Reconstructing Large Scale Production Networks

Ashwin Bhattathiripad* Vipin P. Veetil†

December 2025

Abstract

This paper develops an algorithm to reconstruct large weighted firm-to-firm networks using information about the size of the firms and sectoral input-output flows. Our algorithm is based on a four-step procedure. We first generate a matrix of probabilities of connections between all firms in the economy using an augmented gravity model embedded in a logistic function that takes firm size as mass. The model is parameterized to allow for the probability of a link between two firms to depend not only on their sizes but also on flows across the sectors to which they belong. We then use a Bernoulli draw to construct a directed but unweighted random graph from the probability distribution generated by the logistic-gravity function. We make the graph aperiodic by adding self-loops and irreducible by adding links between Strongly Connected Components while limiting distortions to sectoral flows. After all this, we convert the unweighted network to a weighted network by solving a convex quadratic programming problem that minimizes the Euclidean norm of the weights. This approach is equivalent to minimum-energy assignment under constraints. The solution preserves the observed firm sizes and sectoral flows within reasonable bounds, while limiting the strength of the self-loops. All the optimization problems used in our algorithm are numerically well-behaved. Computationally, the algorithm is $O(N^2)$ in the worst case, but it can be evaluated in $O(N)$ via sector-wise binning of firm sizes, albeit with an approximation error. We implement the algorithm to reconstruct the full US production network with more than 5 million firms and 100 million buyer-seller connections. The reconstructed network exhibits topological properties consistent with small samples of US buyer-seller networks, including fat-tails in degree distribution, mild clustering, and near-zero reciprocity. We provide the algorithm as an open-source library in Python and C to enable researchers to reconstruct large-scale production networks from publicly available data on firm-size and sectoral-flows.

Key Words: Production Networks; Gravity Model of Trade; Markov Chains; Large-Scale Sparse Graphs

*Economics Area, Indian Institute of Management Kozhikode, Kerala 673 570, India.

1 Introduction

A production network is a graph in which the vertices are firms and the edges are the buyer-seller relations. Naturally, these are directed graphs with the direction of the edges indicating the flows of money from one firm to another, and equivalently, the flow of intermediate inputs in the opposite direction. Production networks can also be thought of as weighted graphs where the weights mark the share of one firm's cost of production incurred on the input from another firm[1]. Theoretical and empirical investigations of the last decade tell us that the impact of natural disasters, climate change, and pandemics on the economy as a whole depends not only on the significance of the regions which they decimate or the sectors which they harm, but also on the topology of the production network through which these impulses propagate to the rest of the economy (Carvalho et al., 2021; Palepu et al., 2025). In many of these areas, theoretical models must be calibrated to real-world data to yield empirical insights. We cannot, for instance, know the impact of an increase in the probability of extreme rainfall in the Himalayas on Brittany if we do not know how firms based in certain susceptible parts of the Himalayas are directly and indirectly connected to those in Brittany. Unfortunately, however, real world firm-to-firm network data is not available for most countries[2]. Data on the flow of goods at a more aggregate level, though, is easily accessible: most countries periodically publish national input-output tables, the University of Groningen publishes the World Input-Output Table (WIOT), and the OECD publishes its inter-country input-output tables[3]. Many countries also publish data on several firm-level attributes, including the distribution of firm sizes[4]. In this paper, we develop an algorithm to generate firm-to-firm production networks using publicly available data on sectoral flows and the distribution of firm sizes. Our algorithm generates production networks that preserve sectoral flows of the economy upon the aggregation of firm-to-firm flows while faithfully reproducing certain widely reported properties of granular production networks.

[1]With Cobb-Douglas production functions, these weights would naturally be the Cobb-Douglas exponents since the share of expenditure directed at different goods does not depend on prices.

[2]Near-universe firm-to-firm production-network data exist for Belgium (Dhyne et al., 2021), Hungary (Diem et al., 2022), Ecuador (Bacilieri et al., 2023), Chile (Huneeus, Kroft and Lim, 2021), Japan (Mizuno, Souma and Watanabe, 2014), and the Dominican Republic (Cardoza et al., 2025). Large but sub-national coverage exists for India (Panigrahi, 2021). Note that though these data sets exist, they are not publicly available. Which means that most researchers will not be able to leverage these to estimate the empirical consequences of their theoretical insights.

[3]There is substantial cross-country heterogeneity in the frequency, sectoral detail, and valuation of official input–output releases. Many advanced economies publish annual Supply–Use Tables and less frequent benchmark IO tables (often every five years) with varying sectoral resolution and import/price conventions, while a number of low- and middle-income countries release IO tables more sporadically and at coarser levels of aggregation.

[4]Statistical agencies of the United States (e.g. SBA and Census Bureau), the European Union (Eurostat's SBS for member states), Japan, and Australia regularly release industry-by-size tables. Comparable series are also available for several large developing economies (including India), but typically at lower frequency and with less standardized binning.

Our algorithm follows a four-step procedure. Consider an economy with n firms. In the first step, we embed an augmented gravity model in a logistic function to generate an $N_F \times N_F$ matrix $\mathbf{P}$ of the probabilities of directed connections between firms in the economy. The probability of the connection between any two firms increases with the product of the size of the firms and the flows between the sectors to which the two firms belong. The augmented gravity model is characterized by four parameters. One parameter captures the sensitivity of the probability to firms' sizes. Another parameter uniformly scales the probabilities of connections between all firms irrespective of their size and sector: this allows us to create a production network that matches the empirically observed density of connections between firms. The last two parameters pertain to the sensitivity of the probability of connection to sectoral flow: one of which is non-linear but uniform across all sectors, the other linear but specific to each sector pair. The four parameters together determine the intensity of the augmented gravity function, which, when inserted into a logistic function generates the probability matrix $\mathbf{P}$. The four parameters are estimated using a non-linear optimization problem that minimizes the difference between the expected number of linkages in the network given the probability matrix $\mathbf{P}$ and the actual number of connections in the production network. This objective function is subject to the constraint that the expected sectoral sizes implied by the flows ingrained in $\mathbf{P}$ are within an error bound of the actual sizes of sectors in the economy.

In the second step, we draw an ensemble of directed but unweighted adjacency matrices $\{\mathbf{A}_1, \mathbf{A}_2, ..., \mathbf{A}_N\}$ via Bernoulli draws on the probability matrix $\mathbf{P}$ estimated in step one. The third step involves generating an irreducible closure of the ensemble of graphs. We decompose each graph into its Strongly Connected Components (SCC), then connect the sources and sinks of the SCC. To determine which firms within the SCC to connect as sources and sinks, we formulate an optimization problem that minimizes the disturbances to sectoral sizes resulting from the new connections. We also make the network aperiodic by giving each firm a link to itself. In the fourth and final step, we convert the unweighted network into a weighted network by computing weights that minimize the energy in the network. More specifically, we minimize the sum of the squares of the weights, subject to the constraints that the invariant firm size vector corresponding to the matrix is within a reasonable bound of the empirically observed firm sizes. The minimum energy approach ensures that weight assignment is not such that some links have very high weights, while others have near-zero weights rendering them irrelevant. Put differently, it ensures that the weighted network is a 'true' corollary of the unweighted network.

It is worth noting that our emphasis on Markov closure and preserving the firm size distribution has to do with the use of the reconstructed network for policy applications. More specifically,

a wide range of applications of production networks involves studying the propagation of granular shocks, including those that emanate from climate change and natural disasters. In such circumstances, we may wish to investigate the deviation of the system from equilibrium and the time it takes to return to it. But for such an analysis, an equilibrium—preferably a unique and stable one—must be guaranteed to exist. With the irreducible and aperiodic closure of the graphs, step three of our algorithm guarantees the existence of such an equilibrium. Scientists will therefore be able to use the production networks generated by the algorithm for traditional 'impulse-response' analysis in a granular setting. In a similar vein, certain questions like the propagation of productivity shocks are sensitive to the size distribution of firms (Gabaix, 2011). In such cases, impulse-response experiments can only be run on networks that faithfully ingrain the variation in firm sizes. This is why our algorithm bounds the distance between the stationary vector of firm sizes corresponding to the reconstructed production networks and its empirical counterpart[5].

1.1 Organization of the Paper

The rest of the paper is organized as follows. Section 2 compares our approach to reconstructing production networks with that of those before us. Section 3 presents our four-step network-reconstruction algorithm. Section 3.1 develops the logistic-gravity specification that yields the matrix of firm-to-firm link probabilities and details the optimization used to estimate its four parameters. Section 3.2 describes how we create a directed, unweighted binary backbone of the production network using the probability matrix generated using the parameters of the gravity model. Section 3.3 formulates the optimization that renders the reconstructed network irreducible and aperiodic. Section 3.4 introduces the minimum-energy procedure for assigning weights and proves that the stationary distribution of firm sizes induced by the reconstructed network lies within an ε-neighborhood of the empirical size vector. Section 4 analyzes the computational complexity of the algorithm and proposes a practical sector-wise binning approximation that reduces the complexity from $O(N^2)$ to $O(N)$. It also specifies the error that arises from such an approximation. Section 5 applies the algorithm to the full US economy and compares the reconstructed network's topological properties with those observed in a large sample. Section 6 extends our model to a setting in which firms have multiple production units (factories) located in a geographical space. Section 7 concludes the paper. Appendix A collects supplemental results:

[5]Ideally, the stationary size of each firm in the reconstructed network must be identical to its empirical size. Unfortunately, however, the mathematical structure of the problem is such that one cannot guarantee the existence of network weights once we impose one equality constraint for each firm's size. We, therefore, replace these equality constraints with their inequality counterparts.

one of which provides finite-sample bounds on some global properties of the Bernoulli ensemble, the other proves that our minimum-energy weight estimator is a special case of a maximum-entropy formulation, and the third shows that the adjacency matrix produced by our algorithm is sufficiently 'close' to one for which the empirical firm-size vector is exactly stationary. The network reconstruction code is available at https://github.com/AshwinBTT/NetworkReconstruction[6].

2 Related Literature

There are in some senses two different approaches to reconstructing production networks. The first approach requires an observed unweighted adjacency matrix. The observed buyer-seller relations are treated as a partial view of the true production network. One then extracts information about the 'true' adjacency structure embedded in this partial view to recreate the whole production network. Naturally, there are various ways to extract information about the true network from the observed adjacency matrix. One way is to estimate the correlates that determine the probability of a connection between two firms, using say, a LASSO regression on the observed network. Once the regression coefficients—say pertaining to firm size, location, and sector—are estimated, the probabilistic model is used to generate new links that are added to the observed links to complete the production network. Another way of using information about observed buyer-seller links is to combine it with information about aggregate flows: one postulates that the aggregate flows that emerge from summing firm-to-firm flows must be exactly those reported in the input-output table. The 'missing links', therefore, are the connections that must be added to the observed firm buyer-seller relations so that the two together aggregate to the observed sectoral flows (Hooijmaaijers and Buiten, 2019; Buiten et al., 2021; Welburn et al., 2023). Several different techniques have been used to compute such 'missing links', including deterministic algorithms that generate a point-estimate and maximum-entropy methods that capture the uncertainty involved in reconstructing production networks (Squartini et al., 2017). Typically, once the network is 'completed' with the addition of new buyer-seller connections, weights are assigned by solving an optimization problem that minimizes the deviations from observed firm/sectoral flows or sizes (Bacilieri and Austudillo-Estevez, 2023).

The second approach to reconstructing production networks dispenses with the need for any information on actual buyer-seller connections (Ialongo et al., 2022). Instead, it uses information about firm attributes (e.g., sizes) and information on aggregate flows like those present in IO tables. Some of these methods rely on information about the number of buyers and sellers

[6]The repository contains all pertaining to our four-step network reconstruction algorithm. We are currently integrating these files into a single Python/C library. We expect to complete this by the end of December 2025.

(Mastrandrea et al., 2014), others altogether do away with the need for any firm-level network information (Vece, Garlaschelli and Squartini, 2023). Much as in the first approach, maximum-entropy methods are used here too to create connections and assign weights. Overall, the last decade has witnessed the emergence of a vibrant literature on the reconstruction of production networks, with a common reliance on maximum-entropy formulations and creative ways to harness widely available information on aggregate flows (see Cimini, Mastrandrea and Squartini (2021) for concise overview of this literature).

Naturally, our approach to network reconstruction falls within the second group. We reconstruct the network without using any partially observed buyer-seller links. Instead, we estimate the binary backbone of the network with an augmented gravity model that makes use of sectoral flows and firm sizes. Our approach differs from that of others before us in five ways. The first of which is that we generalize the 'stripes corrected' gravity model, which forbids connections between firms that belong to sectors that have no flows between them (Ialongo et al., 2022). Within our model, the function that generates the probabilities of connections between firms takes as an argument pairwise sector flows. Furthermore, parameters of the function are estimated under constraints that include sectoral sizes. All of which means that the Bernoulli ensemble generated by this process not only does not allow for connections between firms that belong to unconnected sectors but is also consistent with sectoral flows in expectation. Second, we are the first to enforce Markov regularity conditions while reconstructing production networks. After sampling the directed, unweighted graph, we make the adjacency irreducible and aperiodic via a succinct, minimally distortive augmentation through a well-defined optimization problem. Our Markov regularity procedure ensures that the reconstructed network has a unique and stable equilibrium. Third, unlike several other network reconstruction approaches, we generate graphs that remain faithful to the size distribution of firms in the economy. In fact, our algorithm generates graphs that match not only the cross-section of the size distribution of firms but also the steady-state distribution within an error bound. Fourth, our algorithm for reconstructing production networks is perhaps more scalable than those before ours. More specifically, the computational complexity of our algorithm is $O(N^2)$ because of pairwise probability evaluation, but a sector-wise size-binning of firms reduces the complexity to near-$O(N)$. That said, we have been able to reconstruct a nationwide production network at an unprecedented scale even without the approximation. Table 1 summarizes the sizes of the networks reconstructed hitherto. Some papers are theoretical and, therefore, little is known about the practicalities of their algorithms in generating large-scale networks. Several papers generate relatively small networks with approximately 10^4 firms. The largest network reconstructed before our own is that of Ialongo et al. (2024), which comprises

approximately 10^5 firms. The network reconstructed in this paper has 5×10^6 firms. Fifth, prior studies on network reconstruction rarely share usable code. In contrast, we release open-source code of our algorithm that can be used to reconstruct networks by others using publicly available data on sectoral flows and firm sizes.

Table 1: Papers that reconstruct production networks

Paper	**N**	E_{obs}	ΔE	E_{tot}	**Region / Notes**
Welburn et al. (2023)	24,879	137,811	218,259	356,070	Global (FactSet); ensemble thresholding
Reisch et al. (2022)	89,000+	-	235,000	235,000	Hungary; admin-based reconstruction (no E_{obs} split)
Mungo et al. (2023)- Compustat	915	NR	NR	NR	U.S. (public firms); link-prediction study
Mungo et al. (2023)-FactSet	6,714	NR	NR	NR	Global (listed); link-prediction study
Mungo et al. (2023)-Ecuador	10,000	NR	NR	NR	Ecuador (VAT); link-prediction study
Ialongo et al. (2022)	NR	NR	NR	NR	Netherlands (two bank datasets); generalized MaxEnt
Buiten et al. (2021)	NR	NR	NR	NR	Netherlands (CBS method); deterministic pipeline
Mastrandrea et al. (2014)	-	-	-	NR	ECM (MaxEnt with degrees+strengths); no fixed support
Di Vece et al. (2023)	-	-	-	NR	Integrated/conditional MaxEnt (ITN); no fixed support
Ialongo et al. (2024)	10^5	NR	10^6	10^6	Netherlands (multi-scale embeddings); method paper
Bhattathiripad and Veetil (2024)	10^6	0	10^8	10^8	United States

Notes: E_{obs} = observed links in the starting (partial) network; ΔE = links added; $E_{\text{tot}} = E_{\text{obs}} + \Delta E$. '-' = not applicable (from-scratch methods don't start from a fixed E_{obs}); NR = not reported publicly in the cited sources.

3 The Model

Notation Matrices are denoted by bold capital letters and vectors by bold lowercase letters: $\mathbf{X}$ is a matrix, $\mathbf{x}$ a vector, and x a scalar.

3.1 Estimating firm-to-firm link probabilities via sector-aware logistic gravity model

Sectors are denoted by $l, m \in \{1, ..., N_S\}$. Let $\mathbf{IO} \in \mathbb{R}_+^{N_S \times N_S}$ denote the sectoral input-output table oriented with *buyer sectors in rows* and *seller sectors in columns*, with the entry IO_{kl} being

the flow of money from sector k to sector l. We use two normalizations of $\mathbf{IO}$: a max-normalized matrix $\mathbf{S}$ and a row-stochastic matrix $\mathbf{I}$. The max-normalization is

$$S_{kl} \;:=\; \frac{IO_{kl}}{\max_{k',l'} IO_{k',l'}} \in [0,1]$$

This rescales all flows by the largest sector-to-sector flow, naturally, preserving zeros and relative magnitudes. The row-stochastic normalization is

$$I_{kl} \;:=\; \frac{IO_{kl}}{\sum_{k'} IO_{k',l}}, \qquad \mathbf{I1} = \mathbf{1}$$

This scales each row to sum to one so that I_{kl} is the share of buyer-sector k's expenditure allocated to seller-sector l[7].

