schedules (or activity networks) in time can be seen as the outcome of a growth process, where a parent activity can be duplicated or split (Fig. 1A). Generic activities can be duplicated and

Nodes & Links Ltd, Salisbury House, Station Road, Cambridge CB1 2LA, England, UK. ✉email: alexei@nodeslinks.com; christos@nodeslinks.com

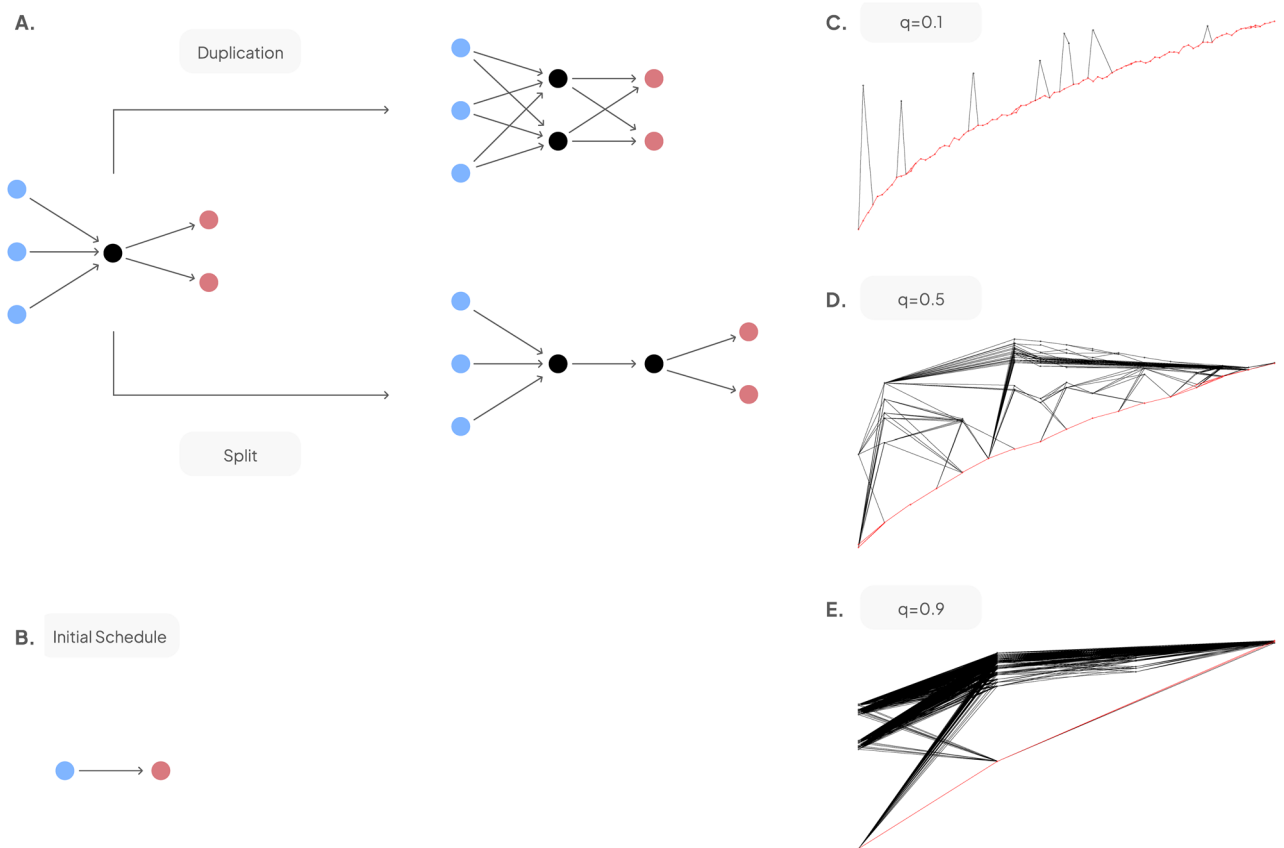


Figure 1. Duplication-split model. (A) Duplication and split rules acting on an activity (black). (B) Initial conditions where the duplication-split model is applied. (C) Examples of activity networks generated by the duplication-split model for three different values of the duplication probability q . Critical path is highlighted in red.

broken into two smaller activities that run in parallel, both inheriting all predecessors and successors of the parent activity. Specialised activities can be split into two activities executed in sequence, such that one specialised contractor executes the first part and another the second.

