Supplementary Material: Information transmission and signal permutation in active flow networks

Francis G. Woodhouse,[1,*] Joanna B. Fawcett,[2,†] and Jörn Dunkel[3]

[1] *Department of Applied Mathematics and Theoretical Physics, Centre for Mathematical Sciences, University of Cambridge, Wilberforce Road, Cambridge CB3 0WA, U.K.*

[2] *Department of Pure Mathematics and Mathematical Statistics, Centre for Mathematical Sciences, University of Cambridge, Wilberforce Road, Cambridge CB3 0WB, U.K.*

[3] *Department of Mathematics, Massachusetts Institute of Technology, 77 Massachusetts Avenue, Cambridge MA 02139-4307, U.S.A.*

(Dated: January 15, 2018)

NUMERICAL METHODS

Incompressible sample space

To reduce dimensionality when evaluating $H(\Phi)$, we sample from a special incompressible space for Φ, as we now describe.

In a network without inputs or outputs, the space of integer-valued incompressible flows (that is, more general flows with $\phi_e \in \mathbb{Z}$) is spanned by a cycle basis $\mathcal{B}_1 = \{\boldsymbol{C}_a\}$, $1 \leqslant a \leqslant |E|-|V|+1$. This is a set of linearly independent cycles, each defined by its flow vector $\boldsymbol{C}_a = (C_{ae})$ on the edges of Γ. Lattices, and planar graphs in general, possess an intuitive form of such a basis comprising one cycle around each face except the external infinite face; Euler's formula $|E| - |V| + |F| = 2$ for finite connected planar graphs with $|F|$ faces then guarantees $\mathcal{B}$ has the correct dimension.

Now, suppose Γ has N inputs and N outputs. To allow for input–output flows, two further components are needed. The first is a set $\mathcal{I} = \{\boldsymbol{D}_i\}$, $1 \leqslant i \leqslant N$, where $\boldsymbol{D}_i = (D_{ie})$ is a flow from input i to output i. These are not basis elements as such, but instead will be added to the flow according to which inputs are activated. On its own, however, $\mathcal{I}$ does not allow an input to connect to an arbitrary output. The second component needed is a set $\mathcal{B}_2 = \{\boldsymbol{E}_j\}$, $1 \leqslant j \leqslant N-1$, where $\boldsymbol{E}_j = (E_{je})$ is a flow from output j to output $j+1$. The $\boldsymbol{E}_j$ are not valid flows in themselves as each disobeys the diode constraint of an output-adjacent edge. Rather, they work in tandem with $\mathcal{I}$ to allow flow from an input to exit from an arbitrary output. For example, $\boldsymbol{D}_1 + \boldsymbol{E}_{12}$ represents a valid flow from input 1 to output 2 if the vectors do not intersect other than at the edge to output 1; if they do, it can be made valid by adding cycles or subtracting cycles from $\mathcal{B}_1$ as necessary.

Given the vector of inputs I_i, we express candidate flows as $\Phi = \Psi + \sum_i I_i \boldsymbol{D}_i$, where Ψ is an integer-coefficient linear combination of vectors in $\mathcal{B} = \mathcal{B}_1 \cup \mathcal{B}_2$. This is still a superset of the exact configuration space of Φ, and flows need to be rejected if they have invalid components $\phi_e \notin \{-1, 0, +1\}$ for any e. Nevertheless, the vertex flux constraints are now always fulfilled and the sample space is in a form that is much more amenable to exhaustive and Monte Carlo sampling.

Flow space construction

In practice, the sets $\mathcal{B}_1, \mathcal{B}_2, \mathcal{I}$ were constructed using Mathematica. We used different algorithmic approaches depending on the type of graph considered.

Random graphs. For the general graphs in Fig. 4 of the main text, the cycle basis $\mathcal{B}_1$ was constructed by a basic implementation of Horton's minimal weight basis algorithm [1]: all cycles of length 3 are found and each progressively appended to $\mathcal{B}_1$ if the added cycle is linearly independent of those already in $\mathcal{B}_1$, followed by cycles of length 4, and so-on until $\mathcal{B}_1$ has the correct size. (Though Mathematica includes a native command to compute a cycle basis, these often contain many large elements sharing the same edge. A minimal weight basis, consisting of the shortest possible cycles, has few overlaps between elements and so is preferable here to reduce rejected state changes in Monte Carlo sampling.) The input–output paths $\mathcal{I}$ were simply constructed by taking (one of) the shortest path(s) between each input–output pair. Finally, in this case the basis $\mathcal{B}_2$ was not necessary as these graphs were always considered with all inputs and outputs activated, leaving no free outputs for flows to switch between; if it were needed, shortest paths between output pairs would again be a simple scheme.