Firms are indexed by $i, j \in \{1, \ldots, N_F\}$. Let $\pi : \{1, \ldots, N_F\} \to \{1, \ldots, N_S\}$ map firms to sectors and write $k = \pi(i)$ and $l = \pi(j)$. Firm sizes are normalized by the largest firm, so $m_i \in (0, 1]$. For $i \in k$ and $j \in l$, define the intensity x_{ij} as:

$$x_{ij} \;:=\; z\, \lambda_{kl}\, S_{kl}^{\kappa}\, (m_i m_j)^{\alpha}$$

The link probability p_{ij} is given by

$$p_{ij} \;:=\; \frac{x_{ij}}{1 + x_{ij}}$$

The probability function being a logistic map is approximately linear for small intensities and smoothly saturates. Let $\mathbf{P} = (p_{ij})_{i \neq j}$ denote the $N_F \times N_F$ matrix of probabilities of connections between firms. The intensity between firms i and j depends on global parameters z, κ, α and a sector-pair multiplier λ_{kl} where $k = \pi(i)$ and $l = \pi(j)$. The domains and the roles of the four sets of parameters are as follows:

(i) Global density $z \in [\underline{z}, \overline{z}]$, with $\underline{z} > 0$ and finite $\overline{z}$. This parameter calibrates the overall density so that the expected number of links implied by $\mathbf{P}$ matches the observed number of links or desired density of the network.

(ii) Firm-size elasticity $\alpha \in [\underline{\alpha}, \overline{\alpha}]$, with $0 < \underline{\alpha} < \overline{\alpha} < 1$. This parameter governs the sensitivity of the probability of connection to firm sizes via $(m_i m_j)^{\alpha}$. The higher the α, the greater the tilt towards large-large firm pairs.

[7]The idea of normalizing the IO table to view it as a Markov transition matrix of money flows has a long history, see for instance Augusztinovics (1965) and Leontief and Bródy (1993).

(iii) Sector-flow sensitivity $\kappa \in (0,1)$. This parameter controls how strongly sectoral flows affect probabilities. Large κ amplifies the differences in the flows across sectors.

(iv) Sector-pair scaling $\lambda_{kl} \in [\underline{\lambda}_{kl}, \overline{\lambda}_{kl}]$, with $\underline{\lambda}_{kl} > 0$ and finite $\overline{\lambda}_{kl}$. This sector-pair specific multiplier linearly scales the gravity-intensity to capture residual sectoral heterogeneity not absorbed by the global exponent κ. Put differently, it captures how intersectoral flows influence the probability of connection between firms beyond that which is contained in a uniform transformation of the flows. Note that there are as many such parameters as the number of sector-pair flows.

These parameters are estimated by solving the following non-linear programing (NLP) problem:

$$\begin{aligned}
\min_{z,\,\kappa,\,\alpha,\,\{\lambda_{kl}\}} \quad & \left(\sum_{i=1}^{N_F} \sum_{j=1}^{N_F} p_{ij} \;-\; n_d \right)^2 \\
\text{s.t.} \quad & \frac{1}{s_l} \left| \sum_{k=1}^{N_S} \sum_{\substack{i:\,\pi(i)=k \\ j:\,\pi(j)=l}} m_i \, p_{ij} \, I_{kl} \;-\; s_l \right| \;\leq\; \varepsilon \qquad \forall\, l \in \{1,\ldots,N_S\}, \\
& z \in [\underline{z}, \overline{z}], \quad \alpha \in [\underline{\alpha}, \overline{\alpha}], \quad \kappa \in (0,1), \quad \lambda_{kl} \in [\underline{\lambda}_{kl}, \overline{\lambda}_{kl}] \;\; \forall\, k,l
\end{aligned}$$

Here $s_l := \sum_{i:\,\pi(i)=l} m_i$ is the empirical size of sector l, n_d is the target (observed) number of links, and $\varepsilon > 0$ is a small error bound. The objective matches the expected number of links implied by $\mathbf{P}$ to n_d. Each sectoral constraint bounds the deviation between the expected inflow into sector l ($\sum_k \sum_{i \in k,\, j \in l} m_i \, p_{ij} \, I_{kl}$) and its empirical size s_l. Intuitively, $m_i \, p_{ij} \, I_{kl}$ is the expected expenditure from firm i to firm j: p_{ij} is the link probability, while I_{kl} proxies the conditional weight share using buyer-sector l's expenditure split across seller-sectors k. With box constraints on $(z, \alpha, \kappa, \{\lambda_{kl}\})$ the feasible set is compact and the objective is continuous, so a minimum exists by the Weierstrass theorem. The solution delivers parameters $(z, \kappa, \alpha, \{\lambda_{kl}\})$ and hence the $N_F \times N_F$ matrix $\mathbf{P} = [p_{ij}]$ of firm-to-firm connection probabilities.

3.2 Bernoulli ensemble for the binary backbone

Given the calibrated probability matrix $\mathbf{P} = [p_{ij}]_{i \neq j}$ from Section 3.1, let $A_{ij,u} \sim \text{Bernoulli}(p_{ij})$ for each ordered pair $i \neq j$, and set $A_{ii,u} = 0$. Note that the u here denotes one particular draw. Within any fixed draw $\mathbf{A}_u$ the edges $\{A_{ij,u}\}_{i \neq j}$ are independent but generally non-identical. For a fixed ordered pair (i,j), the sequence $\{A_{ij,u}\}_{u=1}^{n}$ is iid across draws with common success

probability p_{ij}. Collect the n draws as $\mathcal{G}(\mathbf{P}) := \{\mathbf{A}_1, \ldots, \mathbf{A}_n\}$[8].

For each $u \in \{1, \ldots, n\}$, the adjacency matrix $\mathbf{A}_u := (A_{ij,u})$ is a directed graph with independent Bernoulli edges conditional on $\mathbf{P}$. Let $\mathcal{G}(\mathbf{P}) := \{\mathbf{A}_1, \ldots, \mathbf{A}_n\}$ denote the collection of the Bernoulli ensemble generated by $\mathbf{P}$. Let the total edges of the network $\mathbf{A}_u$ be denoted by $\mathcal{E}_u := \sum_{i \neq j} A_{ij,u}$.

Remark 1 (CLT for the number of edges (density of the network)). Fix the probability matrix $\mathbf{P}$ and consider a single draw $\mathbf{A}_u$ from the Bernoulli network model. Conditional on $\mathbf{P}$, the edge indicators $\{A_{ij,u}\}_{i \neq j}$ are independent (though not identically distributed) Bernoulli(p_{ij}) variables.[9] Define the centered variables

$$X_{ij} \;:=\; A_{ij,u} - p_{ij}$$

which satisfy $\mathbb{E}[X_{ij}] = 0$ and are uniformly bounded in absolute value, $|X_{ij}| \leq 1$. Let

$$\mu_{\mathcal{E}} \;:=\; \sum_{i \neq j} p_{ij} \qquad \sigma_{\mathcal{E}}^2 \;:=\; \sum_{i \neq j} p_{ij}(1 - p_{ij})$$

so that $\mathcal{E}_u = \sum_{i \neq j} A_{ij,u}$ has mean $\mu_{\mathcal{E}}$ and variance $\sigma_{\mathcal{E}}^2$ conditional on $\mathbf{P}$. Assume $\sigma_{\mathcal{E}}^2 \to \infty$ as $N_F \to \infty$[10] Because the X_{ij} are independent, mean-zero, uniformly bounded, and their total variance $\sigma_{\mathcal{E}}^2$ diverges, the Lindeberg condition for sums of independent, non-identically distributed variables is satisfied. The Lindeberg–Feller Central Limit Theorem therefore yields:

$$\frac{\mathcal{E}_u - \mu_{\mathcal{E}}}{\sigma_{\mathcal{E}}} \;\Rightarrow\; \mathcal{N}(0, 1)$$

△

Remark 2 (CLT for empirical degree distributions over finitely many bins). Define the out-degree

[8] After sampling $\mathbf{A}_u$ we drop isolated firms, i.e. firms that have no incoming and no outgoing links. This avoids forcing arbitrary links in later steps (e.g., when enforcing Markov regularity). In practice, we see that such dropping has a negligible impact on the reconstructed network.

[9] In general the edge probabilities p_{ij} vary with both i and j, so even along a fixed row or column of the adjacency matrix the entries are not identically distributed. Our argument therefore relies only on independence and variance growth, not on any i.i.d. structure.

[10] The idea here is that the total variance of the edge sum diverges as the network grows, so fluctuations cannot be concentrated on finitely many edges. This rules out ultra-sparse regimes where the expected number of edges remains $O(1)$, but is fully compatible with highly heterogeneous, even powerlaw, degree sequences encoded in $\mathbf{P}$, as long as the expected average degree does not vanish. This is because the fluctuations we control are the coordinatewise differences $A_{ij,u} - p_{ij}$ between the realized Bernoulli draw and its mean. Fatness of tails will appear in both the expected degrees (implied by $\mathbf{P}$) and the realized degrees (in $\mathbf{A}_u$), but their edgewise differences are always bounded in $[-1, 1]$. Thus no single edge can move the total by more than a fixed amount, i.e. aggregate fluctuations come from many small deviations, which is exactly the regime where the Lindeberg-Feller CLT applies.

of firm i in the Bernoulli draw u from the probability matrix $\mathbf{P}$ as $d^{\mathrm{O}}_{i,u} := \sum_{j\neq i} A_{ij,u}$ and the empirical out-degree probability mass function of $\mathbf{A}_u$ by

$$F^{\mathrm{O}}_u(h) := \frac{1}{N_F}\sum_{i=1}^{N_F} \mathbf{1}_{\{d^{\mathrm{O}}_{i,u}=h\}}, \qquad h \in \mathbb{N},$$

so $F^{\mathrm{O}}_u(h)$ is the realized fraction of firms with out-degree h in draw u. To define $\mathbb{E}_{\mathbf{P}}\big[F^{\mathrm{O}}_u(h)\big]$, i.e. the expectation of $F^{\mathrm{O}}_u(h)$ conditional on the probability matrix $\mathbf{P}$, first define the degree-h probability for firm i as

$$\vartheta^{\mathrm{O}}_i(h) := \mathbb{P}_{\mathbf{P}}\big(d^{\mathrm{O}}_{i,u} = h\big)$$

so that

$$\mathbb{E}_{\mathbf{P}}\big[F^{\mathrm{O}}_u(h)\big] = \frac{1}{N_F}\sum_{i=1}^{N_F} \vartheta^{\mathrm{O}}_i(h)$$

Now fix a finite set of degrees $\mathcal{N} \subset \mathbb{N}$ and assume there exist $h^\star \in \mathcal{N}$ and $\delta \in (0, 1/2]$ such that, for all sufficiently large N_F:

$$\mathbb{E}_{\mathbf{P}}\big[F^{\mathrm{O}}_u(h^\star)\big] \in [\delta,\, 1-\delta]$$

In words, as the network grows it never happens that the bins in $\mathcal{N}$ all become asymptotically empty, nor that almost all probability mass in $\mathcal{N}$ collapses into a single degree: at least one degree $h^\star$ in $\mathcal{N}$ continues to carry a nontrivial fraction of firms in expectation.[11]

Note that, for each fixed h, the terms $\mathbf{1}_{\{d^{\mathrm{O}}_{i,u}=h\}}$ in the average defining $F^{\mathrm{O}}_u(h)$ are independent across firms: conditional on $\mathbf{P}$, each firm's degree depends only on the edges in its own row of $\mathbf{A}_u$, which are independent of the edges determining other firms' degrees.[12] Therefore, $F^{\mathrm{O}}_u(h)$ is the average of N_F independent, bounded random variables $\mathbf{1}_{\{d^{\mathrm{O}}_{i,u}=h\}}$, where the i^{th} summand has mean $\vartheta^{\mathrm{O}}_i(h)$ and the total variance is of order N_F. Furthermore, the centered summands

[11] This non-degeneracy condition rules out ultra-sparse regimes where almost all firms end up with very low degrees (so every fixed bin in $\mathcal{N}$ eventually empties), as well as dense regimes where probability mass drifts to degrees outside $\mathcal{N}$. Under calibrations with uniformly bounded expected degrees, at least one such $h^\star$ must exist by a simple pigeonhole argument. For this $h^\star$, the Bernoulli variables $\mathbf{1}_{\{d^{\mathrm{O}}_{i,u}=h^\star\}}$ have variances $\vartheta^{\mathrm{O}}_i(h^\star)\big(1-\vartheta^{\mathrm{O}}_i(h^\star)\big)$ that remain bounded away from zero on average, so their total variance grows on the order of N_F. Because each $\mathbf{1}_{\{d^{\mathrm{O}}_{i,u}=h\}}$ is bounded in $[0,1]$, this variance growth ensures the Lindeberg condition for sums of independent, non-identically distributed variables.

[12] The probability matrix $\mathbf{P}$ itself is obtained from an optimization problem that may impose a constraint on the mean degree, so the degree probabilities $\{\vartheta^{\mathrm{O}}_i(h)\}_{i\leq N_F}$ are linked at the level of first moments. In our Bernoulli network model, however, once $\mathbf{P}$ is fixed we draw edges independently across (i,j), so degrees are exactly independent across firms for any N_F. In alternative sampling schemes where one insists that the realized total number of edges or the realized mean degree match a target exactly (rather than in expectation), degrees can become weakly dependent, especially in very small networks. For large networks such dependence is typically negligible.

$\mathbf{1}_{\{d^{\mathrm{O}}_{i,u}=h\}} - \vartheta^{\mathrm{O}}_i(h)$ are uniformly bounded in absolute value by 1. This boundedness, together with variance growth, allows us to apply the Lindeberg–Feller Central Limit Theorem for sums of independent (not necessarily identically distributed) variables to obtain the multivariate convergence

$$\sqrt{N_F}\left(F^{\mathrm{O}}_u(h) - \mathbb{E}_{\mathbf{P}}\big[F^{\mathrm{O}}_u(h)\big]\right)_{h\in\mathcal{N}} \;\Rightarrow\; \mathcal{N}\big(0, \Sigma_{\mathrm{O}}(\mathcal{N})\big)$$

where the covariance matrix $\Sigma_{\mathrm{O}}(\mathcal{N})$ has entries

$$\big[\Sigma_{\mathrm{O}}(\mathcal{N})\big]_{h\ell} = \lim_{N_F\to\infty} \frac{1}{N_F}\sum_{i=1}^{N_F}\Big(\mathbf{1}_{\{h=\ell\}}\,\vartheta^{\mathrm{O}}_i(h)\big(1-\vartheta^{\mathrm{O}}_i(h)\big) - \vartheta^{\mathrm{O}}_i(h)\,\vartheta^{\mathrm{O}}_i(\ell)\Big)$$

where the diagonal entry $\big[\Sigma_{\mathrm{O}}(\mathcal{N})\big]_{hh}$ is the asymptotic variance of $\sqrt{N_F}\big(F^{\mathrm{O}}_u(h) - \mathbb{E}_{\mathbf{P}}[F^{\mathrm{O}}_u(h)]\big)$, while for $h \neq \ell$ the off-diagonal entry $\big[\Sigma_{\mathrm{O}}(\mathcal{N})\big]_{h\ell}$ is negative and captures the fact that if more firms fall into bin h, fewer can fall into bin ℓ[13]. The same conclusions hold for the empirical in-degree distribution. △

Remark 3 (CLT for edge-averaged statistics (e.g., assortativity and reciprocity)). An edge-averaged statistic

$$T(\mathbf{A}_u) \;:=\; \frac{1}{\mathcal{E}_u}\sum_{i\neq j} A_{ij,u}\,\phi_{ij}(\mathbf{A}_u)$$

depends, for a given Bernoulli draw $\mathbf{A}_u$ from $\mathbf{P}$, on three random ingredients: (i) which edges $A_{ij,u}$ are present, (ii) how many edges there are in total, $\mathcal{E}_u$, and (iii) the scores $\phi_{ij}(\mathbf{A}_u)$ attached to each realized edge. A natural way to obtain a CLT is to ensure that (a) the denominator $\mathcal{E}_u$ concentrates around its mean, (b) the numerator accumulates nontrivial variance as the network grows, and (c) no single edge has outsized influence on T. Recall that $\mathcal{E}_u := \sum_{i\neq j} A_{ij,u}$ is the realized number of edges and $\mu_{\mathcal{E}} := \sum_{i\neq j} p_{ij}$ is the expected number of edges. Where conditional on $\mathbf{P}$, the edges $\{A_{ij,u}\}$ are independent and each score $\phi_{ij}(\mathbf{A}_u)$ is uniformly bounded and depends only on a finite neighborhood of the edge (i,j) ("edge-local"). As in Remark 1, we assume the edge-variance divergence condition $\sum_{i\neq j} p_{ij}(1-p_{ij}) \;\longrightarrow\; \infty$ as $N_F \to \infty$. This guarantees that the Bernoulli edge array contains plenty of randomness: the expected number of edges $\mu_{\mathcal{E}}$ diverges, and a law of large numbers implies $\mathcal{E}_u \;\xrightarrow{\mathbb{P}}\; \mu_{\mathcal{E}}$, so the denominator $\mathcal{E}_u$ behaves like its mean at the $\sqrt{\mu_{\mathcal{E}}}$

[13] We assume that, for each pair $(h,\ell) \in \mathcal{N}\times\mathcal{N}$, the firm-level averages inside the brackets converge as $N_F \to \infty$, so that $\Sigma_{\mathrm{O}}(\mathcal{N})$ is well-defined. Finiteness of these limits is automatic because $0 \le \vartheta^{\mathrm{O}}_i(h) \le 1$ for all i, h, so each term in the sum is uniformly bounded. Non-convergence would require a highly pathological sequence of networks in which the mix of firms' degree probabilities keeps changing back and forth as N_F grows. In more typical settings where additional firms are generated by random draws from some fixed (or slowly stabilizing) degree specification, these averages converge. Moreover, when the degree distribution is spread over many relatively fine bins so that the probability in any given bin is small, the products $\vartheta^{\mathrm{O}}_i(h)\vartheta^{\mathrm{O}}_i(\ell)$ are correspondingly small. In that case the off-diagonal covariances (which are proportional to $-\vartheta^{\mathrm{O}}_i(h)\vartheta^{\mathrm{O}}_i(\ell)$ on average) become small in magnitude relative to the diagonal terms, just as in a multinomial histogram where finer binning makes pairwise correlations weaker.

scale. Intuitively, the network does not drift into an ultra-sparse regime with only finitely many active edges.