Starting from two activities executed in sequence (Fig. 1B), we can grow the network by a stochastic sequence of duplication and split events, with a probability of duplication q . For small q , activities will be mostly split, generating a mostly linear activity network. For large q , most activities are duplicated, leading to a network with numerous parallel paths (Fig. 1).

Node duplication, also known as copying, has been studied in the context of web networks and protein interactions networks^{10–13}. It has been shown that node duplication generates networks with a power law probability distribution in the number of links associated to a node^{10–13}. In the Methods section we demonstrate that this is indeed the case for our model of duplication-split activity networks, but with a twist. We can show that the distributions of the number k of predecessors and successors to an activity follow the same power law $p_k \sim k^{-1/q}$, where p_k is the probability that an activity has k predecessors (or successors). Our calculations are validated by numerical simulations of the duplication-split model (Fig. 2).

Once we create activity networks, we can populate synthetic project schedules by assigning durations to each activity. We now have project schedules with a critical path, a sequence of activities from the start to the end of the project. The latter carry as a consequence that delaying the finish of any activity in the critical path delays the project end date by the same amount.

Shrinkage of the critical path. Critical path is the perceived centrepiece in project management due to its sensitivity to delays. Yet, a look at the synthetic activity networks in Fig. 1C–E made us question whether that critical-path-centric view is valid for modern projects, given that modern projects have complex structures with many parallel paths of work happening at the same time⁶.

In cases where activity networks are quasi-linear, the critical path is indeed the dominating structural feature (Fig. 1C, $q=0.1$). In contrast, in the $q=0.9$ activity network we observe a large number of parallel paths with similar number of activities (Fig. 1E, $q=0.9$). It is in these cases that the concept of the critical path may be of less relevance to manage the delay risk of the project.

Following these qualitative observations, we show that the larger the project network, the smaller the relative size of the critical path. Furthermore, the larger the duplication probability q , the smaller the relative number of activities in the critical path, in agreement with the visual inspection of the $q=0.1$ and 0.9 synthetic activity

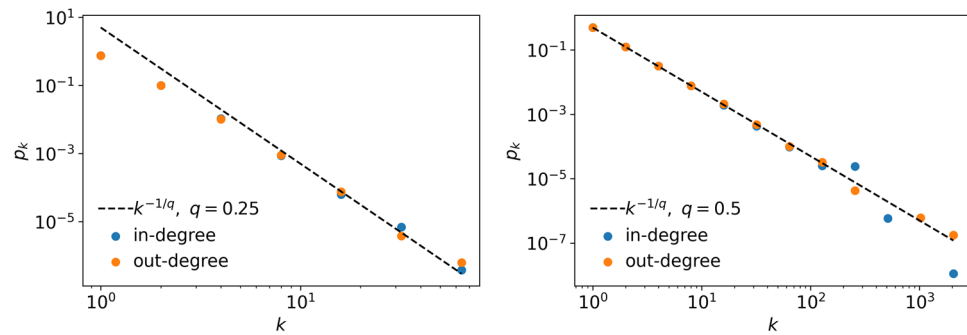


Figure 2. Degree distributions. Distribution of the number of predecessors (in-degree) and successors (out-degree) across activities in networks generated by the duplication-split model, using (A) $q = 0.25$ and (B) $q = 0.5$. The dashed line highlights the power law tail predicted by our calculations.

networks in Fig. 1C,E. We determine the number c of critical path activities in a network of n activities and duplication parameter q . We estimate that $c \sim n^{1-q}$ and therefore the fraction of activities in the critical path decreases as $c/n \sim n^{-q}$. Numerical simulations corroborate the $c/n \sim n^{-\alpha(q)}$ scaling, albeit with $\alpha(q) \leq q$ (Fig. 3).