Planar hexagonal networks. The cycle basis $\mathcal{B}_1$ was taken to be the set of equally-oriented flows around each hexagonal face of the lattice. Base input–output paths $\mathcal{I}$ were constructed to take the same form of route between each input–output pair so as to be pairwise disjoint, and output–output switching flows $\mathcal{B}_2$ were taken as the shortest paths (length 4) between adjacent outputs.

Hexagonal networks with crossover defects. We started from the sets $\mathcal{B}_1, \mathcal{B}_2, \mathcal{I}$ for the equivalent planar hexagonal lattice. Cycle basis elements corresponding to rewired hexagonal faces were removed from $\mathcal{B}_1$ and replacement basis elements then generated by applying Horton's algorithm, as above, starting from the existing

set of good cycles. New input–output paths $\mathcal{I}$ were found as shortest paths between input–output pairs. Finally, output–output connections $\mathcal{B}_2$ remained intact from the planar hexagonal network.

Exhaustive sampling

For sufficiently small graphs, including all those in Figs. 3 and 4 in the main text, we used exhaustive brute force sampling to evaluate all states. To improve efficiency, rather than naively sampling all $3^{|E|}$ candidate flows $\Phi \in \{-1, 0, +1\}^{|E|}$, we employed an approach exploiting the cycle basis to first construct undirected flows which are then counted with the appropriate orientational multiplicity.

For a set of binary coefficients $b_j \in \{0, 1\}$ $(1 \leqslant j \leqslant |\mathcal{B}|)$ of the base flows $\boldsymbol{X}_j \in \mathcal{B}$, the binary vector

$$\hat{\Phi} = \left(\sum_j b_j \boldsymbol{X}_j + \sum_i I_i \boldsymbol{D}_i \right) \bmod 2$$

defines an undirected flow on an unoriented form of Γ. Unless $\hat{\Phi}$ connects a pair of outputs, this is an unoriented form of a permissible flow. In the full AFN, this flow has energy $H(\hat{\Phi})$ and orientational multiplicity 2^m, where m is the number of closed cycles in $\hat{\Phi}$ as can be determined by a depth-first search. Each flow is guaranteed to be unique up to orientation because $\mathcal{B}_1$ is a basis for cycles on the closed graph, so we need only sample the $2^{|\mathcal{B}|}$ sets of coefficients $\{b_i\}$ to cover the entire space.

Monte Carlo sampling

For AFNs that could not be reasonably sampled by brute force, we employed a replica exchange (parallel tempering) algorithm. In essence, this algorithm runs multiple Markov chain Monte Carlo simulations of the AFN in parallel, each at a different temperature, randomly switching states between the replicas to better mix and explore different regions of configuration space. The pairwise switching is performed with an energy-dependent probability guaranteed to preserve a Boltzmann distribution of states for each replica. The details of such algorithms are well documented elsewhere [2], so we only discuss model specifics here.

At each iteration, the state of each replica is a coefficient vector $b_i \in \mathbb{Z}$ over $\boldsymbol{X}_i \in \mathcal{B}$ representing a valid flow $\Phi = \sum_i b_i \boldsymbol{X}_i + \sum_i I_i \boldsymbol{D}_i$. We use a Gibbs algorithm to step forward: a new value for each basis element b_i is picked, one at a time, according to the conditional distribution $p(b_i | \{b_j : j \neq i\})$, with zero probability for any value of b_i resulting in an invalid flow Φ. In practice we only need to check candidate values for b_i between $b_i - 2$ and $b_i + 2$ inclusive, since these are the largest changes that could occur without guaranteeing an edge going beyond flow ± 1. Sweeping over all elements b_i comprises one iteration.

We experimentally found an aggressive replica switching scheme to be effective, attempting to swap a random pair of adjacent replicas after every iteration, and we typically used six replicas as a good balance between computation time and speed of configuration space exploration. The replica temperatures within a run were roughly tuned to give a 30–50% swap acceptance rate, which we found reasonable for our purposes [3].