In addition, we make assumptions that place both upper and lower bounds on how the variance of the statistic grows with the size of the network: we want the variance of $T(\mathbf{A}_u)$ neither to explode nor to collapse to zero as N_F increases. The upper bound is automatic from the fact that the scores $\phi_{ij}(\mathbf{A}_u)$ are uniformly bounded and edge-local: changing a single edge affects only a bounded number of bounded terms in the numerator, so no individual edge can have $O(1)$ leverage on T when $\mathcal{E}_u$ is large.

For a lower bound, we require that the statistic does not "wash out" the edge-level randomness by assigning vanishing weights to almost all edges. More specifically, we assume that

$$\frac{1}{\mu_{\mathcal{E}}} \sum_{i \neq j} p_{ij} \left(\mathbb{E}_{\mathbf{P}} \big[\phi_{ij}(\mathbf{A}_u) \big] \right)^2 \not\longrightarrow 0$$

Put simply, as we add firms and edges, the new edges that appear remain nontrivial for the statistic: they carry conditional scores whose typical magnitude does not collapse to zero. Given the upper and lower bounds on the variance, except for pathological cases, the variance will converge to a constant. Under these conditions, the numerator of $T(\mathbf{A}_u)$ is a sum of independent, uniformly bounded terms with variance of order $\mu_{\mathcal{E}}$, and the denominator $\mathcal{E}_u$ concentrates around $\mu_{\mathcal{E}}$. The Lindeberg–Feller CLT applied to the numerator, combined with Slutsky's theorem to handle the random denominator, yields

$$\sqrt{\mu_{\mathcal{E}}} \Big(T(\mathbf{A}_u) - \mathbb{E}_{\mathbf{P}}[T(\mathbf{A}_u)] \Big) \;\Rightarrow\; \mathcal{N}(0, \sigma_T^2), \qquad 0 < \sigma_T^2 < \infty$$

In particular, edge-averaged quantities such as reciprocity and degree assortativity are asymptotically normal, with fluctuations of order $\mu_{\mathcal{E}}(N_F)^{-1/2}$.[14] △

Comment 1. Remarks 1, 2, and 3 together imply that, for sufficiently large networks, a single Bernoulli draw from a given probability matrix $\mathbf{P}$ is, with high probability, representative of its degree distribution and edge-averaged global statistics, in the sense that these quantities concentrate around their expectations and satisfy central limit theorems.

[14] Many such statistics live on compact intervals (e.g. reciprocity in $[0, 1]$, assortativity in $[-1, 1]$). The CLT concerns the *centered, scaled* statistic; finite-sample distributions remain bounded, so the Gaussian is a local approximation, most accurate away from the boundaries.

3.3 Markov Regularity

Irreducible closure of the network

The Bernoulli reconstruction can produce a graph that is not irreducible: typically, the directed graph has multiple strongly connected components (SCCs). We therefore "close" the graph by adding a small number of cross-component edges so that the resulting network is strongly connected, while minimizing the disturbance to sectoral flows.

Step 1: SCCs, condensation, and component–level link counts

Let $\mathbf{A} \in \{0,1\}^{N_F \times N_F}$ be a Bernoulli draw (with $A_{ii} = 0$). Let $\mathcal{C} = \{1, \ldots, C\}$ be the set of SCCs of $\mathbf{A}$, $V(c)$ the set of firms in component $c \in \mathcal{C}$, and $\mathbf{A}_C$ the condensation directed acyclic graph (DAG). Define sources and sinks of the condensation by

$$\mathscr{T} = \Big\{c \in \mathcal{C} : \mathrm{indeg}_{\mathbf{A}_C}(c) = 0\Big\}, \qquad \mathscr{U} = \Big\{c \in \mathcal{C} : \mathrm{outdeg}_{\mathbf{A}_C}(c) = 0\Big\}$$

and set $R = \max\{|\mathscr{T}|, |\mathscr{U}|\}$ (naturally if there there is only one SCC $C = 1$, then $R = 0$). The idea here is that each source needs at least one sink and vice-versa for the graph to be irreducible.

We next decide how many component-level arcs to add between SCCs. Let $\mathcal{P} \subseteq \mathscr{U} \times \mathscr{T}$ be any collection of ordered sink–source pairs (u, t) such that adding one arc $u \to t$ for each $(u, t) \in \mathcal{P}$ would make the condensation $\mathbf{A}_C$ strongly connected; by the result above, we can choose $\mathcal{P}$ with $|\mathcal{P}| \geq R$, and we fix one such choice. For each ordered pair $(a, b) \in \mathcal{P}$, let

$$n_{ab} \;=\; \min\Big\{|V(a)|,\; |V(b)|\Big\}$$

denote the size of the smaller component. We choose a simple increasing function $f_\eta : \mathbb{N} \to (0, 1)$ that saturates as n grows; more specifically, one can take

$$f_\eta(n) \;=\; \theta\Big(1 - e^{-\eta n}\Big), \qquad 0 < \theta < 1,\; \eta > 0$$

The number of firm-level edges we plan to add from component a to component b is

$$k_{ab} \;=\; \Big\lceil f_\eta(n_{ab})\, n_{ab} \Big\rceil$$

so k_{ab} is weakly increasing in the size of the smaller SCC, never exceeds n_{ab}, and plateaus as n_{ab} grows. The total number of new links is $K = \sum_{(a,b)\in\mathcal{P}} k_{ab}$, and by construction $K \geq R$. Allowing more than one firm-level edge between two SCCs serves two purposes. First, it prevents

extremely thin bottlenecks, which is useful when we later impose conditions ensuring that small deviations from near-stationary weights do not push the vector of money holdings far from its stationary value. Second, and in a similar vein, convergence to stationarity can be very slow if two large blocks are connected by a single very weak link; a modest multiplicity of links between large SCCs avoids such pathologies while still keeping the number of additional edges small.

Step 2: firm-level placement of new links via a sectoral-flow objective

We now decide *where*, at the firm level, to place the k_{ab} links prescribed for each component pair $(a, b) \in \mathcal{P}$, choosing all new edges jointly so as to minimize the disturbance to sectoral flows at the level of the entire graph (so that imbalances created by some component pairs can be partially offset by others). Let $\pi(i) \in \{1, \dots, N_S\}$ be the sector of firm i, and let $\mathcal{F}_\ell = \{i : \pi(i) = \ell\}$ be the set of firms that belong to sector ℓ. Let $m_i > 0$ be size of firm i and $\mathbf{I}_{k\ell} \in \{0, 1\}$ the sectoral incidence (as in Section 3.1). Define baseline sectoral inflows

$$B_\ell \;=\; \sum_{k=1}^{N_S} \sum_{i \in \mathcal{F}_k} \sum_{j \in \mathcal{F}_\ell} m_i \, a_{ij} \, \mathbf{I}_{k\ell}, \qquad E_\ell \;=\; B_\ell - s_\ell, \quad s_\ell = \sum_{j \in \mathcal{F}_\ell} m_j$$

Note that here $a_{ij} \in \{0, 1\}$ indicates whether firm i buys from firm j, while $\mathbf{I}_{k\ell}$ encodes the sector-sector incidence and serves as a proxy for the weight of links between firms in sector k and firms in sector ℓ. For each $(a, b) \in \mathcal{P}$, let

$$\Omega_{ab}^{\text{raw}} \;=\; \Big\{(i, j) : i \in V(a),\; j \in V(b),\; a_{ij} = 0\Big\}$$

denote the set of all absent firm-level edges from component a to component b. To keep the optimization problem computationally tractable, we do not use all of Ω_{ab}^{raw}. Instead, we randomly select a smaller subset of candidate edges whose cardinality grows at most linearly in the size of the smaller component. More precisely, recall that $n_{ab} = \min\{|V(a)|, |V(b)|\}$ and k_{ab} is the prescribed number of new links from a to b. We fix constants $\bar{\gamma} > 0$, $n_0 \in \mathbb{N}$, and $\eta > 0$, and define an explicit increasing function

$$g_\eta(n) \;:=\; \bar{\gamma} \left(\frac{n}{n_0 + n}\right)^\eta, \qquad n \in \mathbb{N},$$

which is increasing in n and satisfies $g_\eta(n) \uparrow \bar{\gamma}$ as $n \to \infty$. The parameter η controls the speed at which $g_\eta(n)$ approaches its saturation level $\bar{\gamma}$. We then set

$$L_{ab} \;=\; \max\Big\{k_{ab},\; \big\lceil g_\eta(n_{ab})\, n_{ab} \big\rceil\Big\}$$

and sample, uniformly at random and without replacement, a subset

$$\Omega_{ab} \subseteq \Omega_{ab}^{\text{raw}} \quad \text{with} \quad |\Omega_{ab}| = L_{ab}$$

By construction $|\Omega_{ab}|$ is $O(n_{ab})$ and $|\Omega_{ab}| \geq k_{ab}$, so there are always enough candidates to place the prescribed k_{ab} links while ensuring that the total number of decision variables grows at most linearly with network size. Finally, we define the global candidate set

$$\Omega \;=\; \bigcup_{(a,b)\in\mathcal{P}} \Omega_{ab}$$

We choose new edges $x_{ij} \in \{0,1\}$ only for absent, cross-SCC pairs in Ω, and solve a single optimization problem for the whole graph:

$$\min_{x} \;\; \sum_{\ell=1}^{N_S} \frac{1}{s_\ell^2} \left(E_\ell + \sum_{(i,j)\in\Omega} a_{ij,\ell}\, x_{ij} \right)^2$$

that is, we minimize a quadratic loss in sectoral inflows, normalized by squared sector size, so that proportional disturbances to sectoral sizes are penalized equally across sectors[15]. This objective is subject to the following constraints:

Binary decisions:	$x_{ij} \in \{0,1\} \quad \forall (i,j) \in \Omega$
Prescribed number of edges between SCCs:	$\sum_{(i,j)\in\Omega_{ab}} x_{ij} \;=\; k_{ab} \quad \forall (a,b) \in \mathcal{P}$

The first step fixes how many links are added between each sink–source SCC pair, ensuring strong connectivity at the component level with a small and size-adaptive number of arcs. The second step then solves a single global optimization problem that allocates these k_{ab} links to specific firm pairs drawn from the thinned candidate sets Ω_{ab}, allowing sectoral imbalances created by some SCC pairs to be offset by choices in others, and thereby keeping sectoral inflows—and hence sectoral sizes—as close as possible to their Bernoulli baseline.

[15] At this stage we do not yet impose any stationary-money constraint, and we therefore do not attempt to track how the new links would propagate changes in sectoral inflows through multi-step dynamics. This issue is addressed in the subsequent step of the reconstruction procedure, where we compute edge weights under a stationary-money constraint that pins down sectoral sizes. The present step is only designed to ensure that, among all irreducible completions of the Bernoulli graph, we select one that makes it possible to stay close to the original sectoral sizes once weights are chosen.

Aperiodic closure of the network. Let $\widehat{\mathbf{A}}$ denote the irreducible adjacency obtained above. We enforce aperiodicity by adding self-loops on every firm:

$$\widetilde{\mathbf{A}} \;:=\; \widehat{\mathbf{A}} \;+\; \mathbf{I}$$

At this stage the graph remains unweighted. Weights are assigned in Section 3.4 using $\widetilde{\mathbf{A}}$ as the binary backbone[16].

3.4 Computing Weights of the Production Network

We convert the (irreducible and aperiodic) binary adjacency $\widetilde{\mathbf{A}} \in \{0,1\}^{N_F \times N_F}$, with entries $a_{ij} \in \{0,1\}$ (including self-loops), into a row-stochastic matrix $\mathbf{W} = (w_{ij}) \in [0,1]^{N_F \times N_F}$ supported on $\widetilde{\mathbf{A}}$ (i.e., $w_{ij} = 0$ whenever $a_{ij} = 0$, and $w_{ij} > 0$ whenever $a_{ij} = 1$). Recall that $\mathbf{m} = (m_i)$ denotes firm sizes. Let $\pi : \{1, \ldots, N_F\} \to \{1, \ldots, N_S\}$ map firms to sectors and $\mathcal{F}_\ell = \{i : \pi(i) = \ell\}$ be the set of firms in sector ℓ. Define observed sector sizes $s_\ell = \sum_{j \in \mathcal{F}_\ell} m_j$ and the model-implied sector size $\widehat{s}_\ell$ as:

$$\widehat{s}_\ell(\mathbf{W}) \;:=\; \sum_{i=1}^{N_F} m_i \sum_{j \in \mathcal{F}_\ell} w_{ij}$$

which is to say that the sectoral totals are computed from allocations of each firm's outflow across destinations.

We estimate $\mathbf{W}$ via the convex quadratic program

$$\min_{\mathbf{W}} \; \sum_{i=1}^{N_F} \sum_{j=1}^{N_F} w_{ij}^2$$

subject to

$$\sum_{j=1}^{N_F} w_{ij} = 1 \;\; (\forall i) \qquad \text{(row-stochasticity)}$$

$$0 \le w_{ij} \le a_{ij} \;\; (\forall i, j) \qquad \text{(support and bounds)}$$

[16] Markov closure can be implemented either via a universal 'closure' hub that trades with every firm (Mandel and Veetil, 2021), or by minimally augmenting the firm graph to make it strongly connected and enforcing aperiodicity via self-loops. We adopt the latter. A universal hub guarantees irreducibility and aperiodicity but also induces excessive reflux ($i \to H \to j$), acting as a dense rank-one channel that synchronizes flows and creates an outlier eigenvalue dominating dynamics (Baik, Ben Arous and Péché, 2005). Such a hub accelerates mixing and hastens convergence to stationarity (Langville and Meyer, 2006, Thm. 4.7.1), which is undesirable when studying out-of-equilibrium propagation and persistent post-shock dynamics.

$$w_{ij} > 0 \text{ whenever } a_{ij} = 1 \ (\forall i, j) \qquad \text{(no zero weights on existing edges)}^{17}$$

$$\left(\frac{(\mathbf{W}^\top \mathbf{m})_j - m_j}{m_j} \right)^2 \ \leq \ \delta^2 \ (\forall j) \qquad \text{(firm-level L2 relative error; } \delta \text{ small)}$$

$$\left| \frac{\widehat{s}_\ell(\mathbf{W}) - s_\ell}{s_\ell} \right| \ \leq \ \varepsilon \ (\forall \ell) \qquad \text{(sectoral size L2 deviation; } \varepsilon \text{ small)}$$

$$\frac{1}{N_F} \sum_{i=1}^{N_F} w_{ii} \ \leq \ \eta_1, \qquad \frac{1}{N_F} \sum_{i=1}^{N_F} w_{ii}^2 \ \leq \ \eta_2 \qquad \text{(self-weight moment bounds; } \eta_1, \eta_2 \text{ small)}$$

Row-stochasticity allocates each firm's outflow fully across its outgoing links. Naturally, there are N_F such equalities. The support and bound conditions ensure all non-edges get zero weights and keep weights within $[0, 1]$. Positivity on existing edges preserves the realized binary backbone (including self-loops), which in particular implies $w_{ii} > 0$ and thus aperiodicity. The firm-level L2 condition enforces small one-step budget consistency at every destination j. There are N_F such inequalities, with the common tolerance $\delta > 0$ chosen to be small. The sectoral L2 condition keeps model-implied sector totals close to observed sector sizes. There are N_S such inequalities, controlled by a small $\varepsilon > 0$. Finally, the two self-weight moment bounds regularize the prevalence and dispersion of self-flows across firms, preventing degenerate solutions with overly large w_{ii}[18]. Both η_1 and η_2 are small bounds.

The objective $\|\mathbf{W}\|_F^2 = \sum_{ij} w_{ij}^2$ is strictly convex and all constraints are affine or convex quadratic. Hence, the program is a convex QP with a unique solution whenever the feasible set is nonempty.

Remark 4 (Stationary money near the empirical firm-size vector). By construction, the estimated matrix $\mathbf{W}$ is row-stochastic, irreducible, and aperiodic, and the firm-level constraints

$$\left(\frac{(\mathbf{W}^\top \mathbf{m})_j - m_j}{m_j} \right)^2 \ \leq \ \delta^2, \qquad \forall j$$

enforce one-step consistency: for each firm j, the one-step update $(\mathbf{W}^\top \mathbf{m})_j$ is within a relative error δ of the observed firm-size m_j. Let $M = \mathbf{1}^\top \mathbf{m} > 0$ and normalize to $\boldsymbol{\mu} = \mathbf{m}/M$. Let

[17] In numerical implementations one may impose $w_{ij} \geq \varepsilon_0 \, a_{ij}$ with a tiny $\varepsilon_0 > 0$.