We note that the duplication-split networks are not small-world networks^{14,15}. In small-world networks the typical distances between nodes scale logarithmically with network size ($c \sim \ln n$)¹⁵. Duplication-split networks are a new class of networks with power-law degree distributions and power-law scaling of node distances with network size. In fact, these are fractal networks ($c \sim n^{1/D}$), with a fractal dimension $D = 1/(1 - \alpha(q))$.

Vanishing critical path in empirical activity networks. To demonstrate that our observations are representative of the real-world challenge, we shift our focus to empirical data for 77 construction projects (total value + \$10bn), with activity networks representing different stages of the project lifecycle, adding up to 323 project schedules. These activity networks vary in size, from 100 to 16,000 activities.

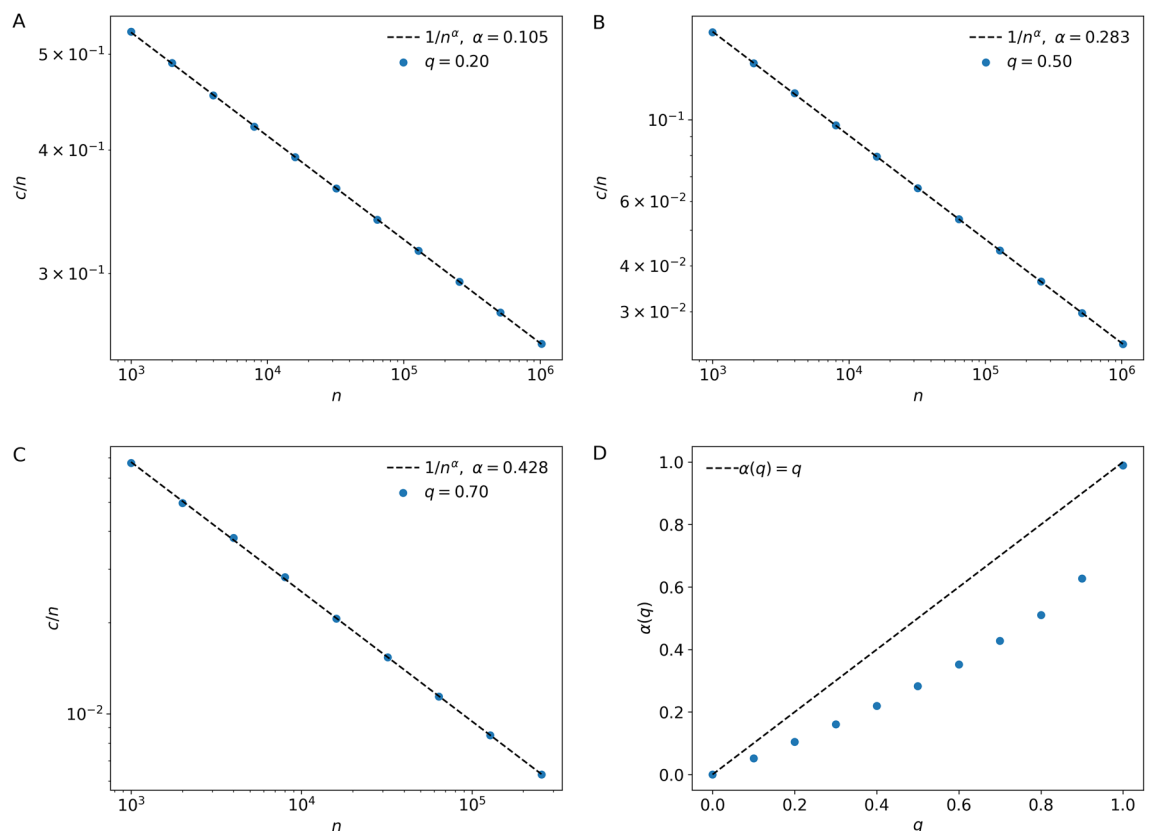


Figure 3. Relative size of the critical path. (A–C) The number of activities c in the critical path relative to the total number of activities n , for activity networks generated by the duplication-split model. The dashed line highlights the power law decay predicted by our calculations. (D) The scaling exponent of the power law decay $c \sim n^{-\alpha}$ as a function of q (symbols). The dashed line is the theoretical upper-bound.