Termination was conditional on the running standard errors $\boldsymbol{\varepsilon}$ in the simulated observables $\langle \boldsymbol{x} \rangle$, determined by a subsampling procedure as follows. Every n_s iterations, a checkpoint i is reached and we compute the mean $\boldsymbol{m}^i$ of $\boldsymbol{x}$ only from iterations between this checkpoint and the last. From all past means $\{\boldsymbol{m}^i\}$ the running standard error $\boldsymbol{\varepsilon}^i$ is computed, which is used to approximate the standard error of the numerically computed mean $\langle \boldsymbol{x} \rangle$ from all iterations. The simulation was then terminated when both $\max_j \varepsilon_j^i < e_{\max}$ and $\langle \varepsilon_j^i \rangle_j < e_{\text{avg}}$ was satisfied for all replicas, for maximum and average absolute error thresholds set to $e_{\max} = 0.005$ and $e_{\text{avg}} = 0.0025$. For this to be statistically valid, a sufficiently large number of inter-checkpoint iterations n_s must be used such that the $\{\boldsymbol{m}^i\}$ are independent; in practice we found $n_s = 10000$ sufficed, verified by evaluating the autocorrelation of the inter-checkpoint means.

Numerical determination of permutation groups

The converged groups in Fig. 3 of the main text were determined as follows. For a given defect-riddled 3×5 hexagonal lattice, exhaustive sampling was first performed to determine the base set of permutations Σ. This was then imported into GAP [4], with which the convergence or periodicity of Σ was established by taking powers of Σ. (See later for proof that a convergent or periodic sequence is guaranteed.) Finally, if the concatenation converged, the generated group $G = \langle \Sigma \rangle$ was characterised using the `StructureDescription` and `MovedPoints` functions of GAP.

Note that composition of permutations is performed left-to-right in GAP, as opposed to right-to-left as we use here.

Random graph generation

In the main text, we consider permutations generated by samples of general random graphs at fixed numbers of bulk vertices V and inputs and outputs N. These were generated in Mathematica, as follows.

First, generate a random connected 3-regular base graph Γ_0 with $V - 2N$ internal vertices; if Γ_0 is isomorphic to any previous base graphs, discard and repeat as necessary. Next, pick $2N$ random edges. For each edge, split it in two by inserting a new vertex, and onto that new vertex add a new edge with a single degree-1 vertex at its other end. The first N new degree-1 vertices are then designated inputs and the other N are designated outputs; if both the input and output split-edge sets were chosen before for this Γ_0, repeat until distinct edges are picked. This process is then repeated multiple times with Γ_0 to create a set of random input–output augmentations (in our case 10 times), and this process is then repeated over multiple base graphs Γ_0 (in our case, 50 samples) at each number of internal vertices V. Finally, graphs not permitting any valid flow state with all inputs activated are discarded.

CONTINUUM MODEL

In the following, we present a definition of active flow networks for fully continuous flows $\phi_e \in \mathbb{R}$ employing local edge flux potentials and soft constraints, as considered in previous work [5, 6]. The discretisation employed in the main text is a simplification of this model allowing for much easier state space exploration and comprehension while still retaining the same qualitative features.

As in the main text, let Γ be a graph with edge set E and vertex set $V \cup \partial\Gamma$, where V is the set of interior vertices and $\partial\Gamma$ are degree-1 boundary vertices used as inputs and outputs, and assign every edge $e \in E$ an arbitrary orientation. These orientations define the $|V| \times |E|$ incidence matrix $\mathbf{D} = [D_{ve}]$ where D_{ve} is -1 if edge e is oriented outwards from vertex v, $+1$ if e points into v, and 0 if v and e are not incident. A continuous flow configuration $\Phi = (\phi_e)$ on Γ is a vector of signed flows $\phi_e \in \mathbb{R}$ along each $e \in E$, where $\phi_e > 0$ represents flow with the orientation of e and $\phi_e < 0$ is flow against the orientation of e. The pseudo-equilibrium model governing the system behaviour is then defined by an energy $H = H_0 + H_{\partial\Gamma}$ comprising a bulk energy H_0 and a boundary energy $H_{\partial\Gamma}$. We will discuss each of these in turn.