[18] A common design choice is whether to control self-weights via the objective or via constraints. We keep the objective purely minimum-energy, $\min \sum_{ij} w_{ij}^2$, and regulate self-flows through explicit moment bounds on $\{w_{ii}\}$ (the η_1, η_2 constraints), which gives direct, interpretable control over the prevalence and dispersion of self-weights (Mandel and Veetil, 2021). If self-weights are allowed to grow large, $\mathbf{W}$ becomes nearly diagonal and Gershgorin's theorem plus standard perturbation bounds imply a spectrum bunched near 1 and eigenvectors localized at individual nodes, which in practice is associated with inhibited diffusion (Goltsev et al., 2012). Large self-weights trap flows locally: shocks recycle at the originating firms instead of propagating along supplier–buyer links. The η_1, η_2 constraints rule out this degeneracy while leaving the objective focused on energy minimization.

$\boldsymbol{\nu}^\top = \boldsymbol{\nu}^\top \mathbf{W}$ be the stationary distribution of $\mathbf{W}$, normalized by $\mathbf{1}^\top \boldsymbol{\nu} = 1$. Define the one-step residual

$$\mathbf{r} \;:=\; \boldsymbol{\mu} - \mathbf{W}^\top \boldsymbol{\mu} \;=\; \frac{1}{M}\big(\mathbf{m} - \mathbf{W}^\top \mathbf{m}\big)$$

From the constraints above we obtain

$$|r_j| = \left|\frac{m_j - (\mathbf{W}^\top \mathbf{m})_j}{M}\right| \le \frac{\delta\, m_j}{M} = \delta\, \mu_j$$

so $\|\mathbf{r}\|_1 = \sum_j |r_j| \le \delta \sum_j \mu_j = \delta$. A single step of the Markov chain moves the normalized empirical money vector $\boldsymbol{\mu}$ only slightly: the total L^1 discrepancy between $\boldsymbol{\mu}$ and $\mathbf{W}^\top \boldsymbol{\mu}$ is at most δ. To convert this small one-step residual into a bound on the distance to stationarity, we use the spectral gap of the chain. Let

$$\gamma \;:=\; 1 - |\lambda_2(\mathbf{W})|$$

be the spectral gap, where $\lambda_2(\mathbf{W})$ is the second eigenvalue in modulus. For a fixed finite, irreducible, aperiodic Markov chain we have $\gamma > 0$. Intuitively, a larger γ means faster mixing and fewer 'almost-invariant' directions in which $\mathbf{W}^\top v \approx v$ while v is still far from the stationary distribution.

We can now write the distance between $\boldsymbol{\mu}$ and $\boldsymbol{\nu}$ as follow:

$$\begin{aligned}(\mathbf{I} - \mathbf{W}^\top)(\boldsymbol{\mu} - \boldsymbol{\nu}) &= (\boldsymbol{\mu} - \mathbf{W}^\top \boldsymbol{\mu}) - (\boldsymbol{\nu} - \mathbf{W}^\top \boldsymbol{\nu}) \\ &= \mathbf{r} - \mathbf{0}\end{aligned}$$

Now note that since both $\boldsymbol{\mu}$ and $\boldsymbol{\nu}$ are probability vectors, their difference lies in the zero-sum subspace $\mathcal{H} := \{\mathbf{u} : \mathbf{1}^\top \mathbf{u} = 0\}$. On $\mathcal{H}$, all eigenvalues of $\mathbf{W}^\top$ have modulus at most $1 - \gamma$, so the linear map $\mathbf{I} - \mathbf{W}^\top$ has eigenvalues $1 - \lambda_k$ satisfying $|1 - \lambda_k| \ge \gamma$. Thus $\mathbf{I} - \mathbf{W}^\top$ is invertible on $\mathcal{H}$, and its inverse has operator norm at most $1/\gamma$ in L^1. Applying $(\mathbf{I} - \mathbf{W}^\top)^{-1}$ and taking L^1 norms:

$$\begin{aligned}(\boldsymbol{\mu} - \boldsymbol{\nu}) &= (\mathbf{I} - \mathbf{W}^\top)^{-1}\mathbf{r} \\ \|\boldsymbol{\mu} - \boldsymbol{\nu}\|_1 &= \left\|(\mathbf{I} - \mathbf{W}^\top)^{-1}\mathbf{r}\right\|_1 \\ &\le \left\|(\mathbf{I} - \mathbf{W}^\top)^{-1}\right\|_1 \|\mathbf{r}\|_1 \\ &\le \frac{\delta}{\gamma}\end{aligned}$$

Thus, for a fixed network with spectral gap $\gamma > 0$, tightening the one-step fit (by choosing

smaller δ in the weighting program) forces the stationary money vector $\boldsymbol{\nu}$ to be arbitrarily close, in L^1, to the normalized empirical firm-size vector $\boldsymbol{\mu} = \mathbf{m}/M$.

4 Computational Complexity of the Algorithm

We assume that the network is sparse, and that the number of sectors is sizeably smaller than the number of firms $N_S \ll N_F$. We first evaluate the complexity of our four-step algorithm under serial execution, then under parallel execution.

4.1 Serial Execution

Step 1: Logistic-gravity

In the logistic-gravity step, the naive cost of evaluating the objective and constraints is $O(N_F^2)$ per solver iteration, because in principle all firm pairs (i, j) must be visited on each pass. We can reduce this cost by grouping firms into B size bins within each sector and working with sector-bin summaries instead of individual firms. Concretely, a one-time $O(N_F)$ preprocessing pass assigns each firm to a sector-bin cell and accumulates bin-level quantities (e.g., counts or power sums). Thereafter, each evaluation of the objective, constraints, and gradients only loops over sector-bin pairs, with cost $O(N_S^2 B^2)$, which is independent of N_F when N_S and B are fixed.[19] Thus the overall cost of the logistic-gravity step is $O(N_F + T N_S^2 B^2)$ for T iterations, i.e., essentially linear in N_F when T, N_S, and B are moderate. In large-scale convex logistic models, first-order methods typically require a number of iterations that grows only modestly with dimension (Nesterov, 2004), so per-iteration cost is the dominant factor and binning removes the quadratic bottleneck in N_F.

Step 2: Bernoulli ensemble

In a naive scheme, we would draw independent Bernoulli links over all admissible firm pairs, which costs $O(N_F^2)$ per draw because there are $O(N_F^2)$ potential edges. In sparse production networks, however, the number of nonzero links grows only linearly with the number of firms, so the expected edge count is $E = O(N_F)$. If we exploit sparsity by restricting attention to admissible pairs, the expected work becomes proportional to the number of realized edges, i.e.

[19] *Binning error.* Let $\phi(x_i, x_j)$ be the smooth pairwise score inside the logistic link. Replacing features by bin centroids induces a second-order error controlled by the within-bin diameter Δ: if $\nabla^2\phi$ is bounded, the per-pair error is $O(\Delta^2)$, and the aggregate objective/gradient discrepancy per iteration is $\varepsilon_{\text{bin}} = O\big(\sum_{\text{bins}} \text{within-bin variance}\big)$. Under equal-width binning on a bounded range, $\Delta = O(1/B)$ and hence $\varepsilon_{\text{bin}} = O(B^{-2})$.

$O(E) = O(N_F)$ (Batagelj and Brandes, 2005). Subsequent pruning of isolated nodes is also $O(N_F)$, so the Bernoulli-ensemble stage runs in $O(N_F)$ time in expectation under sparsity.

Step 3: Markov regularity (connectivity augmentation and self-loops)

On a sparse support with $E = O(N_F)$, the strongly connected components (SCCs) of the Bernoulli draw can be found in $O(N_F)$ time using standard linear-time algorithms (Tarjan, 1972). Constructing the condensation digraph, computing in-degree and out-degrees, and identifying source and sink components $\mathscr{T}, \mathscr{U}$ are likewise $O(N_F)$. If we were only interested in adding the minimum number of arcs to make the condensation strongly connected, the Eswaran–Tarjan procedure achieves this in linear time in the size of the condensation graph (Eswaran and Tarjan, 1976).

In our implementation, however, the choice of cross-component edges is governed by a sectoral-flow objective with binary decision variables x_{ij}, as described in Step 2. For each ordered component pair $(a, b) \in \mathcal{P}$ we start from the set Ω_{ab}^{raw} of all absent firm-level edges from a to b, but we then randomly thin this set to a subset Ω_{ab} of size $L_{ab} = \max\Big\{k_{ab},\ \big\lceil g_\zeta(n_{ab})\, n_{ab} \big\rceil\Big\}$, where n_{ab} is the size of the smaller component and $g_\zeta : \mathbb{N} \to (0, \bar{\gamma}]$ is an increasing function that saturates at a constant level $\bar{\gamma} > 0$. By construction $L_{ab} = O(n_{ab})$ and $L_{ab} \geq k_{ab}$, so each pair (a, b) has enough candidates to place the prescribed number k_{ab} of new links while keeping $|\Omega_{ab}|$ at most linear in component size. The global candidate set is $\Omega = \bigcup_{(a,b)\in\mathcal{P}} \Omega_{ab}$, and when the number of SCCs grows sublinearly in N_F this implies $|\Omega| = O(N_F)$.

The resulting closure problem is a mixed-integer quadratic program with one binary variable per candidate edge in Ω. Such problems are NP-hard in general, and we solve this stage with a generic mixed-integer QP solver, so no polynomial worst-case bound is available. However, the combination of the SCC structure and the thinning procedure ensures that the number of binary variables grows only linearly with network size in our setting, rather than quadratically in N_F as it would on the full $N_F \times N_F$ grid. This substantially reduces the effective search space and yields instances that are easily handled in practice for the network sizes we consider. Self-loops for aperiodicity are then added in a final $O(N_F)$ pass.

Step 4: Minimum-energy weighting QP

Weights live only on realized edges, so the number of variables is $E = \sum_i \deg(i)$, which is $E = \Theta(N_F)$ under sparsity. Each iteration of a first-order method or projected-gradient scheme consists of sparse passes: (i) computing firm-level updates $(\mathbf{W}^\top \mathbf{m})_j$ and sector totals $\widehat{s}_\ell(\mathbf{W})$ via sparse matrix-vector operations, and (ii) enforcing row-stochasticity and bounds by projecting each row of (w_{ij}) onto a (capped) probability simplex. Both (i) and (ii) cost $O(E)$ per iteration,

since $\sum_i \deg(i) = E$. Thus the per-iteration arithmetic cost is $O(E) = O(N_F)$. If I_{QP} denotes the number of iterations required to reach the desired accuracy, the total cost of the weighting step is $O(I_{\mathrm{QP}} N_F)$. For fixed tolerances and moderate I_{QP}, this stage is therefore essentially linear in the number of firms.

Comment 2 (Serial Complexity). All of this means that in a fully naive serial implementation, the dominant cost comes from (i) evaluating the logistic-gravity objective and constraints over all firm pairs (i, j) and (ii) drawing Bernoulli links over all admissible pairs. Both of which are $O(N_F^2)$ per pass. The minimum-energy weighting QP is also quadratic in the worst case if treated as a dense matrix problem. Thus, at the level of big-O serial scaling $T_{\text{serial}}^{\text{exact}} = O(N_F^2)$. Under sparsity and binning, the picture changes: the logistic-gravity step is reduced to $O(N_F)$ preprocessing plus $O(N_S^2 B^2)$ work per iteration over sector-bin pairs (independent of N_F for fixed N_S, B). The Bernoulli ensemble uses edge-skipping on a sparse support with $E = \Theta(N_F)$. Since the minimum-energy weighting QP operates only on realized edges, with per-iteration cost $O(N_F)$. For fixed N_S, B and modest iteration counts, the overall serial runtime scales essentially linearly in the number of firms $T_{\text{serial}}^{\text{binned}} = O(N_F)$ up to constants depending on N_S, B, and solver iteration counts. These should be interpreted as practical scaling laws under sparsity, rather than worst-case complexity guarantees.

4.2 Parallel Execution

Let φ denote the number of threads. We shall now assess the complexity of each step of our network reconstruction algorithm when run on φ parallel threads.

Step 1: Logistic-gravity

Since the computation of pairwise firm intensity is independent of each other, they can be wholly parallelized, which means that per-iteration complexity is $O(N_F^2/\varphi)$ in the exact case and $O(N_F/\varphi)$ with binning (after an $O(N_F\varphi)$ preprocessing pass).

Step 2: Bernoulli ensemble

Bernoulli draws for different pairs of firms are independent of each other. Therefore, a procedure that considers only non-zero probabilities in sparse networks yields a complexity of $O(N_F/\varphi)$.

Step 3: Markov regularity (connectivity augmentation and self-loops)

Unlike the computations in Steps 1–2, the identification of strongly connected components is

not easily decomposed over many threads. To decide whether two firms lie in the same SCC, one is in effect concerned with directed paths that may traverse arbitrarily many intermediate nodes, so the procedure depends on the global connectivity structure of the graph rather than on local, edgewise operations. This makes it difficult to partition the work cleanly across threads without incurring non-trivial synchronization and communication overhead.[20] For this reason we retain the effective complexity of the SCC and condensation steps as $O(N_F)$ on a sparse support with $E = O(N_F)$ even in a parallel implementation.

Once SCCs and the condensation digraph have been computed, we form inter-component candidate sets $\Omega_{ab}^{\mathrm{raw}}$ and then thin them to subsets Ω_{ab} of size $L_{ab} = \max\Big\{k_{ab},\ \big\lceil g_\zeta(n_{ab})\, n_{ab}\big\rceil\Big\}$ where n_{ab} is the size of the smaller component and g_ζ is a plateauing function. By construction $L_{ab} = O(n_{ab})$ and, when the number of SCCs grows sublinearly in N_F, the total number of candidate edges satisfies $|\Omega| = \sum_{(a,b)} |\Omega_{ab}| = O(N_F)$. The closure stage then solves a 0–1 quadratic program over these $|\Omega|$ variables to select the cross-SCC edges. Although modern mixed-integer QP solvers can exploit several cores internally, the objective couples all x_{ij} through global sectoral-flow constraints, so the search cannot be decomposed into embarrassingly parallel subproblems. As a result, parallel hardware yields mainly constant-factor speedups rather than a different asymptotic order, and we treat the wall-clock cost of the closure stage as growing linearly with N_F for fixed solver settings. The final insertion of self-loops to ensure aperiodicity is an embarrassingly parallel pass over firms: with φ threads its wall-clock cost is $O(N_F/\varphi)$, which is negligible compared with the preceding steps and does not affect the overall asymptotic order.

Step 4: Minimum-energy weighting QP

The minimum-energy reweighting stage solves a quadratic program over the row-stochastic weight matrix W. Ignoring global constraints, the basic workhorse operation is a rowwise capped-simplex projection: each firm i projects its outgoing weights $(w_{ij})_j$ onto a truncated simplex under box constraints. These projections decouple across rows and therefore parallelize straightforwardly across φ threads, giving $O(N_F/\varphi)$ arithmetic per iteration.

To keep the chain's stationary size vector close to m and to enforce sector consistency, each iteration must additionally form (i) the firm-level column vector $g = W^\top m$, (ii) sector totals $\widehat{s}_\ell(W)$, and (iii) two moments of the self-weights (e.g. $\sum_i w_{ii}$ and $\sum_i w_{ii}^2$). These are global quantities and thus require communcation across threads. Let $\mathcal{R}_\varphi$ denote the per-iteration cost of

[20] Parallel algorithms for SCC do exist and are used in large-scale graph analytics, but they typically pay a coordination overhead and achieve only moderate speedups compared with embarrassingly parallel primitives. We therefore do not treat them as changing the relevant asymptotic complexity in our setting.

these reductions. In the standard α–β (latency–bandwidth) model for all-reduce[21], we have the ballpark bound $\mathcal{R}_\varphi \;\le\; c_\beta N_F + c_\alpha \log \varphi$ since the bandwidth term scales with N_F and latency contributes $O(\log \varphi)$ or $O(\varphi)$ depending on the algorithm (Rabenseifner, 2004). On shared memory, reductions are essentially memory-bandwidth limited and $\mathcal{R}_\varphi = \Theta(N_F)$ up to a hardware constant. Hence, at the level of approximation relevant for this paper, the per-iteration wall-clock time of the QP in our parallel setting is $T_{\mathrm{QP}}(\varphi) \;=\; \Theta\Big(N_F/\varphi + \mathcal{R}_\varphi\Big)$ with $\mathcal{R}_\varphi$ capturing the communication cost associated with enforcing the global Markov and sectoral constraints.

Comment 3 (Parallel Complexity). For the exact (non-binned) pipeline, aggregating the costs over all steps, the parallel wall-clock time satisfies $T^{\mathrm{exact}}_{\mathsf{parallel}} \;=\; \Theta\Big(\frac{N_F^2}{\varphi} + \frac{N_F}{\varphi} + \mathcal{R}_\varphi\Big)$ where the quadratic term comes from the pairwise logistic–gravity intensities, the linear term from Bernoulli draws and rowwise projections, and $\mathcal{R}_\varphi$ collects the global reductions needed to enforce the Markov and sectoral constraints in Step 4. In what follows we simply treat $\mathcal{R}_\varphi = O(N_F)$ as a communication term that limits parallel scaling, so that a standard implementation yields $T^{\mathrm{exact}}_{\mathsf{parallel}} \;=\; \Theta\Big(N_F^2/\varphi + N_F\Big)$ with the N_F^2/φ term dominating for moderate φ and the $O(N_F)$ communication term eventually becoming the bottleneck as φ grows. With sector-wise binning (fixed B), the leading N_F^2 work in Step 1 collapses to $O(N_F)$ and the parallel complexity becomes $T^{\mathrm{binned}}_{\mathsf{parallel}} \;=\; \Theta\Big(\frac{N_F}{\varphi} + \mathcal{R}_\varphi\Big)$. Under our ballpark assumption $\mathcal{R}_\varphi = O(N_F)$, this is $\Theta(N_F)$: the local rowwise work scales down with φ, but global reductions impose a linear communication floor. More elaborate implementations could reduce the effective $\mathcal{R}_\varphi$, but we do not model these refinements explicitly. In all cases, the MILP stage in Step 3 remains NP-hard and is treated as a black-box term whose runtime can benefit from multicore hardware but does not change the asymptotic worst-case complexity of the pipeline[22].