Driven by our synthetic schedule analysis, our prediction that the relative size of the critical path decreases as the number of activities increases is further confirmed in the empirical data.

First, we corroborate the distribution of the number of predecessors (in-degree) and the number of successors (out-degree) to an activity are almost identical and they follow a power law decay (Fig. 4A). Assuming the power-law decay of the duplication split model, we obtained a maximum likelihood estimate q from the distribution of the number of predecessors and independently from the number of successors. The duplication-split model predicts that the two should coincide. Indeed, the data for the construction projects fall at or in the vicinity of the equality line (Fig. 4B). Furthermore, the duplication q index of real projects is distributed between 0.1 and 0.5, with most values between above 0.2 (Fig. 4C).

Second, we tested the $c/n \sim n^{-\alpha}$ scaling of the fraction of activities in the critical path. The fraction of activities in the critical path c of real activity networks decreases as the number of activities n increases (Fig. 4D, blue symbols). This decrease approximately follows the scaling $c/n \sim n^{-\alpha}$ with $\alpha = 0.79$.

Network complexity drives delay risk. Now we switch our attention to delay propagation in activity networks. Exogenous delays such as extreme weather events, pandemics or financial crises can cause some activities to be delayed beyond their planned finish date. When activity delays exceed the spare time between activities (free floats) they propagate downstream triggering a delay cascade. We view activity delays exceeding the free floats as microscopic events and the delay cascades reaching the project end as the macroscopic behaviour. The microscopic events are quantified by the probability p that an activity dependency will transmit a delay. The macroscopic behaviour is quantified by the fraction of activities where the activity delay exceeds its total float. We call the latter the delay incidence.

If the critical path is a key delay risk factor, then the incidence of delay across activities should increase with increasing $p \times c$, where c is the critical path size as denoted above. However, when we plot the delay incidence vs $p \times c$ we actually observe a negative non significant correlation (Fig. 5A, Pearson correlation coefficient = -0.1, significance = 0.7). Therefore, the delay risk is not determined by the critical path size.