The bulk energy H_0 encodes spontaneous self-organised flow as well as incompressibility at internal vertices. This takes the form

$$H_0(\Phi) = \lambda \sum_{e\in E} U(\phi_e) + \tfrac{1}{2}\mu \sum_{v\in V} [(\mathbf{D}\cdot\Phi)_v]^2. \qquad (1)$$

The first term, with coupling constant λ, uses a double-welled potential $U(\phi_e) = -\frac{1}{4}\phi_e^4 + \frac{1}{6}\phi_e^6$ with minima at $\phi_e = \pm 1$ to model the propensity for spontaneous active flows $\phi_e \approx \pm 1$ along each edge $e \in E$. The second term, with coupling constant $\mu \gg \lambda$, imposes the soft incompressibility constraint $(\mathbf{D}\cdot\Phi)_v \approx 0$ at every bulk vertex $v \in V$. This results in states where every edge $e \in E$ is either flowing, with $\phi_e \approx \pm 1$, or in a non-flowing state, $\phi_e \approx 0$, with inflow and outflow balanced at every internal vertex $v \in V$. Configurations with greater numbers of flowing edges have lower H_0 and are therefore energetically preferred. Note that the higher-order form of $U(\phi_e)$ than a typical quartic potential in a Landau-like theory is to avoid a hidden symmetry when the incompressibility constraint is added, which leads to unrealistic continuously-valued flow configurations rather than semi-discretised flows $\phi_e \approx \{-1, 0, +1\}$; any double-welled potential of form other than simple quartic suffices. When the AFN is closed, so that $\partial\Gamma = \emptyset$, we have $H = H_0$ and overdamped Langevin dynamics then yield topology-dependent stochastic cycle switching phenomena [5].

Inputs and outputs are set and read through the boundary vertices $\partial\Gamma = \partial\Gamma_{\text{in}} + \partial\Gamma_{\text{out}}$. For a given digital input vector $\boldsymbol{I} = (I_v) \in \{0, 1\}^{|\partial\Gamma_{\text{in}}|}$ we constrain the vertex $v \in \partial\Gamma_{\text{in}}$ corresponding to input I_v to have net flux $(\mathbf{D}\cdot\Phi)_v \approx -I_v$, so that an activated input $I_v = 1$ injects matter into the network through v. Output vertices, on the other hand, are left unconstrained to allow matter to flow out of them or not as network interactions dictate; the output O_v of $v \in \partial\Gamma_{\text{out}}$ is read off as the (integer-rounded) flow $(\mathbf{D}\cdot\Phi)_v$ through v, forming the output vector $\boldsymbol{O} = (O_v) \in \{0, 1\}^{|\partial\Gamma_{\text{out}}|}$. Finally, to prevent spurious matter inflow through the outputs, we impose that the unique edge incident to each output vertex only permits flow toward the output, like a diode, as could be realised microfluidically through geometric channel patterning [7]. The input flux constraints and diode edges are implemented through the boundary energy

$$H_{\partial\Gamma} = \tfrac{1}{2}\mu \sum_{v\in\partial\Gamma_{\text{in}}} [(\mathbf{D}\cdot\Phi)_v + I_v]^2 + H_+, \qquad (2)$$

where the diode energy H_+ is infinite if any output-adjacent edge has flow into the network and zero otherwise.

Provided the coupling constants λ and μ are sufficiently large with respect to the intrinsic noise strength (pseudo-temperature) $T = \beta^{-1}$, stable states of H—that is, local minima—comprise flowing edges with $|\phi_e| = 1$ and non-flowing edges with $\phi_e = 0$, with flows balanced at every internal vertex and flow out of each activated input. The energy H is then proportional to the number of flowing edges, favouring states with more flow.

In this work, to tractably explore the entire state space on general networks, we employ a discrete form of this model by restricting exclusively to flows $\Phi \in \{-1, 0, +1\}^{|E|}$. Furthermore, taking the incompressible limit $\mu \to \infty$ with λ fixed restricts to the space of flows that exactly obey the inputs $\boldsymbol{I}$ and are perfectly incompressible at all internal vertices. In this subspace, the energy reads $H(\Phi) = -\frac{1}{12}\lambda \sum_e |\phi_e|$ proportional to the number of flowing edges $\phi_e = \pm 1$. As discussed in the main text, this results in a system that at least qualitatively replicates the full continuous lattice field model of

Eqs. (1) and (2).

PASSIVE FLOW

In the main text, we compare AFNs to passive flow networks, which we now give a brief overview of. We frame the mathematics in terms of microfluidics here, but the same formulae apply for all of Newtonian microfluidics, linear resistor networks and symmetric random walks [8]. In this section, repeated indices imply summation.