Table 2 summarizes the computational complexity of our network reconstruction algorithm for sparse networks with sector-wise binning for serial and parallel execution. Serial complexity is given in terms of N_F (firms), N_S (sectors), B (bins), $E = \Theta(N_F)$ (edges), and iteration counts.

[21] In the α–β model, communication time is approximated as $T_{\mathsf{comm}} \approx \alpha m + \beta w$, where α is the per-message latency, β is the time per word, m is the number of messages, and w is the number of words. For all-reduce of a length-n vector across φ workers, standard tree-based algorithms achieve $T = O(\alpha \log \varphi + \beta n)$ (Rabenseifner, 2004). In our discussion we simply treat the bandwidth term as $O(n)$.

[22] The computational complexity of our algorithm compares favourably with many commonly used firm-level network reconstruction methods in economics and finance, such as IPFP/RAS/Sinkhorn and maximum-entropy (ECM) schemes with heterogeneous firm-level constraints and unknown support. In these methods, the reconstruction is typically carried out on a dense $N_F \times N_F$ table: each iteration rescales or updates all rows and columns to match firm-level marginals, and normalizing constants or gradients aggregate over all potential pairs. As a result, their per-iteration work is usually of order $\Theta(N_F^2)$ unless a correct sparse structural-zero mask or other special structure is available. Our discussion is limited to this generic firm-level setting and does not attempt to cover specialised variants that exploit additional structure to reduce complexity in particular cases.

Parallel complexity assumes φ worker threads and treats global reductions and the closure problem as communication-limited and or solver-limited terms.

Table 2: Computational complexity of each step of the network reconstruction algorithm

Step	**Description**	**Serial**	**Parallel**
1: Logistic–gravity	Binned logistic–gravity over sector–size cells to estimate P_{ij}.	$O(N_F)+O(T\,N_S^2B^2)$	$O(N_F/\varphi)+$ $O(T\,N_S^2B^2/\varphi)$
2: Bernoulli ensemble	Draw sparse Bernoulli backbone and prune isolated nodes.	$O(N_F)$	$O(N_F/\varphi)$
3: Markov regularity	Compute SCCs, solve closure problem on inter-SCC candidates, then add self-loops.	$O(N_F)$	$O(N_F)$
4: Weights-assignment	Minimum-energy reweighting on fixed support to match stationary firm sizes and sectoral totals.	Per iteration $O(N_F)$; total $O(I_{\mathrm{QP}}N_F)$	Per iteration $O(N_F/\varphi+\mathcal{R}_\varphi)$; total $O(I_{\mathrm{QP}}(N_F/\varphi+\mathcal{R}_\varphi))$

5 Reconstruction of the Complete US Production Network

As discussed above, our reconstruction algorithm scales well. In this section we illustrate its capacity to handle large-scale production networks by reconstructing the production network of the full US economy, with more than 5×10^6 firms and around 10^8 firm-to-firm links. To the best of our knowledge, this is the largest production network reconstruction to date. Section 5.1 describes the publicly available data that we use as inputs. Section 5.2 presents the empirical distributions of the parameters α, κ, z, and λ. It shows that they are tightly concentrated around their means. Section 5.3 then presents the reconstructed US production network: we report degree distributions and standard network summary statistics. We also compare these statistics with those from the largest available sample of the real US firm-to-firm network.

5.1 The Data

We use two sets of publicly available data to reconstruct the US production network. The first set pertains to the size distribution of firms by sector. This dataset comes from the Small Business Administration (SBA). It includes nearly all firms with employees in the US economy, amounting to about 6.28 million firms. The dataset does not contain information on the exact size of firms;

instead, it places each firm into one of nine size bins.[23] Within each sector-size-bin cell we assume firms are uniformly distributed and sample firm sizes accordingly. The second dataset pertains to sectoral flows. More specifically, we use the input-output table provided by the Bureau of Economic Analysis (BEA) of the US. We aggregate industries to an $S = 24$ sector system and form a 24×24 matrix of sectoral flows.

Unfortunately, the two datasets cannot be readily used together, as the SBA dataset on firm sizes codes sectors using the NAICS classification, whereas the BEA input-output table uses the SIC classification. We map NAICS sectors to BEA IO industries using the official BEA concordances. This concordance is not a one-to-one map and does not cover all sectors. As a result, we had to drop firms for which there was no clear sector code, leaving us with about 5.4 million firms for which we know sectors, size bins, and the flows between the sectors to which they belong.

5.2 Estimation procedure and parameter values

In this section, we detail the procedures used to sample firms from the SBA size distribution of firms by sector, estimate the gravity parameters $(z, \alpha, \kappa, \{\lambda_{k\ell}\})$ by solving a NLP problem, draw the Bernoulli backbone of the network, enforce Markov regularity, and solve a QP problem to compute weights.

5.2.1 Sampling firms from SBA data on firm size by sector

The SBA data provide, for each sector, counts of firms in fixed size bins. We partition this table into *cells*, where each cell corresponds to a sector–size–bin combination, and draw a random subsample of firms from each cell. Let $r \in (0, 1)$ denote the fraction of firms in the economy that we retain. For a cell with c firms, we first set $\tilde{c}_0 = \lfloor rc \rfloor$ and then, with probability $rc - \lfloor rc \rfloor$, we increase this by one. Thus the final number of sampled firms in the cell is $\tilde{c} = \tilde{c}_0 \quad \text{or} \quad \tilde{c}_0 + 1$ with $\mathbb{E}[\tilde{c}] = rc$. For example, if $c = 7$ and $r = 0.3$, then $rc = 2.1$: we take 2 firms for sure, and a third firm with probability 0.1. We then assign sizes to these $\tilde{c}$ by drawn from a uniform distribution for which the bin size serves as the support. This yields a synthetic population of firms, each with an ID, a sector (SIC), and a size m_i.

[23] The bin breakpoints are [0, 100k), [100k, 500k), [500k, 1m), [1, 2.5m), [2.5, 5m), [5, 10m), [10, 50m), [50, 250m), [250m, 1b]. We assume an upper bound of 1 billion for the highest bin.

5.2.2 Estimation of the parameter values in the gravity equation

On the sampled population we estimate the augmented gravity model embedded in the logistic link function specified in Section 3. The objective and constraints in that section define a smooth, sparse nonlinear program in the parameters $(z, \{\lambda_{k\ell}\}, \alpha, \kappa)$. We solve this program with IPOPT, a primal–dual interior–point solver for constrained nonlinear optimization.[24] IPOPT takes as input the objective, constraints, and their derivatives, and returns a point that approximately satisfies first–order optimality and all constraints.

Stable parameterization and the solvers

To keep the nonlinear program numerically stable, we optimize over unconstrained variables and map strictly positive parameters into $(0, \infty)$ via an exponential transform: $z = \exp(\zeta), \lambda_{k\ell} = \exp(\ell_{k\ell})$, with $\zeta, \ell_{k\ell} \in \mathbb{R}$. This reparameterization is one–to–one and does not change the model; it simply enforces positivity automatically and avoids numerical issues when iterates approach zero. The remaining parameters, such as α and κ, are handled with simple bound constraints directly within IPOPT. We provide IPOPT with analytical derivatives[25]. More specifically, we derive and feed the analytic gradients and the corresponding constraint Jacobians. Recall the following intensity equation and probability function.

$$x_{ij} \;=\; z\,\lambda_{\pi(i)\pi(j)}\,S^{\kappa}_{\pi(i)\pi(j)}\,(m_i m_j)^{\alpha}, \qquad p_{ij} \;=\; \frac{x_{ij}}{1+x_{ij}}$$

The derivative of p_{ij} with respect to any parameter θ has the compact form

$$\frac{\partial p_{ij}}{\partial \theta} \;=\; p_{ij}(1-p_{ij})\,\frac{\partial \log x_{ij}}{\partial \theta}$$

and

$$\frac{\partial \log x_{ij}}{\partial \zeta} = 1, \qquad \frac{\partial \log x_{ij}}{\partial \ell_{k\ell}} = \mathbb{1}_{\{\pi(i)=k,\ \pi(j)=\ell\}}$$
$$\frac{\partial \log x_{ij}}{\partial \alpha} = \log m_i + \log m_j, \qquad \frac{\partial \log x_{ij}}{\partial \kappa} = \log S_{\pi(i)\pi(j)}$$

[24] IPOPT is a primal–dual interior–point method: inequality constraints are handled by adding a barrier term to the objective and solving a sequence of barrier subproblems. In each subproblem, the iterate is kept strictly inside the feasible region, and a barrier parameter $\mu > 0$ controls how strongly the constraints are enforced. IPOPT automatically decreases μ as the method progresses, so that the iterates move from a safely interior region toward the boundary where the original constraints are tight. This continuation strategy is well suited to large, sparse problems with nonlinear constraints and avoids the combinatorial complexity of active set methods.

[25] Supplying analytic derivatives allows IPOPT to compute search directions using accurate first–order information rather than finite–difference approximations. This improves robustness, avoids additional function evaluations, and typically reduces overall solve time, especially in large sparse problems.

with $\mathbf{1}_{\{\pi(i)=k,\ \pi(j)=\ell\}}$ denoting the indicator function: for unsupported sector pairs with $S_{k\ell} = 0$, the corresponding multipliers are inactive and their derivatives are set to zero in the implementation.

We also use a simple warm start obtained by raking the $\lambda_{k\ell}$ columns and adjusting z by bisection to match the target mean degree, so the solver starts from a feasible point that is already close to the desired scale. Furthermore, in our implementation, IPOPT is run in limited–memory quasi–Newton mode (L–BFGS) with a sparse direct linear solver (MA97).[26] MA97 is designed for speed and numerical stability on the KKT systems produced by interior–point methods. We scale the objective and constraints so that the main quantities of interest are of comparable magnitude, which helps IPOPT balance the gradient and constraint residuals. We then set relatively tight stopping tolerances for both optimality and feasibility, requiring small residuals in the Lagrangian gradient and in all constraints before termination. This ensures that the final parameter estimates yield probabilities p_{ij} that match the target mean degree and sectoral flow bands to the desired accuracy.

Parameters values for the complete US production network

We estimated parameter values of the US economy with approximately 5.4×10^6 firms and 1.25×10^8 connections between them. Table 3 presents the parameter values. Note that there are 173 values of $\{\lambda_{k\ell}\}$: one pertaining to each sector pair with a non-zero flow[27]. Figure 1 and Figure 2 present information on the 173 non-zero values of $\{\lambda_{k\ell}\}$ estimated by our procedure. Figure 1 plots the 173 values of $\{\lambda_{k\ell}\}$ with the y-axis being on log scale. Figure 2 presents a heatmap of all the values of $\{\lambda_{k\ell}\}$ with 24 sectors at the 2-digit SIC level on both axis.

Parameter	Symbol	Value
Global scale	z	0.30
Size elasticity	α	0.44
Sector score exponent	κ	0.32
Sector multipliers (range)	λ	10^{-2} to 2

Table 3: Estimated gravity parameters.

[26] Forming and storing the exact Hessian is prohibitively expensive in large problems, so IPOPT supports limited–memory quasi–Newton updates such as L–BFGS. L–BFGS builds a compact approximation of the Hessian (or its inverse) from a small number of recent gradient differences, using very little memory while preserving the curvature needed for efficient steps. Each interior–point iteration then requires solving a large, sparse Karush–Kuhn–Tucker linear system; MA97 is a sparse direct solver tailored to symmetric indefinite systems of this kind, exploiting sparsity and fill–reducing orderings to keep both memory usage and factorization time manageable.

[27] These parameter values were estimated by running the problem on a Mac Studio with M1 processor and 128 GB RAM. The machine ran for about 48 hours to generate the parameter values.

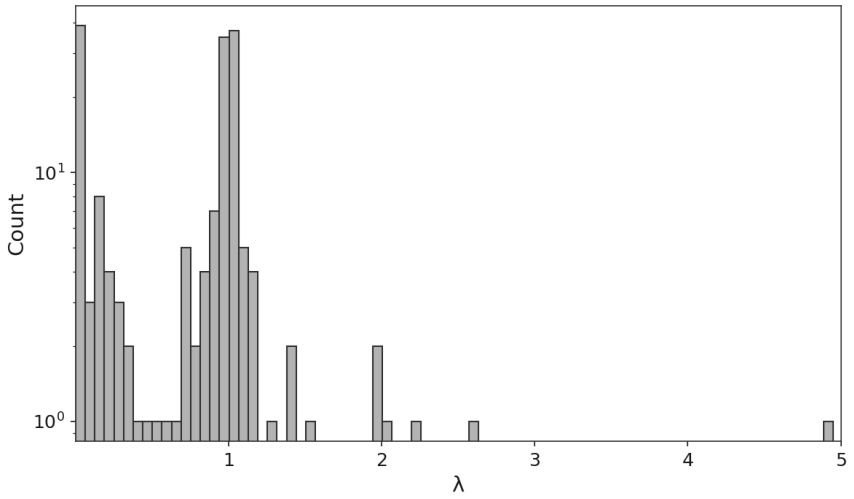

Figure 1: Sectoral multiplier parameters $\{\lambda_{l\ell}\}$

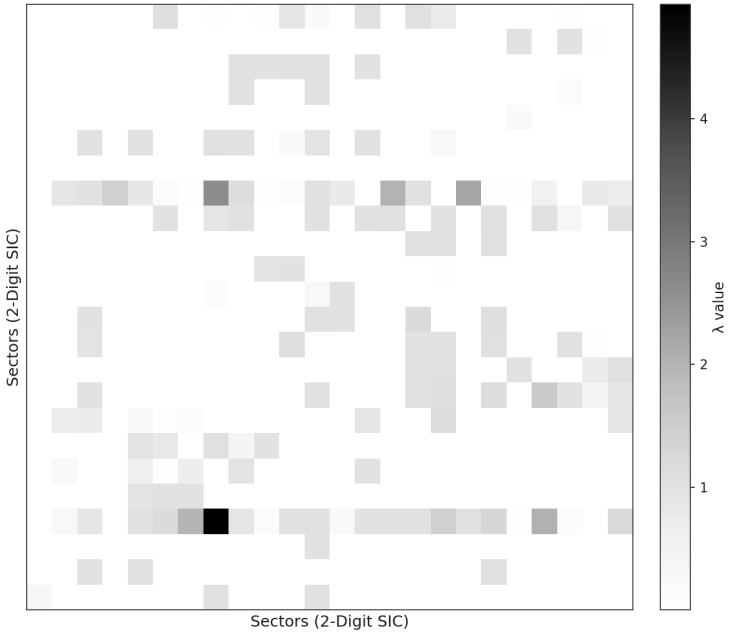

Figure 2: Heat-map of the sectoral multiplier parameter $\{\lambda_{l\ell}\}$ with 24 sectors at the 2-digit SIC level

Bootstrapped empirical distribution of the parameter values

We use a simple bootstrap design to gauge how much sampling variability there is in our gravity parameters. Concretely, we draw 100 bootstrap samples, each consisting of 10^5 firms and 2.5×10^6 firm–firm links, and re-estimate the gravity model on each sample. Each run delivers one draw of the size elasticity parameter α, the sector score exponent κ, and the global scaling

parameter z, together with a set of sector multipliers $\{\lambda_{k\ell}\}$.

Figures 3, 4, and 5 show the empirical distributions of α, κ, and z across these 100 bootstrap replications. The histograms are reasonably tight: each parameter is concentrated tightly around its mean (note the x-values). Figure 6 reports, for each of the 173 sectors, the standard deviation of the multiplier $\{\lambda_{l\ell}\}$ across the same 100 runs. These standard deviations are small as well, indicating that the sector multipliers are themselves tightly concentrated around their sectoral means. Taken together, the bootstrap evidence suggests that our gravity parameters are statistically stable and not unduly sensitive to the particular draw of firms and links used for estimation.

Once the gravity parameters are estimated, we form a binary backbone by independent Bernoulli draws with probabilities p_{ij}. To exploit sparse structure, the sampling is implemented in two passes over sector blocks: for each sector pair (k, ℓ) with $I_{k\ell} > 0$ we generate links between firms i in k and j in ℓ according to p_{ij}, and record the resulting directed edge list E.