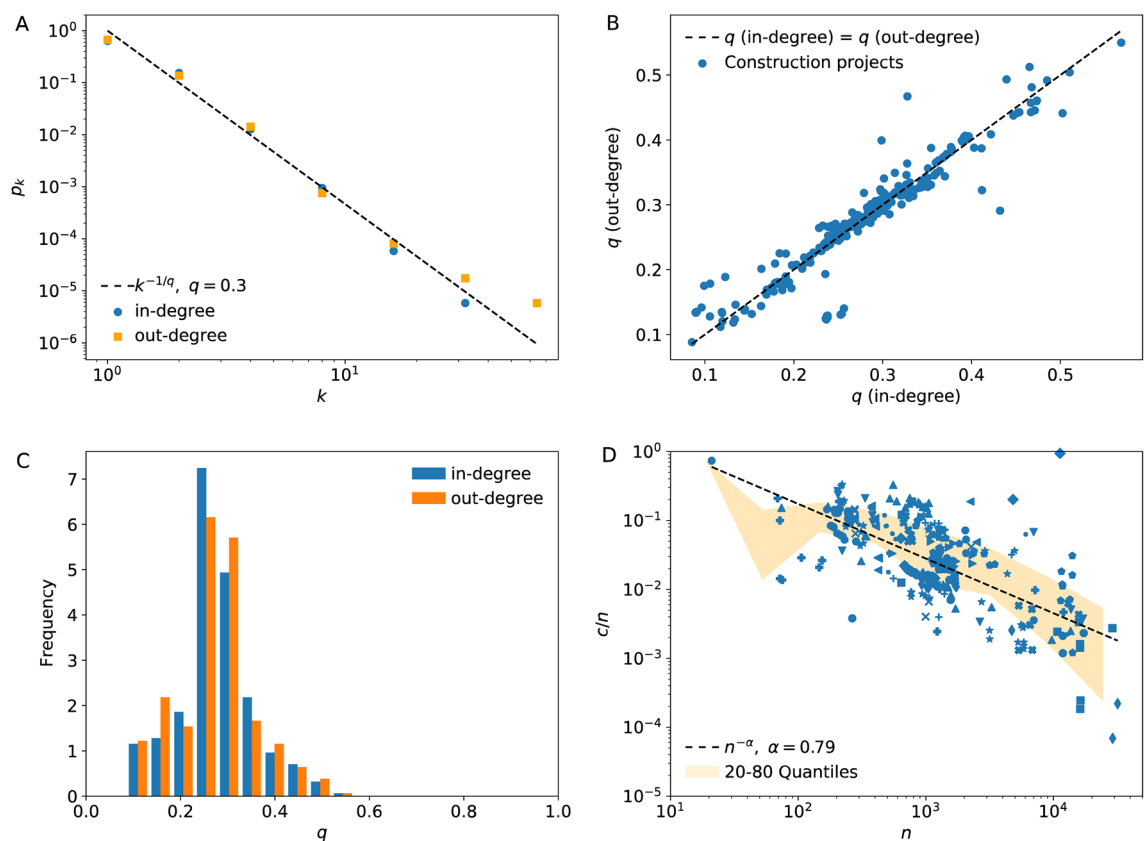


Figure 4. Empirical data analysed. (A) Distribution of the number of predecessors (in-degree) and successors (out-degree) across activities of a typical construction project. The dashed line highlights a power law fitting the data. (B) The exponents q obtained from the fit to the distribution of the number of predecessors $q(\text{in-degree})$ and successors $q(\text{out-degree})$. Each point represents a project schedule. The dashed line is the equality line. (C) The distribution of estimated duplication rate q across projects. (D) The fraction of activities in the critical path as a function of the number of activities. Different symbols represent a project schedule at different stages of completion. The solid background represents the [20,80]% confidence interval.

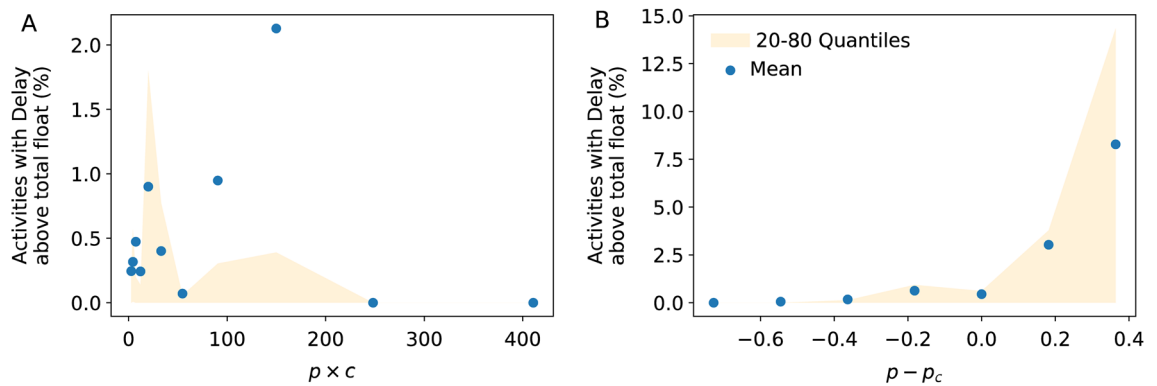


Figure 5. Delay incidence. (A) Observed delay incidence in construction projects as a function of $p \times c$, control parameter associated with the critical path. (B) Observed delay incidence in construction projects as a function of $p - p_c$, the control parameter of percolation theory. p is the fraction of delay transmissions along direct activity-activity relations, c the number of activities in the critical path and p_c is the critical threshold from percolation theory ($p_c = 1 / \langle k \rangle$).

