# SUPPORTING MATERIAL

## Team assembly mechanisms determine collaboration network structure and team performance

Roger Guimerà†, Brian Uzzi†, Jarrett Spiro, and Luís A. Nunes Amaral*

†These two authors contributed equally to this work.
*To whom correspondence should be addressed; E-mail:amaral@northwestern.edu

# 1 Collaboration network construction

For each of the datasets considered, we assemble a list of collaborations—papers or musicals—sorted by chronological order. Each entry in the list contains the names of the agents involved in the collaboration. The following is an extract of the list for the journal Ecology, corresponding to a subset of the publications in 1965:

```
PLUMMER_GL CROSSLEY_DA GARDINER_DA
TISDALE_EW HIRONAKA_M FOSBERG_MA
RICKARD_WH DAVIS_JJ HANSON_WC WATSON_DG
WATT_RF HEINSELMAN_ML
ZAHNER_R DEBYLE_NV
STEPHENSON_SN BUELL_MF
CUSHING_EJ WRIGHT_HE
FRITTS_HC SMITH_DG CARDIS_JW BUDELSKY_CA
WHITTAKER_RH NIERING_WA
BRANT_DH KAVANAU_JL
BICK_GH BICK_JC
EALEY_EHM BENTLEY_PJ MAIN_AR
WURSTER_DH WURSTER_CF STRICKLAND_WN
```

From this list, we build the network of collaborations as follows:

1. Add the agents involved in the first collaboration to the network. Classify the agents as newcomers, and connect each of them to all other agents in the collaboration. Classify

all links as newcomer-newcomer links.

2. Select the next collaboration.

3. For each agent in the collaboration, determine if she is an incumbent or a newcomers. If an agent is already in the network—that is, if the agent has participated in a previous collaboration in the list—she is or becomes an incumbent. Otherwise, she is a newcomer.

4. Add the newcomers to the network. Connect each agent in the collaboration to all other agents in the collaboration. Keep track of the different types of links established.

5. Calculate for how long each agent in the network has been inactive—that is, without participating in a collaboration. If the inactivity period of an agent is longer than $\tau$, remove the agent from the network (see the next section for details on how we determine $\tau$).

6. Repeat steps 2-5 until all collaborations have been considered.

# 2 Agent decay

In our analysis, we remove from the network agent that remain inactive for longer than $\tau$ time steps. This choice is justified by the observation that agents will not remain in the network forever. Furthermore, it enables the network to reach a state in which its size and connectivity are approximately constant.

Due to the particularities of the different creative enterprises, we obtain individual estimates of $\tau$ for each of the networks considered. To obtain $\tau$, we calculate the number $t$ of productions between each pair of consecutive productions in which an agent takes part. The cumulative distribution $P(> t)$ gives the probability that an agent reappears $t$ productions after she appeared for the last time. We find that for all enterprises $P(>t)$ decays with an exponential tail

$$P(>t) \sim \exp\left(-t/t^*\right), \qquad (1)$$

where $t^*$ is an enterprise-specific characteristic time (Fig. 1A). We choose $\tau$ so that