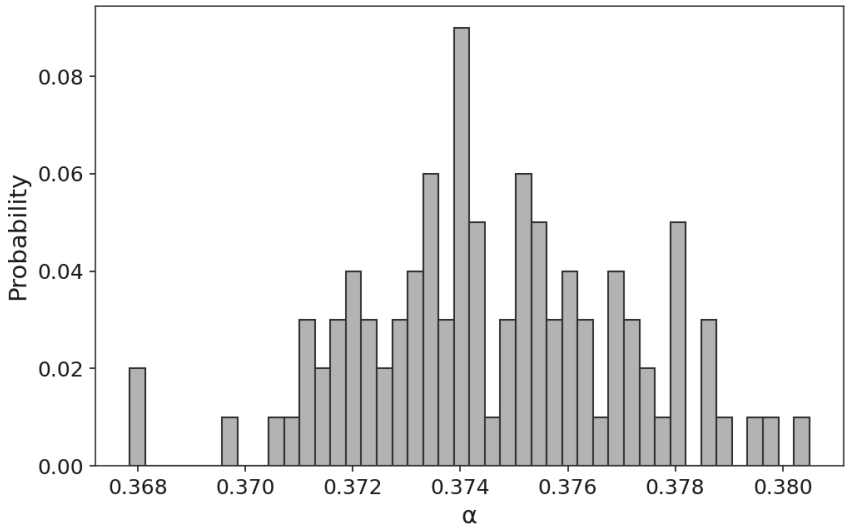

Figure 3: Distribution of size elasticity paramerter α

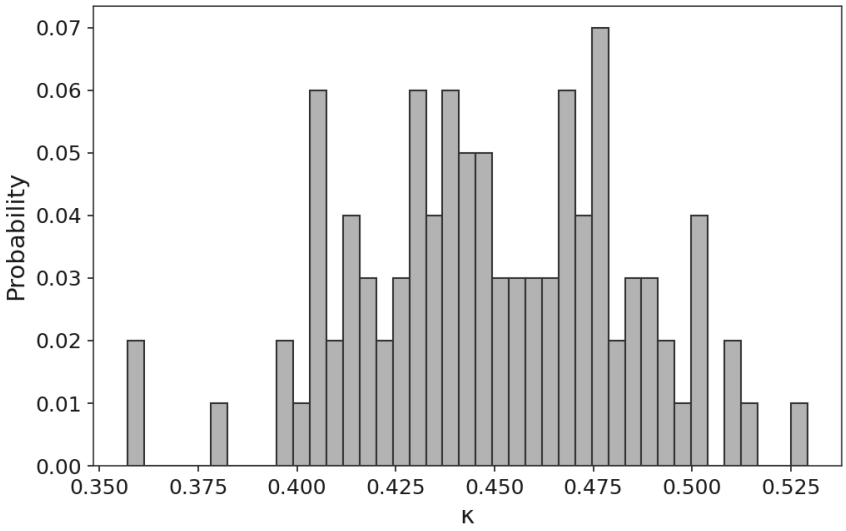

Figure 4: Distribution of sector score exponent parameter κ

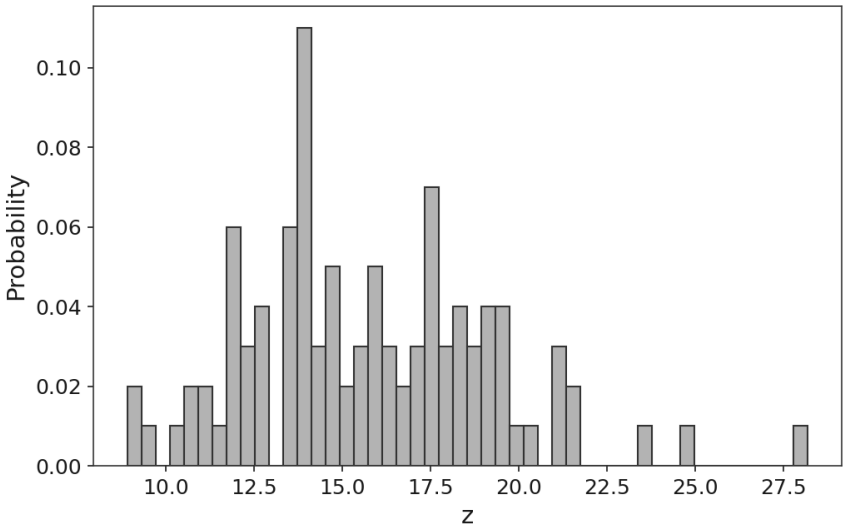

Figure 5: Distribution of global scaling parameter z

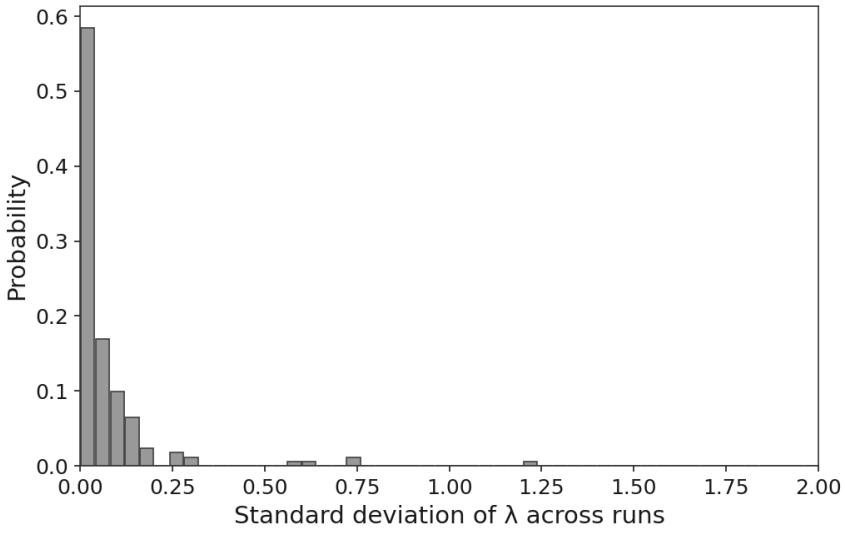

Figure 6: Distribution of the standard deviation of sector multiplier parameters

5.2.3 Tarjan's algorithm and Markov closure

We convert the Bernoulli backbone into a well-behaved Markov chain by enforcing irreducibility and aperiodicity. As discussed in Section 3, irreducibility is achieved by first decomposing the graph into strongly connected components (SCCs) using Tarjan's algorithm, and then adding a small number of cross-component links. We did not rely on specialized graph libraries for this step. Instead, we implemented Tarjan's algorithm directly to compute the SCCs and their condensation digraph, and we build the sink-source pairing and thinned candidate sets of inter-SCC edges as described in Section 3. The resulting Markov-closure problem is encoded as a 0–1 quadratic program with one binary variable per candidate edge, and solved with the mixed-integer quadratic programming (MIQP) module of CPLEX. Once the cross-SCC links are chosen, we add self-loops on the diagonal to eliminate periodicity, yielding the irreducible, aperiodic backbone on which the minimum-energy weighting step operates.

5.2.4 Using CPlex to compute the weights of the production network

Finally, we assign weights to the edges of the unweighted network by solving the minimum–energy quadratic program introduced in Section 3.4. We solve this convex QP with CPLEX, a general–purpose solver for linear and quadratic optimization problems. In our specification the quadratic objective is strictly convex and all constraints are linear, so any point satisfying the KKT conditions is a global optimum. CPLEX is designed to exploit exactly this structure and the sparsity of the constraint matrix, making it well suited to the large, sparse models that arise when

the network has millions of firms and edges.[28]

The QP includes bands on firm-level and sector-level seller totals, and caps on the mean and second moment of self-loop weights. These parameters act as error tolerances: they allow reconstructed flows to deviate modestly from their targets while preventing the solution from concentrating too much mass on self-loops. Table 4 reports the values used in our baseline US application.

Parameter description	Symbol	Value
Firm-level seller band	δ	0.10
Sector-level seller band	ε	0.10
First moment self-loop cap	η_1	0.10
Second-moment self-loop cap	η_2	0.10

Table 4: Parameter values used in the minimum–energy weighting step.

5.3 The Reconstructed US Production Network

We now report details of the reconstructed production network of the near-universe of firms in the US economy. This network has more than 5.4×10^6 firms and almost 1.3×10^8 directed links between them. To benchmark its structure, we compute standard directed-network statistics and compare them to a sample of the original US firm-to-firm network. The real-network sample (78,020 nodes and 172,636 directed edges) is based on S&P Capital IQ data presented in Mandel and Veetil (2025). Table 5 presents the summary statistics of the reconstructed network and the sample of the real world US network. Both networks are extremely sparse: densities are of order 10^{-5}, with the reconstructed network somewhat sparser than the S&P sample. Reciprocity and clustering are also lower in the reconstructed network, but remain small in absolute terms in both cases, far from a highly clustered or highly reciprocal structure. Degree assortativity coefficients are modest in magnitude throughout. The (in, in) and (out, out) coefficients are negative in both networks, indicating a mild tendency for high-degree firms to connect to lower-degree firms, with similar orders of magnitude. The cross (in, out) and (out, in) terms differ somewhat between the reconstructed and original sample, but all values are small in absolute terms. Overall, the reconstructed network reproduces the key qualitative features of the observed US firm network:

[28] For convex QPs, CPLEX typically uses a barrier or primal–dual interior–point algorithm, often combined with simplex–based refinement. In each barrier iteration it forms and solves a sparse KKT system that encodes the first–order optimality conditions. These systems are factorized using sparse linear algebra (fill–reducing orderings, sparse Cholesky or $\text{LDL}^\top$ factorizations), allowing the solver to handle models with hundreds of thousands or millions of variables. The positive semidefiniteness of the Hessian guarantees that the algorithm converges to the unique global optimum of the QP.

very sparse, low reciprocity and clustering, and weak degree mixing. Figure 7 presents the empirical counter-CDF of the degree distribution of the reconstructed US production network: both axis are in log-scale. Figure 7 shows that the degree distribution has fat tails, which can possibly be fit by a powerlaw. The steep descent in the tail of the degree distribution between 10^3 and 10^4 connections is a consequence of the fact that we have had to set an arbitrary endpoint in the largest size-bin for the data on the distribution of firm size by sector. We set the largest size to USD 1 billion, making the largest size bin range from 250 million to 1 billion, then sampled firm sizes treating this bin as a uniform distribution (exactly like we treat all other bins). The degree distribution would be different had we set a larger endpoint for the bin and had imposed a non-uniform distribution on the largest bin. This, however, would have imposed too much structure on the data. What is interesting is that even without imposing such a structure, if we limit our attention to the firms with 10^1 to somewhat more than 10^3 connections, a straight line appears not to be a bad fit in the log-log plot, suggesting a powerlaw degree distribution.

Table 5: Summary statistics: reconstructed network vs. original US sample

Statistic	**Reconstructed network**	**Original US sample**
Density	2.4×10^{-5}	6×10^{-5}
Reciprocity	0.00	0.03
Average clustering	0.00	0.01
Degree assortativity (in, in)	-0.02	-0.02
Degree assortativity (in, out)	0.02	0.00
Degree assortativity (out, in)	0.02	-0.09
Degree assortativity (out, out)	-0.03	-0.05

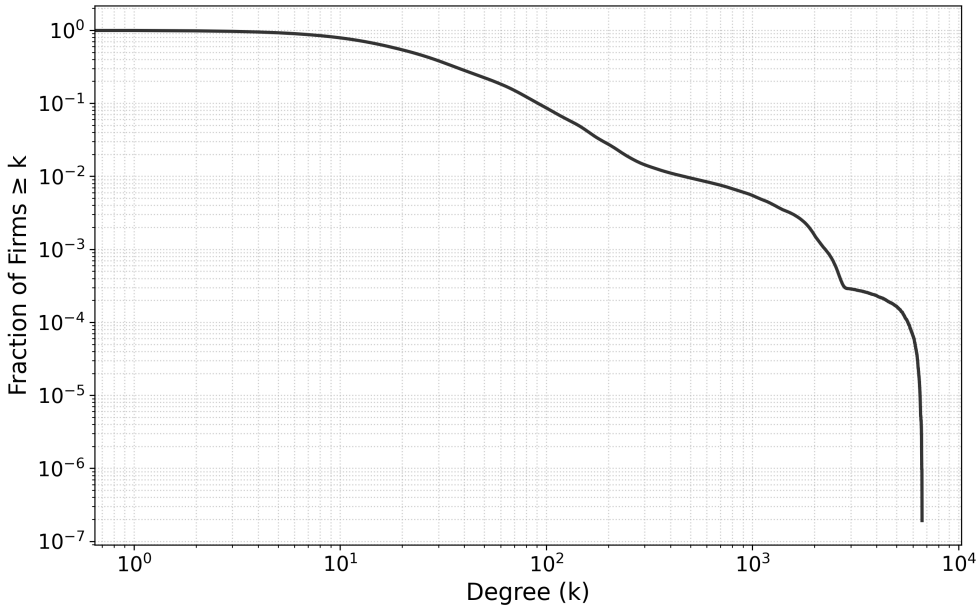

Figure 7: Counter CDF of the degree distribution of reconstructed US production network.

6 Extension: Network of Firms with Multiple Production Units (Factories)

In some applications—most notably those involving climate or natural-disaster shocks—we care not only about which firms trade with each other, but also about where production actually takes place. Weather shocks are tied to latitude and longitude, and many large firms operate multiple plants scattered across space. In such settings a purely firm-level network is too coarse: we need to extend the reconstructed production network into a network of factories (production units), while preserving the firm-level structure. To implement this extension, we need data on the geographic coordinates of all factories. Suppose each firm i owns a finite set of factories

$$\mathcal{A}(i) \;=\; \{a_{i,1}, \ldots, a_{i,k_i}\}, \qquad k_i \geq 1$$

and each factory $a \in \mathcal{A}(i)$ is located at latitude–longitude coordinates (φ_a, θ_a) in radians. We start from the firm-level weighted network estimated using the procedure in Section 3. This delivers a list of weighted directed firm edges $\{(i \to j)\}$ together with firm-level in-degree and out-degrees d_i^{O} and d_i^{I}, defined as before. The extension keeps these firm-level quantities intact: we do not change the number or total weight of links between any two firms. Instead, we decide how to allocate each firm-level edge $(i \to j)$ across the factories in $\mathcal{A}(i)$ and $\mathcal{A}(j)$, thereby

embedding the network in geographic space.

Distance-based factory weights and the connection matrix

Distances between factories are measured using the standard great-circle (Haversine) formula:

$$d(a,b) \;=\; 2R\,\arcsin\sqrt{\sin^2\!\Big(\tfrac{\varphi_a-\varphi_b}{2}\Big)+\cos\varphi_a\cos\varphi_b\,\sin^2\!\Big(\tfrac{\theta_a-\theta_b}{2}\Big)}\,, \qquad R\simeq 6371\ \text{km}$$

which converts latitude and longitude into kilometers. We translate distance into a simple weight via an exponentially decaying kernel

$$G(d;\tau) \;:=\; \exp(-d/\tau)\in(0,1], \qquad \tau>0$$

Closer factories receive larger weights. The single parameter τ controls how strongly distance affects factory-level assignment: a small τ makes distance matter a lot (links become very local).

We collect all factories of all firms into a single index set and define a factory–factory matrix $\mathcal{Q}$ with one row and one column per factory. Entry $\mathcal{Q}_{ab}$ is an unnormalized measure of how likely factory a is to send a link to factory b. We set

$$\mathcal{Q}_{ab} \;=\; \begin{cases} G\big(d(a,b);\tau\big), & \text{if factories } a \text{ and } b \text{ belong to different firms that are linked at the firm level} \\ 0, & \text{if } a \text{ and } b \text{ belong to the same firm, or the corresponding firms have no link} \end{cases}$$

We then normalize each row so that $\sum_b \mathcal{Q}_{ab}=1$ whenever that row has at least one positive entry. Thus each row of $\mathcal{Q}$ yields a probability distribution over the potential destinations of a given source factory.

The matrix $\mathcal{Q}$ is *dynamic*: after we assign a factory–factory connection that uses up the last remaining link between a pair of firms (i,j), we set to zero all entries $\mathcal{Q}_{ab}$ with $a\in\mathcal{A}(i)$ and $b\in\mathcal{A}(j)$, and then renormalize the affected rows. One can think of this as working with a sequence of matrices $\mathcal{Q}^{(0)},\mathcal{Q}^{(1)},\dots$ updated after each assignment (for notational simplicity we write $\mathcal{Q}$ and leave this dependence implicit).

Factory prominence within firms

In addition to $\mathcal{Q}$, we construct for each firm i a vector of prominence weights over its factories. Intuitively, a factory should be more prominent if it is close to many other factories in the network.

For each factory $a \in \mathcal{A}(i)$ we define a score

$$L_i(a) \;:=\; \sum_{j \neq i} \; \sum_{b \in \mathcal{A}(j)} G\Big(d(a,b); \tau\Big)$$

and then normalize within the firm so that

$$\psi_i(a) \;:=\; \frac{L_i(a)}{\sum_{a' \in \mathcal{A}(i)} L_i(a')}$$

is a probability distribution over factories $a \in \mathcal{A}(i)$. The vector p_i is fixed throughout the procedure; it captures how well-connected a factory of firm i is to the broader production geography.

Step 1: Activate all factories

We first give each factory a chance to be active. For each firm i with k_i factories and d_i^{O} outgoing firm-level links:

- If $d_i^{\mathrm{O}} \geq k_i$, we can assign at least one outgoing link to every factory. We loop once over factories $a \in \mathcal{A}(i)$. For each a, we draw a destination factory b using the current row $\mathcal{Q}_{a\cdot}$ and assign one outgoing link from a to b. This corresponds to assigning the full weight of one of firm i's remaining links to the firm of b. After each assignment we update $\mathcal{Q}$ to reflect the reduced remaining capacity between the corresponding firms and renormalize the affected rows.
- If $d_i^{\mathrm{O}} < k_i$, there are more factories than outgoing links. We still loop once over factories $a \in \mathcal{A}(i)$, drawing destinations b using the row $\mathcal{Q}_{a\cdot}$ as before. Some firm-level links from i to j will now be shared by multiple factories. We divide the weight of the underlying firm-level link evenly across the factory edges that use it, so that the total weight from i to j is preserved.