If the critical path vanishes for large projects, and we know that almost all complex projects are delayed, where does this risk come from? After ruling out the standard hypothesis (critical path) we shift our focus to activities outside the critical path.

We use percolation theory as a framework to help us quantify the propensity of the project to exhibit a delay, driven by delays at the activity level^{16–18}. Bond percolation indicates that when p exceeds a critical threshold p_c delay cascades will take place with a finite probability. For directed networks with uncorrelated in-degrees and out-degrees $p_c = 1 / \langle k \rangle$ ¹⁸, where $\langle k \rangle$ is the average out-degree. Percolation theory predicts a phase transition from no macroscopic cascades when $p < p_c$ to a finite risk of macroscopic cascades for $p > p_c$.

This is exactly what we observe for real project networks (Fig. 5B), highlighting that project end overruns are indeed a property of the whole network. For $p < p_c$ the delay incidence is below 1%, almost no risk of project delay. In contrast, for $p > p_c$ the delay incidence increases gradually, and in some cases impacting 15% of the entire project. We note that for some projects with $p > p_c$ the delay incidence is below 1% and the confidence interval reaches zero (Fig. 5B, orange band, $p - p_c > 0$). This is expected from percolation theory. The occurrence of macroscopic events is probabilistic. What is different from zero is the probability that such macroscopic events occur.

Conclusions

We focus on activity networks that describe large-capital projects, showing that their broader structure contains information about their propensity for delays. Our first contribution is the introduction of the duplicate-split model, and the fact that the duplication index q is a core feature of activity networks. Networks with small q are indicative of quasi-linear topologies, and a good fit for using the critical path. Large q indicates a complex project, where the critical path is relatively small, and parallel paths tend to dominate the overall structure. We then use synthetic and empirical data to both validate the output of the duplicate-split model. Our second contribution is showing that the number of activities in the critical path decreases as $n^{-\alpha}$ and therefore the critical path vanishes in the limit of large activity networks. As a result, the critical path is of limited applicability when it comes to large and complex projects. Our third contribution is the application of percolation theory in order to go beyond the limitations of critical path analysis, whilst showcasing that project end overruns are a network property.

Methods

Estimation of the degree distribution. Let $n_k(n)$ the number of activities with k predecessors in the activity network. As new activities are added $n_k(n)$ changes according to the equation

$$n_k(n+1) = n_k(n) + q \left[\frac{k-1}{n} n_{k-1}(n) - \frac{k}{n} n_k(n) + \frac{1}{n} n_k(n) \right] + (1-q) \delta_{k1} \quad (1)$$

The first term inside [...] corresponds to activities with $k-1$ predecessors and the duplication of one predecessor with probability $(k-1)/n$, moving to the k predecessors group. The second term inside the [...] is the same but for activities with k predecessors, moving from the k predecessors group. The third term inside [...] is the chance that an activity with k predecessors is duplicated, thus generating a new activity with k predecessors. Finally, the last term in (1) is the creation of a new activity with one predecessor following a splitting event, where $\delta_{k1} = 1$ if $k = 1$ and 0 otherwise (Kronecker delta symbol).

Assuming a steady state solution we obtain

$$p_k = q[(k-1)p_{k-1} - kp_k + p_k] + (1-q)\delta_{k1}. \quad (2)$$

We can iterate this equation to obtain an expression for all $k > 1$ as a function of p_1

$$p_k = \frac{q(k-1)}{1+q(k-1)} p_{k-1} = \prod_{s=1}^{k-1} \frac{s}{\frac{1}{q} + s} = \Gamma\left(\frac{1}{q} + 1\right) \frac{\Gamma(k)}{\Gamma(\frac{1}{q} + k)} p_1 \quad (3)$$

where $\Gamma(x)$ is the gamma function. For $k \gg 1$ the later equation has the asymptotic behavior

$$p_k \sim k^{-1/q} \quad (4)$$

The same reasoning can be repeated using k as the number of activity successors. That is, the distributions of the number of predecessors (in-degree) and successors (out-degree) are identical in the $n \rightarrow \infty$ limit.