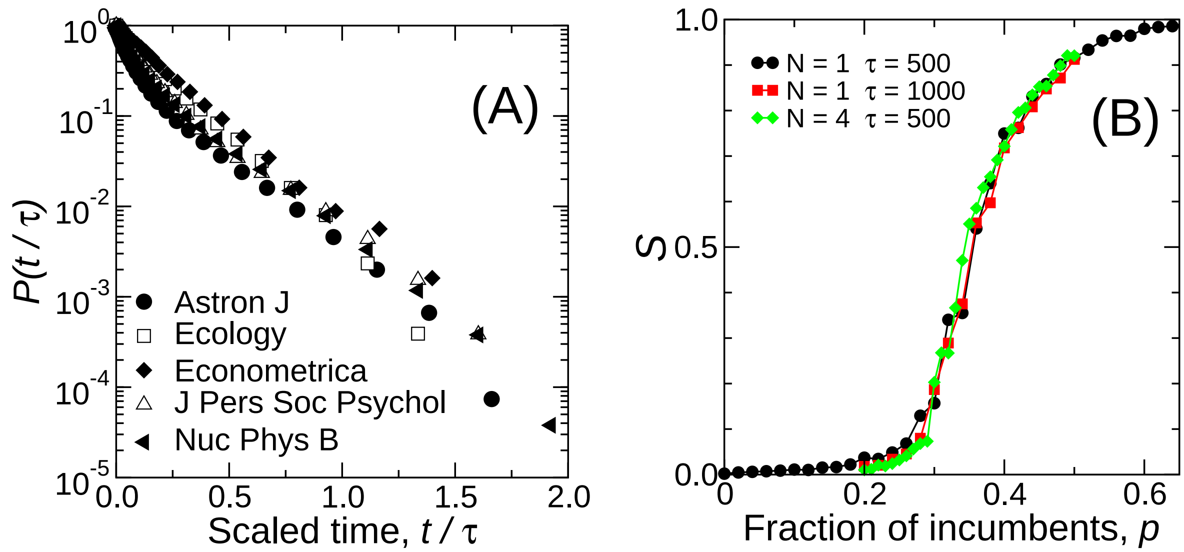


Figure 1: Node decay and time scales in the analysis of creative networks. **(A)** Probability distribution function $P(>t/\tau)$ for the enterprises considered in the manuscript. The time is rescaled with a different $\tau$ for each enterprise: $\tau_{\rm AJ}=5000$, $\tau_{\rm Ecol}=3000$, $\tau_{\rm Econ}=800$, $\tau_{\rm JPSP}=3000$, $\tau_{\rm NPB}=3000$, and $\tau_{\rm Bway}=85$. **(B)** Relative size of the largest cluster of the network as a function of $p$ for different values of $\tau$ and $N$.

$P(>\tau)=0.01$, implying that we remove agents from the network when the probability of they appearing in a new production becomes smaller than 1%.

It is important to note that changing $\tau$ does not alter the predictions made by the model for the main quantities discussed in the manuscript (Fig. 1B). Specifically, the emergence of the giant component and its relative size do not depend on $\tau$, even though the size $S$ of the network and the number $L$ of links depend on $\tau$. Modifying the time scales by other means does not change the behavior of the giant component, either. For example, instead of creating one production each time step, one can create $N$ productions. As we show in Fig. 1B, the value of $N$ does not affect the emergence of a giant component.

# 3 Calculation of the relative size of the largest component

To calculate the relative size $S$ of the largest component, we start by identifying all the components of the network. A component is a set of agents that are directly or indirectly connected to each other by a series of links—i.e. that are mutually reachable—but that are otherwise disconnected from the rest of the network. We then calculate the number of agents in the largest component and divide it by the total number of agents in the network.

# 4 Network dynamics

As discussed in the manuscript, not all networks with the same relative size $S$ of the largest cluster have the same properties. In particular, networks with different fraction $f_R$ of repeat incumbent-incumbent links have different dynamics. To measure how fast the network evolves, we define the *turnover time* $T$ as the time required to "replace" all the links in the network

$$T = \frac{2L}{m(m-1)(1-f_R)}\,, \tag{2}$$

where $L$ is the number of links in the steady state.

For $p = 0$—that is, when all nodes are newcomers—there are no repeat links and $T = \tau$.

When $p = 1$—that is, when all nodes are incumbents—all links are repeat incumbent-incumbent links and $T \to \infty$. For intermediate values of $p$, the fraction of repeat links increases with $q$, and so does $T$ (Fig. 2).

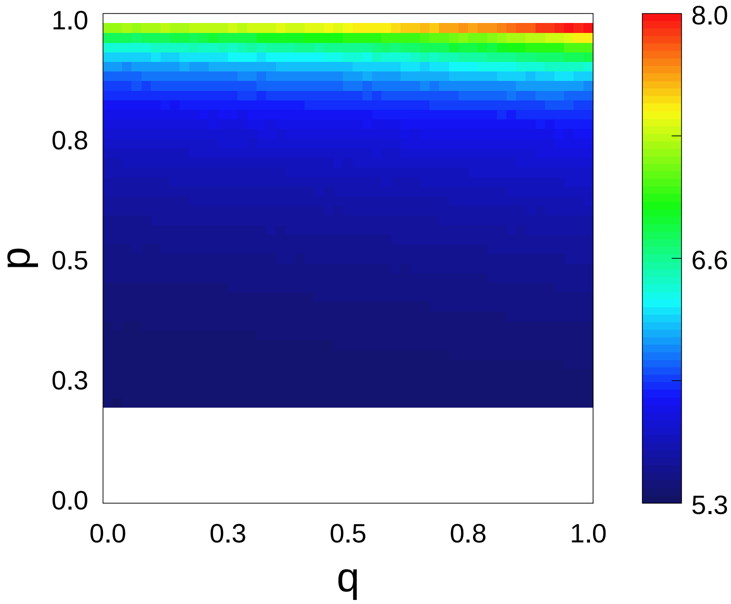


Figure 2: Network dynamics. The figure displays the logarithm of the turnover time $\log T$ as a function of $p$ and $q$. The turnover time increases with with $p$ and, for a fixed $p$, it also increases with $q$.