At the end of this step, every factory of every firm has at least one outgoing connection (unless its firm has zero outgoing links), and the remaining firm-level capacity between each pair of firms (i, j) is reduced appropriately. Let r_i denote the number of outgoing firm-level links of i that are still unassigned after Step 1.

Step 2: Allocate remaining links using prominence and distance

In the second step we allocate the remaining r_i links of each firm i using both the prominence weights ψ_i and the connection matrix $\mathcal{Q}$. For each unassigned outgoing link of firm i:

1. We first pick a source factory $a \in \mathcal{A}(i)$ according to the prominence distribution $\psi_i(\cdot)$. This makes factories that sit in dense production regions (close to many other factories) more likely to carry additional links.

2. Given the chosen source a, we then select a destination factory b according to the current row $\mathcal{Q}_{a\cdot}$, which reflects both distance and the remaining firm-level capacities.

3. We assign the link from i to j (the firm of b) to the factory edge $(a \to b)$, updating $\mathcal{Q}$ whenever the remaining capacity between the relevant firms is exhausted, and proceed to the next link.

Preservation of firm degrees and the spatial structure

By construction, when we sum over factories we recover the original firm-level network. Let d_a^{O} and d_b^{I} denote the out-degree and in-degree of factories in the extended network. This preservation is precisely due to how we construct and update $\mathcal{Q}$: initially, $\mathcal{Q}_{ab} > 0$ only if the corresponding firms have a link at the firm level, so no factory edge can appear between firms that are not connected. Whenever the remaining capacity of a firm pair (i, j) is exhausted, we set all entries $\mathcal{Q}_{ab}$ with $a \in \mathcal{A}(i)$ and $b \in \mathcal{A}(j)$ to zero and renormalize. As a result, we can never assign more factory–factory links (or total weight) between i and j than the original firm-level link allows.

The sum of factory-level weights from i to j equals the original firm-level weight from i to j, including the case $d_i^{\mathrm{O}} < k_i$ where some links are deliberately split, since weights are divided evenly across the corresponding factory edges by construction. Because each factory inherits its firm's sector label, sectoral totals computed from the firm-level network are also preserved.

What changes is the intra-firm-pair structure. Within a fixed firm pair (i, j), the exact pairing of factories is governed by the distance-based kernel $G(d; \tau)$ and the prominence weights ψ_i, so that nearby factories carry more of the firm-to-firm links on average, and factories located in dense production regions carry more links overall. The parameter τ provides a simple tuning knob for how local or dispersed this extension is: if the resulting factory network looks too local (links clustered very tightly in space), we can increase τ. If it looks too dispersed (links spread almost uniformly across geography), we can decrease τ. Note that this factory-level extension is a modular post-processing step. Once the firm-to-firm weighted network has been estimated using the procedure detailed in Section 3, the construction above can be applied directly to generate a geographically embedded network of factories that preserves all firm-level statistics while adding spatial detail[29].

[29]We do not simply treat each factory as a separate "firm" and re-run the reconstruction procedure (augmented with distance) for two reasons. First, such a factory-level gravity step would generically create many links between factories belonging to the *same* firm: conditional on sector, size (and distance, if included), a factory of firm i would be

7 Concluding Remarks

Theoretical and empirical investigations of the first quarter of the twenty-first century tell us that production networks play an important role in the transmission and amplification of a wide variety of shocks, including those that emanate from natural disasters, political events, monetary policy, and even ordinary business decisions (European Commission, 2023). In many of these areas 'scale' matters, i.e., the dynamics that emerge from a large-scale network can be very different from those that emerge from its smaller samples or subsets. Put differently, we cannot go from micro to macro on networks without working our way through 'complexity' and 'emergence' (Weaver, 1948; Anderson, 1972). Naturally, a careful study of such problems requires granular production network data at a sufficiently large scale. Unfortunately, however, such data are rare. The absence of such datasets has become a major bottleneck for frontier research on some of the most pressing problems of our era. Consider the problem of the propagation of climate change shocks. Shifts in average temperatures and the frequency of extremes will impact production in different places at different times. The aggregate impact on the world economy and human wellbeing will not be the simple sum of local damages as these local shocks are amplified by the production network. If floods inundate coal mines in Australia, the direct loss is coal. The indirect losses propagate through coking plants, steel mills, cement kilns, power stations, and rail depots, and then further into shipyards, construction firms, automotive and appliance manufacturers, engineering services, and even data centres and households that depend on reliable electricity and steel-intensive infrastructure. Naturally, many of these effects spill over national boundaries. To size and time these effects, we need simulations on the world production network at true scale, with millions of firms and billions of connections. Our reconstruction algorithm makes such simulations feasible in principle: using sectoral-flow data from the World Input–Output Table (WIOT), together with sector-wise firm-size distributions for the countries covered by WIOT, one can synthesize a global firm-to-firm production network at this scale.

Perhaps one day we will have truly granular global production network data. The push has already begun with multi-institution efforts to map firm-to-firm supply chains at scale (Pichler

just as likely to connect to another factory of i as to a factory of any other observationally similar firm, even though intra-firm buyer-seller relations are not the object of interest here. Second, while firm-size distributions by sector are available for many countries, analogous information on factory sizes is essentially absent, so a from-scratch factory-level reconstruction would require strong additional assumptions about how each firm's size is split across its plants. In our extension we avoid both issues by keeping the firm-level network (and its weights) fixed and only allocating each firm-firm edge across factories using a distance-based rule. Any implicit assumptions about within-firm allocation thus enter only as a modular add-on and do not play a role in determining the original network weights or the stationary firm-size distribution.

et al., 2024, 2023)[30]. But what should theory and policy do till then? Our answer in this paper is simple and practical: build large-scale synthetic production networks and use them to run the simulations we cannot yet run on real data. In this paper we present an algorithm which does exactly this. More specifically, we develop an algorithm that reconstructs networks that are faithful to the granularity of the economy (firms, not sectors) and to the flows to which those grains aggregate (sectoral flows). Figure 8 presents a flowchart summary of the reconstruction algorithm. Our approach begins by constructing a matrix of binary link probabilities between all firms in the economy. We estimate this matrix by fitting a gravity model through a logistic link, subject to aggregate-flow constraints. Our probability matrix acknowledges the uncertainties inherent in reconstructing production networks from aggregate, partial, or noisy data used to reconstruct the network. Having estimated $\mathbf{P}$, we draw an ensemble of directed, unweighted production networks. We then impose a Markov closure so that each draw is irreducible and aperiodic. We render the network irreducible by decomposing it into strongly connected components and solving a targeted optimization to connect these components while minimizing deviations from sectoral flows. We ensure aperiodicity by adding self-loops. Note that Markov-regularity ensures a well-behaved impulse-response system. In the final step, we assign weights via a minimum-energy program that remains faithful to the unweighted backbone by preventing edge weights from being arbitrarily squashed toward zero. The program includes several constraints, one of which keeps the stationary firm-size distribution implied by the network close to the empirically observed size distribution. This matters for applications where the distribution of firm sizes influences system dynamics. We also present an extension of the basic model to create a production network when firms have multiple factories in specific geographical locations. Overall, our procedure allows for the construction of large-scale granular production networks using publicly available data.

We are not the first to attempt a network reconstruction, but we are the first to undertake it on such a large scale. In fact, we designed an algorithm that scales linearly in the number of firms when firm sizes are appropriately binned, which means that the procedure can be implemented within a reasonable clock time even for networks with millions of firms. To demonstrate feasibility, we reconstruct the production network of the entire US economy and show that it reproduces some salient large-network features observed in real-world network data. Most importantly, we share the code of our algorithm as an open-source library. By providing publicly available code, we hope to have made it possible for research teams around the world to pursue empirical and theoretical questions of their interest without having to bear the enormous cost of procuring

[30] In fact, input-output tables were constructed by many countries and international organizations in response to efforts of a similar sort initiated by Leontief himself (Leontief, 1951; Leontief and the Harvard Economic Research Project, 1953)

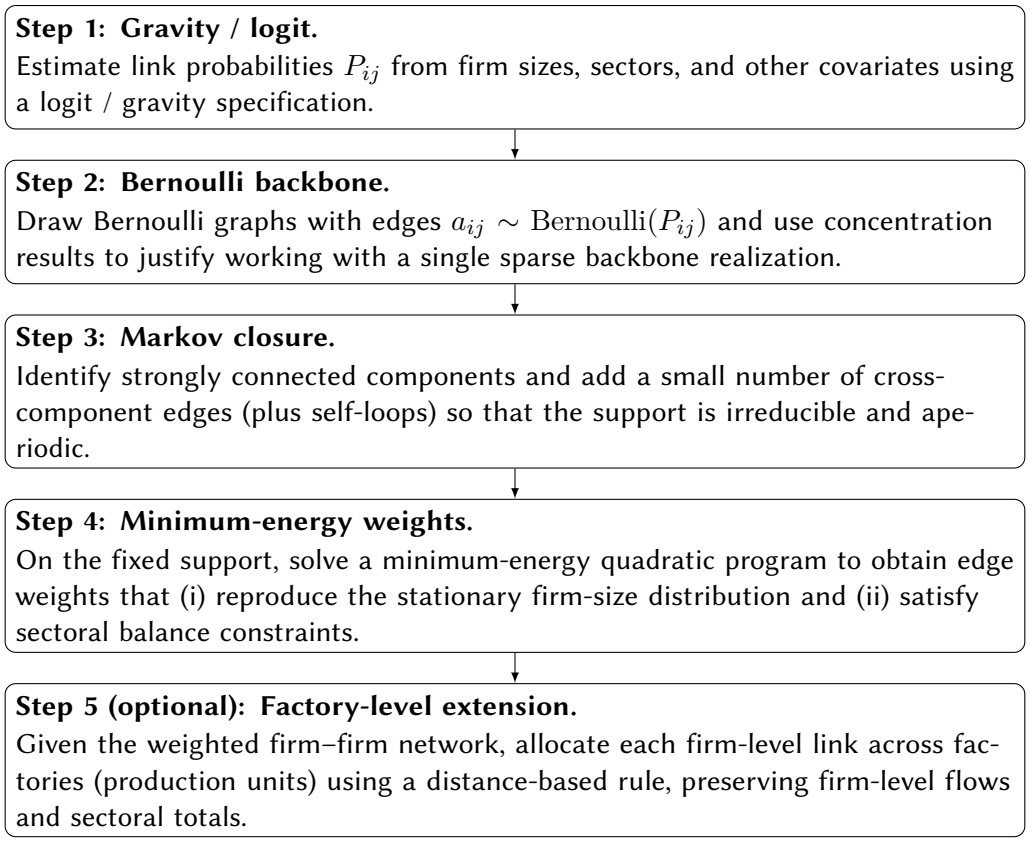

Figure 8: Schematic overview of the reconstruction algorithm. Steps 1–4 produce a weighted, irreducible firm–firm network. Step 5 is an optional extension that refines this network to the level of individual factories.

real-world network data.

A Mathematical Appendix

Remark 5 (Finite-sample error bounds for a randomly drawn adjacency matrix). In Remark 1 we showed, via a CLT, that as $N_F \to \infty$ the total number of edges $\mathcal{E}_u$ of a Bernoulli draw u concentrates around its model-implied counterpart $\mu_{\mathcal{E}}$. Here we provide finite-sample deviation bounds for the degrees of individual firms (bounds for $\mathcal{E}_u$ follow immediately by summing over firms). Assume independent draws $A_{ij,u} \sim \text{Bernoulli}(p_{ij})$ for $i \neq j$ with $A_{ii,u} = 0$, and let $\mathbb{E}$ and $\mathbb{P}$ denote expectation and probability under this Bernoulli product measure. For each firm i, define

$$\mu_i := \sum_{j \neq i} p_{ij} \qquad \sigma_i^2 := \sum_{j \neq i} p_{ij}(1 - p_{ij})$$

the model-implied mean and variance of its (out-)degree,[31] and write

$$d_{i,u} := \sum_{j \neq i} A_{ij,u}$$

for the realized degree of firm i in draw u. By construction,

$$\mathbb{E}[d_{i,u}] = \mu_i, \qquad \text{Var}(d_{i,u}) = \sigma_i^2$$

because the edge indicators $\{A_{ij,u}\}_{j \neq i}$ are independent.

Bernstein bounds for firm degrees

For a firm i, define $X_{ij,u} := A_{ij,u} - p_{ij}$ so that

$$d_{i,u} - \mu_i = \sum_{j \neq i} (A_{ij,u} - p_{ij}) = \sum_{j \neq i} X_{ij,u}$$

By construction: (i) the variables $\{X_{ij,u}\}_{j \neq i}$ are independent (because the edges $\{A_{ij,u}\}_{j \neq i}$ are independent under the Bernoulli product measure), (ii) they are mean-zero, $\mathbb{E}[X_{ij,u}] = 0$, (iii) they are uniformly bounded, $|X_{ij,u}| \leq 1$ since $A_{ij,u} \in \{0, 1\}$, and (iv) their total variance is

$$\sum_{j \neq i} \text{Var}(X_{ij,u}) = \sum_{j \neq i} p_{ij}(1 - p_{ij}) = \sigma_i^2$$

These are precisely the conditions under which Bernstein's inequality applies to $d_{i,u} - \mu_i$. Applying Bernstein's inequality for sums of independent, bounded random variables (with variance proxy

[31] For concreteness we state all results for out-degrees; the corresponding statements for in-degrees follow by transposing $\mathbf{P}$ and are entirely analogous.

σ_i^2 and bound $|X_{ij,u}| \le 1$) therefore gives, for every $t > 0$,

$$\mathbb{P}\Big(\Big|d_{i,u} - \mu_i\Big| \ge t\Big) \le 2\exp\left(-\frac{t^2}{2\sigma_i^2 + \frac{2}{3}t}\right)$$

Equivalently, for any $\delta \in (0,1)$,

$$\mathbb{P}\left(\Big|d_{i,u} - \mu_i\Big| \le \sqrt{2\sigma_i^2 \log\tfrac{2}{\delta}} + \tfrac{2}{3}\log\tfrac{2}{\delta}\right) \ge 1 - \delta$$

By a union bound over $i = 1, \dots, N_F$, with probability at least $1 - \delta$ the degree bound holds simultaneously for all firms after replacing δ by δ/N_F on the right-hand side. The bounds are sharp (up to logarithmic factors) when expected degrees are bounded and no single dyad carries a large probability mass. These tightness conditions correspond to economically plausible structures in which buyers distribute small procurement probabilities across many potential suppliers, sectoral structure guides but does not concentrate probabilities on a few dyads, and link formation is granular and idiosyncratic conditional on $\mathbf{P}$. In short, even for finite but large N_F, a typical sample A_u has firm-level degrees that are close to the model-implied degrees encoded in $\mathbf{P}$.

Degree distributions

The same Bernstein machinery yields finite-sample control for the empirical degree distribution. Fix thresholds $\tau_1, \dots, \tau_L$. For each ℓ define

$$\hat{F}_{\ell,u} := \frac{1}{N_F}\sum_{i=1}^{N_F} \mathbf{1}_{\{d_{i,u} \le \tau_\ell\}} \qquad F_\ell := \frac{1}{N_F}\sum_{i=1}^{N_F} \mathbb{P}\Big(d_i \le \tau_\ell\Big)$$

as the empirical and model-implied cumulative degree fractions at τ_ℓ respectively. Here d_i denotes the random out-degree of firm i under the Bernoulli model (i.e. before realizing a particular draw u), while $d_{i,u}$ is its realized value in draw u. Since degrees across firms depend on disjoint sets of edges, the indicators $\mathbf{1}_{\{d_{i,u} \le \tau_\ell\}}$ are independent across i and bounded in $[0,1]$. Applying Bernstein to $\hat{F}_{\ell,u} - F_\ell$ and then a union bound over $\ell = 1, \dots, L$[32] yields:

$$\max_{1 \le \ell \le L} \Big|\hat{F}_{\ell,u} - F_\ell\Big| = O\Big(\sqrt{\tfrac{\log(L/\delta)}{N_F}}\Big) \quad \text{with probability at least } 1 - \delta$$

[32]For each fixed threshold τ_ℓ, Bernstein's inequality yields a deviation bound of order $\sqrt{\log(1/\delta)/N_F}$. Requiring the bound to hold simultaneously for all L thresholds introduces an additional $\log L$ factor via the union bound, leading to the rate $\sqrt{\log(L/\delta)/N_F}$.