Estimation of the critical path size. Consider a network schedule with n activities and c activities in the critical path. As new activities are added, the size of the critical path can increase if a task in the critical path is subject to the split rule. Since the split rule is executed with probability $1-q$ at each activity addition and the probability that the activity selected for splitting is in the critical path is equal to c/n , then

$$\frac{dc}{dn} = (1-q) \frac{c}{n} \quad (5)$$

Integrating this equation we obtain

$$\frac{c}{n} \sim n^{-q} \quad (6)$$

This result is an approximation. As the network grows there could be changes in what activities are in the critical path, making the critical path shorter. We conjecture the scaling $c/n \sim n^{-\alpha(q)}$, where $\alpha(q) \leq q$.

Python code for the duplication-split model.

```
import numpy as np
import networkx as nx

def duplication_split_digraph(n,p):
    np.random.seed(0)
    G = nx.DiGraph()
    G.add_nodes_from(range(n))
    G.add_edge(0,1)
    duplicate = np.random.random(n) < p
    for i in range(2,n):
        j = np.random.randint(i)
        G.add_edges_from([(i,k) for k in G.successors(j)])
        if duplicate[i]:
            G.add_edges_from([(k,i) for k in G.predecessors(j)])
        else:
            G.remove_edges_from([(j,k) for k in G.successors(j)])
            G.add_edge(j,i)
```

Generative model simulations. Project schedules are generated in three steps. (1) We generate an activity network by successive application of the duplication/split rules up to we reach n activities. At each activity addition we select an activity with equal probability among all current activities in the network, execute the duplication rule with a probability q otherwise the split rule. (2) We assign a duration x to each activity from a distribution with probability density function $f(x)$. Here we use an exponential distribution with mean 1 day. (3) We assume that all activity relations are of the standard Finish-Start type, that all activities with no predecessors start at day 0 and apply forward/backward passing^{6,7} to determine the early and late start and end dates for all other activities. Average statistics and distributions are estimated from 100 simulations of these steps for each set of parameters (n,q) .

Critical path. Once a schedule has been generated, we perform a second backward pass to calculate the total float of each activity. The total float is defined as the amount of time that the end date of an activity can be postponed without affecting the project end date^{6,7}. The critical path is the set of activities with total float 0 and it will be denoted by C . The size of C , the number of activities in the critical path, is denoted by c .

Probability of delay transmission. We estimate the probability p that an activity dependency will transmit delays by looking at all completed activities, and computing the fraction of dependencies with slack time that is smaller than the delay at the parent activity, across all relations out-going from finished activities.

Control parameter of the critical path method. The probability that there are no delay transmissions in the critical path is $P(p,c) = 1 - (1-p)^c$. For small p it can be approximated by $P(p,c) \approx 1 - e^{-pc}$. This latter equation shows that the delay risk associated with the critical path should decrease with increasing pc .

Empirical data of construction projects. The dataset is composed of 77 construction projects, with multiple project schedules for each construction project, totalling 323 project schedules. The project schedules were generated by the project managers using an industry standard enterprise software package (Oracle Primavera P6).

Data availability and code availability

All the data necessary to support our conclusions is reported in the figures. Code for the duplication-split model is provided in the methods. Raw data for construction projects has restricted access and can be provided upon consultation. Request for data should be directed to corresponding authors.