Thus the empirical degree distribution in draw u uniformly tracks its model-implied counterpart at the usual parametric rate, up to a $\sqrt{\log L}$ factor for L evaluation points[33]. $\triangle$

Remark 6 (Minimum-energy as a special case of maximum-entropy). In Section 3.4 we assign edge weights $\mathbf{W} = [w_{ij}]$ on the fixed support $\widetilde{\mathbf{A}}$ by solving the convex program

$$\widehat{\mathbf{W}} \;\in\; \arg\min_{\mathbf{W}} \sum_{i=1}^{N_F}\sum_{j=1}^{N_F} w_{ij}^2 \quad \text{s.t.} \quad \mathbf{W} \in \mathcal{C}$$

where $\mathcal{C}$ is the set of matrices satisfying the row-stochasticity, support and bound conditions, firm- and sector-level consistency constraints, and self-weight moment bounds listed in Section 3.4. Identifying $a = \mathrm{vec}(\mathbf{W}) \in \mathbb{R}^{|\mathcal{E}|}$, the objective becomes

$$\sum_{i,j} w_{ij}^2 = \|\mathbf{W}\|_F^2 = \|a\|_2^2$$

so our estimator is the minimum-energy solution

$$\hat{a} \;=\; \arg\min_{a\in\mathcal{C}} \|a\|_2^2$$

We now show that this minimum-energy rule is a special case of a maximum-entropy (MaxEnt) construction. Let $\mathcal{C} \subset \mathbb{R}^{|\mathcal{E}|}$ denote the feasible set for $a = \mathrm{vec}(\mathbf{W})$ induced by the constraints above. Consider probability densities f supported on $\mathcal{C}$, and the MaxEnt problem

$$\max_f \left\{ -\int_{\mathcal{C}} f(a)\log f(a)\,da \right\} \quad \text{s.t.} \quad \int_{\mathcal{C}} f(a)\,da = 1,\;\; \mathbb{E}_f\big[\|a\|_2^2\big] = \tau$$

for some prescribed second moment $\tau > 0$. The Lagrangian with multipliers (λ, β) is

$$\mathcal{L}(f) = -\int_{\mathcal{C}} f\log f\,da + \lambda\Big(\int_{\mathcal{C}} f\,da - 1\Big) + \beta\Big(\int_{\mathcal{C}} \|a\|_2^2 f\,da - \tau\Big)$$

Formal first-order stationarity in f yields

$$\log f^\star(a) = \lambda' - \beta\|a\|_2^2 \quad \text{on } \mathcal{C} \qquad \Rightarrow \qquad f^\star(a) \;\propto\; \exp\Big(-\beta\,\|a\|_2^2\Big)\,\mathbf{1}\{a\in\mathcal{C}\},\; \beta > 0$$

with β chosen so that $\mathbb{E}_{f^\star}\|a\|_2^2 = \tau$. This is a Gibbs (truncated Gaussian) density supported on

[33]The same argument extends to bounded edge-averaged statistics by replacing the degree indicators $\mathbf{1}_{\{d_{i,u}\le\tau_\ell\}}$ with bounded functions of edges or node pairs.

the feasible set. Since $f^\star$ is log-concave on $\mathcal{C}$ and

$$\log f^\star(a) = -\beta\|a\|_2^2 + \text{const}$$

its mode is

$$a^{\text{mode}} \;=\; \arg\max_{a\in\mathcal{C}} f^\star(a) \;=\; \arg\min_{a\in\mathcal{C}} \|a\|_2^2$$

Thus the MaxEnt density over $\mathcal{C}$ with fixed second moment places its peak at exactly the minimum-energy point. In our case, $\mathcal{C}$ is the feasible set defined in Section 3.4, so $\hat{a}$ coincides with a^{mode}. Equivalently, the matrix $\hat{\mathbf{W}}$ we compute is the mode of the MaxEnt ensemble

$$f^\star(a) \;\propto\; \exp\Big(-\beta\|a\|_2^2\Big)\,\mathbf{1}\{a\in\mathcal{C}\}$$

Maximum entropy selects, among all edge-weight configurations satisfying the constraints in $\mathcal{C}$ (support, row-stochasticity, firm and sector consistency, and self-weight control) and a fixed quadratic moment $\mathbb{E}\|a\|_2^2$, the distribution that is as "spread out" as possible. Under this view, the peak of that MaxEnt distribution occurs at the feasible point with the smallest energy $\|a\|_2^2$. Our minimum-energy estimator simply chooses that point: the least structured network compatible with the information encoded in $\mathcal{C}$, without imposing any additional pattern in the weights. $\triangle$

Remark 7 (Error bound on the weighted matrix). Here we show that if an exactly money-preserving matrix exists on the observed support, then the minimum-energy solution of our approximate weighting problem lies close, in Frobenius norm, to such an exact matrix whenever the residual money and sector errors are small. Intuitively, loosening the constraints slightly does not push us far away from the minimum-energy solution that would satisfy all constraints exactly.

Let $\mathbf{m}$ denote the empirical money vector and $\boldsymbol{\mu} := \mathbf{m}/M$ its normalization, with $M = \mathbf{1}^\top\mathbf{m} > 0$. Assume there exists a row-stochastic, irreducible, aperiodic matrix $\widetilde{\mathbf{W}}$ on the support of $\widetilde{\mathbf{A}}$ such that

$$\widetilde{\mathbf{W}}^\top\mathbf{m} = \mathbf{m}$$

i.e., $\boldsymbol{\mu}$ is stationary for $\widetilde{\mathbf{W}}$. Define $\Delta := \mathbf{W} - \widetilde{\mathbf{W}}$. Then

$$\underbrace{\boldsymbol{\mu} - \mathbf{W}^\top\boldsymbol{\mu}}_{\mathbf{r}} = \underbrace{(\mathbf{I} - \mathbf{W}^\top)\boldsymbol{\mu}}_{\text{one-step difference}} = \widetilde{\mathbf{W}}^\top\boldsymbol{\mu} - \mathbf{W}^\top\boldsymbol{\mu} = -\Delta^\top\boldsymbol{\mu}$$

Thus the one-step residual $\mathbf{r}$ is exactly the $\boldsymbol{\mu}$-weighted column action of the perturbation Δ.

Stability of the minimum-energy solution

To quantify how large Δ can be, it is convenient to view the QP in vector form. Vectorize the weights on the fixed edge set $\mathcal{E}$ as $a = \mathrm{vec}(\mathbf{W})$. Let $\mathcal{C}_0$ collect the linear equalities and inequalities that do not involve squared terms (row sums, support, optional lower bounds), and let $\mathbf{C}a = \mathbf{d}$ encode the firm- and sector-level equalities (exact preservation of firm money balances and sector totals) that we only impose approximately in practice.

Consider the two convex quadratic programs:

$$\text{(Exact)} \quad \widetilde{a} = \arg\min_{a} \tfrac{1}{2}\|a\|_2^2 \quad \text{s.t.} \quad a \in \mathcal{C}_0, \;\; \mathbf{C}a = \mathbf{d}$$

$$\text{(Approx)} \quad a^\star = \arg\min_{a} \tfrac{1}{2}\|a\|_2^2 \quad \text{s.t.} \quad a \in \mathcal{C}_0, \;\; \|\mathbf{C}a - \mathbf{d}\|_2 \le \eta$$

Both problems are strongly convex (Hessian $= \mathbf{I}$), hence each has a unique minimizer. Suppose the exact problem is feasible and that $\mathbf{C}$ has full row rank on the relevant subspace. Then standard sensitivity results for strongly convex QPs imply the Lipschitz-type stability bound (Boyd and Vandenberghe, 2004, Ch. 4-5)

$$\|a^\star - \widetilde{a}\|_2 \;\le\; \|\mathbf{C}^+\|_2 \, \eta$$

where $\mathbf{C}^+$ is the Moore–Penrose pseudoinverse and $\|\mathbf{C}^+\|_2 = 1/\sigma_{\min}(\mathbf{C})$ on the constraint subspace. In our application the tolerance η is directly controlled by the L_2 constraints in the weighting program, so that

$$\eta \;\lesssim\; \Big\|\mathrm{diag}(\mathbf{m})^{-1}\big(\mathbf{W}^\top\mathbf{m} - \mathbf{m}\big)\Big\|_2 \;+\; \Big(\sum_{\ell=1}^{N_S}\Big((\widehat{s}_\ell(\mathbf{W}) - s_\ell)/s_\ell\Big)^2\Big)^{1/2}$$

Rewriting in matrix form gives

$$\|\mathbf{W} - \widetilde{\mathbf{W}}\|_F \;\le\; \kappa\left[\Big\|\mathrm{diag}(\mathbf{m})^{-1}\big(\mathbf{W}^\top\mathbf{m} - \mathbf{m}\big)\Big\|_2 \;+\; \Big(\sum_{\ell}\Big((\widehat{s}_\ell(\mathbf{W}) - s_\ell)/s_\ell\Big)^2\Big)^{1/2}\right]$$

for some constant κ that depends only on the conditioning of $\mathbf{C}$ (support pattern, row-sum equations, and how sector/firm constraints enter).

Consequence and significance

If the residual tolerances in the weighting program are tightened so that

$$\Big\|\mathrm{diag}(\mathbf{m})^{-1}(\mathbf{W}^\top\mathbf{m} - \mathbf{m})\Big\|_2 \to 0 \qquad \text{and} \qquad \sum_{\ell}\Big((\widehat{s}_\ell(\mathbf{W}) - s_\ell)/s_\ell\Big)^2 \to 0$$

and if the constraint operator remains well conditioned (i.e. $\|\mathbf{C}^{+}\|_2$ stays bounded), then

$$\|\mathbf{W} - \widetilde{\mathbf{W}}\|_F \;\to\; 0$$

In other words, among all matrices that exactly preserve firm and sector totals on the observed support, the minimum-energy solution of the approximate program converges to the minimum-energy solution of the exact program whenever such a matrix $\widetilde{\mathbf{W}}$ exists. This strengthens Remark 4: that remark shows that stationary money holdings are close. Here we show that the *entire* transition matrix $\mathbf{W}$ is close (in Frobenius norm) to an exactly money-preserving matrix. The most significant implication of this Remark is that small perturbations in $\mathbf{W}$ imply small perturbations in $\mathbf{W}^t$ for fixed horizons t. And therefore the dynamic response of the reconstructed network shocks is stable with respect to the small constraint violations we allow in the weighting step.

References

Anderson, P. W. 1972. "More Is Different: Broken Symmetry and the Nature of the Hierarchical Structure of Science." *Science*, 177(4047): 393–396.

Augusztinovics, Mária. 1965. "A Model of Money-Circulation." *Economics of Planning*, 5(3): 44–57. Institute of Economic Planning, Budapest.

Bacilieri, Andrea, and Pablo Austudillo-Estevez. 2023. "Reconstructing firm-level input–output networks from partial information." *arXiv preprint arXiv:2304.00081*.

Bacilieri, Andrea, András Borsos, Pablo Astudillo-Estevez, Mads Hoefer, and Francois Lafond. 2023. "Firm-Level Production Networks: What Do We (Really) Know?" INET Oxford INET Oxford Working Paper 2023-08. Also available as SSRN 5344255.

Baik, Jinho, Gérard Ben Arous, and Sandrine Péché. 2005. "Phase transition of the largest eigenvalue for nonnull complex sample covariance matrices." *Annals of Probability*, 33(5): 1643–1697.

Batagelj, Vladimir, and Ulrik Brandes. 2005. "Efficient generation of large random networks." *Physical Review E*, 71(3): 036113.

Boyd, Stephen, and Lieven Vandenberghe. 2004. *Convex Optimization.* Cambridge:Cambridge University Press.

Buiten, Gert, Edwin de Jonge, Gideon Mooijen, Sjoerd Hooijmaaijers, and Patrick Bogaart. 2021. "Reconstruction Method for the Dutch Interfirm Network Including a Breakdown by Commodity for 2018 and 2019 (v1.0)." Statistics Netherlands (CBS) Technical Report, The Hague. Version 1.0.

Cardoza, Marvin, Francesco Grigoli, Nicola Pierri, and Cian Ruane. 2025. "Worker mobility in production networks." *Review of Economic Studies*, 92(6): 3682–3703.

Carvalho, Vasco M, Makoto Nirei, Yukiko U Saito, and Alireza Tahbaz-Salehi. 2021. "Supply chain disruptions: Evidence from the great east japan earthquake." *The Quarterly Journal of Economics*, 136(2): 1255–1321.

Cimini, Giulio, Rossana Mastrandrea, and Tiziano Squartini. 2021. *Reconstructing Networks. Cambridge Elements: Structure and Dynamics of Complex Networks*, Cambridge University Press.

Dhyne, Emmanuel, Ayumu Ken Kikkawa, Magne Mogstad, and Felix Tintelnot. 2021. "Trade and Domestic Production Networks." *The Review of Economic Studies*, 88(2): 643–668.

Diem, Christian, András Borsos, Tobias Reisch, János Kertész, and Stefan Thurner. 2022. "Quantifying Firm-Level Economic Systemic Risk from Nation-Wide Supply Networks." *Scientific Reports*, 12: 7719.

Eswaran, Kapali P., and Robert E. Tarjan. 1976. "Augmentation problems." *SIAM Journal on Computing*, 5(4): 653–665.

European Commission. 2023. "Climate stress test of the global supply chain network: the case of river floods." Publications Office of the European Union.

Gabaix, Xavier. 2011. "The Granular Origins of Aggregate Fluctuations." *Econometrica*, 79(3): 733–772.

Goltsev, Alexander V., Sergey N. Dorogovtsev, J. G. Oliveira, and José F. F. Mendes. 2012. "Localization of eigenvectors of the adjacency matrix of complex networks." *Physical Review Letters*, 109(9): 098701.

Hooijmaaijers, Sjoerd, and Gert Buiten. 2019. "A Methodology for Estimating the Dutch Interfirm Trade Network, Including a Breakdown by Commodity." Statistics Netherlands (CBS) Technical Report, The Hague. Method paper; presented at OECD Conference "New Analytical Tools and Techniques for Economic Policy-making".

Huneeus, Federico, Kory Kroft, and Kevin Lim. 2021. "Earnings Inequality in Production Networks." National Bureau of Economic Research NBER Working Paper 28424.

Ialongo, Leonardo Niccolʻo, Camille de Valk, Emiliano Marchese, Fabian Jansen, Hicham Zmarrou, Tiziano Squartini, and Diego Garlaschelli. 2022. "Reconstructing firm-level interactions: the Dutch input–output network." *Scientific Reports*, 12(11847).

Ialongo, Leonardo Niccolʻo, Sylvain Bangma, Fabian Jansen, and Diego Garlaschelli. 2024. "Multi-scale reconstruction of large supply networks." *arXiv preprint arXiv:2412.16122*.

Langville, Amy N., and Carl D. Meyer. 2006. *Google's PageRank and Beyond: The Science of Search Engine Rankings.* Princeton, NJ:Princeton University Press.

Leontief, Wassily. 1951. *The Structure of the American Economy, 1919–1939: An Empirical Application of Equilibrium Analysis.* Oxford University Press.

Leontief, Wassily, and András Bródy. 1993. "Money-Flow Computations." *Economic Systems Research*, 5(3): 225–233.

Leontief, Wassily, and the Harvard Economic Research Project. 1953. *Studies in the Structure of the American Economy: Theoretical and Empirical Explorations in Input–Output Analysis.* Oxford University Press.

Mandel, Antoine, and Vipin P Veetil. 2021. "Monetary dynamics in a network economy." *Journal of Economic Dynamics and Control*, 125: 104084.

Mandel, Antoine, and Vipin P Veetil. 2025. "Do granular shocks generate sizeable aggregate volatility?" *Journal of Evolutionary Economics*, 35(1): 71–94.

Mastrandrea, Rossana, Tiziano Squartini, Giorgio Fagiolo, and Diego Garlaschelli. 2014. "Enhanced reconstruction of weighted networks from strengths and degrees." *New Journal of Physics*, 16(4): 043022.

Mizuno, Takayuki, Wataru Souma, and Tsutomu Watanabe. 2014. "The Structure and Evolution of Buyer-Supplier Networks." *PLOS ONE*, 9(7): e100712.

Nesterov, Yurii. 2004. *Introductory Lectures on Convex Optimization: A Basic Course.* Springer.

Palepu, Sai, Brian J. Clark, Bill B. Francis, and Raffi E. García. 2025. "Cascading Effects of Climate Change in Production Networks: Evidence from Extreme Weather Events." *SSRN Working Paper*.

Panigrahi, Piyush. 2021. "Endogenous spatial production networks: Quantitative implications for trade and productivity." CESifo Working Paper.

Pichler, Anton, Christian Diem, Alexandra Brintrup, François Lafond, Glenn Magerman, Gert Buiten, Thomas Choi, Vasco M. Carvalho, J. Doyne Farmer, and Stefan Thurner. 2023. "Building an alliance to map global supply networks." *INET Oxford publication*.

Pichler, Anton, Christian Diem, Alexandra Brintrup, François Lafond, Glenn Magerman, et al. 2024. "The need for a better map of the global supply network." SUERF Policy Brief No. 803.

Rabenseifner, Rolf. 2004. "Optimization of Collective Reduction Operations." Vol. 3036 of *Lecture Notes in Computer Science*, 1–9. Springer.

Squartini, Tiziano, Giulio Cimini, Andrea Gabrielli, and Diego Garlaschelli. 2017. "Network reconstruction via density sampling." *Applied Network Science*, 2: 3.

Tarjan, Robert E. 1972. "Depth-first search and linear graph algorithms." *SIAM Journal on Computing*, 1(2): 146–160.

Vece, Marzio Di, Diego Garlaschelli, and Tiziano Squartini. 2023. "Reconciling econometrics with continuous maximum entropy network models." *Chaos, Solitons & Fractals*, 166: 112958.

Weaver, Warren. 1948. "Science and Complexity." *American Scientist*, 36(4): 536–544.

Welburn, Jonathan W., Aaron M. Strong, Giovanni Malloy, Prateek Puri, James Syme, and Jessie Wang. 2023. "The Global Economy at the Firm-Level: Estimating Input–Output Linkages in Production Networks and the Potential for Systemic Risk." RAND Corporation Working Paper WR-A2625-1, Santa Monica, CA.