Received: 31 August 2022; Accepted: 27 December 2022

Published online: 10 January 2023

References

- Jensen, A., Thuesen, C. & Geraldi, J. The projectification of everything: Projects as a human condition. *Proj. Manag. J.* **47**, 21–34 (2016).
- Scranton, P. Projects as a focus for historical analysis: Surveying the landscape. *Hist. Technol.* **30**, 354–373 (2014).
- Flyvbjerg, B., Skamris Holm, M. K. & Buhl, S. L. How common and how large are cost overruns in transport infrastructure projects? *Transp. Rev.* **23**, 71–88 (2003).
- Park, J. E. Schedule delays of major projects: What should we do about it? *Transp. Rev.* **41**, 814–832 (2011).
- Kelley, J. E. & Walker, M. R. Critical-path planning and scheduling. in *Papers presented at the December 1–3, 1959, eastern joint IRE-AIEE-ACM computer conference* 160–173 (Association for Computing Machinery, 1959).
- Santolini, M., Ellinas, C. & Nicolaidis, C. Uncovering the fragility of large-scale engineering projects. *EPJ Data Sci.* **10**, 1–13 (2021).
- Ellinas, C. The domino effect: An empirical exposition of systemic risk across project networks. *Prod. Oper. Manag.* **28**, 63–81 (2019).
- Ellinas, C., Allan, N. & Johansson, A. Toward project complexity evaluation: A structural perspective. *IEEE Syst. J.* **12**(228–239), 1 (2018).
- Vespignani, A. Modelling dynamical processes in complex socio-technical systems. *Nat. Phys.* **8**, 32–39 (2012).
- Kleinberg, J. M., Kumar, R., Raghavan, P., Rajagopalan, S. & Tomkins, A. S. The web as a graph: Measurements, models, and methods. In *Computing and Combinatorics* (eds Asano, T. et al.) 1–17 (Springer, 1999).
- Vázquez, A., Flammini, A., Maritan, A. & Vespignani, A. Modeling of protein interaction networks. *CPU* **1**, 38–44 (2003).
- Pastor-Satorras, R., Smith, E. & Solé, R. V. Evolving protein interaction networks through gene duplication. *J. Theor. Biol.* **222**, 199–210 (2003).
- Chung, F., Lu, L., Dewey, T. G. & Galas, D. J. Duplication models for biological networks. *J. Comput. Biol.* **10**, 677–687 (2003).
- Watts, D. J. & Strogatz, S. H. Collective dynamics of ‘small-world’ networks. *Nature* **393**, 440–442 (1998).
- Amaral, L. A. N., Scala, A., Barthélémy, M. & Stanley, H. E. Classes of small-world networks. *Proc. Natl. Acad. Sci.* **97**, 11149–11152 (2000).
- Albert, R., Jeong, H. & Barabási, A.-L. Error and attack tolerance of complex networks. *Nature* **406**, 378–382 (2000).
- Buldyrev, S. V., Parshani, R., Paul, G., Stanley, H. E. & Havlin, S. Catastrophic cascade of failures in interdependent networks. *Nature* **464**, 1025–1028 (2010).
- Schwartz, N., Cohen, R., ben-Avraham, D., Barabási, A.-L. & Havlin, S. Percolation in directed scale-free networks. *Phys. Rev. E* **66**, 015104(R) (2002).

Acknowledgements

This work was partly supported by the European Union’s Horizon 2020 and the Cyprus Research Innovation Foundation under the SEED program (grant agreement 0719(B)/0124). Nodes & Links Ltd provided support in the form of salary for Alexei Vazquez, Iacopo Pozzana, Georgios Kalogridis and Christos Ellinas, but did not have any additional role in the conceptualisation of the study, analysis, decision to publish, or preparation of the manuscript.

Author contributions

A.V. conceived the duplication-split model, estimated the model properties, performed the generative model simulations and analysed the data. I.P. and G.K. created and curated the construction project dataset. C.E. directed the work and provided expertise in project scheduling. A.V. and C.E. wrote the paper.

Competing interests

The authors declare no competing interests.

Additional information

Correspondence and requests for materials should be addressed to A.V. or C.E.

Reprints and permissions information is available at www.nature.com/reprints.

Publisher’s note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Open Access This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>.

© The Author(s) 2023