Let p_v be the hydrodynamic pressure at vertex v and let u_e be the flow along edge e (signed according to the graph orientation, as for the active flows). Mass conservation implies that at a bulk vertex $v \in V$, we have $D_{ve}u_e = 0$. Now, assuming the flow along an edge is proportional to the pressure gradient across it, as in Poiseuille flow, we have $v_e = -\kappa p_v D_{ve}$ for all edges e. In this formulation, κ is the flow conductivity which we take as uniform, modelling all edges as having the same length and diameter. Combining this with mass conservation thus results in the discrete Poisson problem $D_{ve}D_{we}p_w = 0$ for all bulk $v \in V$.

The Poisson problem must be augmented with boundary conditions. The input vertices $v \in \partial\Gamma_{\text{in}}$ have net flux $D_{ve}u_e = -I_v$, whence $\kappa D_{ve}D_{we}p_w = I_v$. Conversely, the output vertices $v \in \partial\Gamma_{\text{out}}$ are left open, which corresponds to fixing them at a reference pressure $p_v = 0$. Specifying these boundary conditions closes the problem, leaving an easily-solved linear system. (Note also that with these boundary conditions the flow is independent of the conductivity κ, as can be seen by rescaling p_v, so we can simply set $\kappa = 1$.)

MUTUAL INFORMATION

We briefly discuss here the derivation of our simplified formulae for input–output uncertainty in the main text.

For two random variables X and Y with joint distribution $p(x, y)$, the mutual information $J(X;Y)$ between X and Y is defined as [9]

$$J(X;Y) = \sum_{x,y} p(x,y) \log_2 \left[\frac{p(x,y)}{p(x)p(y)} \right],$$

where $p(x)$ and $p(y)$ are the marginal distributions of X and Y. (Taking the base-2 logarithm implies units of bits, and note that we are using J instead of the standard I to avoid a notational clash.)

We now identify X with inputs and Y with outputs. If we have z possible combinations of activated inputs from which we select uniformly at random, this implies $p(x) = 1/z$. Then since $p(x, y) = p(y|x)p(x) = p(y|x)/z$, we have

$$J(X;Y) = \frac{1}{z} \sum_{x,y} p(y|x) \log_2 \left[\frac{p(y|x)}{p(y)} \right].$$

The relative mutual information in determining X from Y is given by $U(X|Y) = J(X;Y)/H(X)$, where $H(X)$ is the entropy of X. For uniformly-chosen inputs the entropy is $H(X) = \log_2 z$ bits, giving

$$U(X|Y) = \frac{1}{z \log z} \sum_{x,y} p(y|x) \log \left[\frac{p(y|x)}{p(y)} \right].$$

The formulae for U_1 and U_2 in the main text are then obtained by taking $z = N$ and $z = N(N-1)$ for one and two labelled inputs, respectively.

PERMUTATION NETWORKS

When an N-input, N-output cubic AFN has all N inputs activated and uniquely labelled, mutual exclusivity of input–output paths means that the network permutes the input streams. Copies of the network can then be concatenated together to generate more complex permutations from the initial set. In this section we prove a variety of results on the convergence or non-convergence of this concatenation process. Given a set Σ of permutations an AFN is capable of performing, we first establish the fundamental behaviour of the n-fold concatenation Σ^n as a function of n. This foundation then lets us view concatenation as a random walk on the Cayley graph generated by Σ, from which we can use Markov chain theory to write down a graph-theoretic condition for convergence of concatenation.

Fundamental behaviour of Σ^n

In the following, Theorem 2 proves concatenation always settles into a repeated finite sequence and Theorem 4 gives equivalent conditions for convergence to a group. We also briefly give a sufficient condition to guarantee convergence and another to guarantee non-convergence.

Let Σ and Π be non-empty subsets of S_N where N is a positive integer. Define the product $\Sigma\Pi$ to be $\{\sigma\pi : \sigma \in \Sigma, \pi \in \Pi\}$, and for a positive integer m, let $\Sigma^m := \{\sigma_1 \cdots \sigma_m : \sigma_1, \ldots, \sigma_m \in \Sigma\}$. Note that $\Sigma^{k+\ell} = \Sigma^k\Sigma^\ell$ for all positive integers k and ℓ. Further, if Σ is a group, then $\Sigma^m = \Sigma$ for all $m > 0$.

Lemma 1. *Let Σ be a non-empty subset of S_N.*

(i) If the identity $\iota \in \Sigma$, then $\Sigma^m = \Sigma^{m+1}$ for some $m > 0$, and Σ^m is a group.

(ii) There exists $k > 0$ such that $\iota \in \Sigma^k$.

(iii) There exists $\ell > 0$ such that Σ^ℓ is a group.

Proof. (i) Suppose that $\iota \in \Sigma$. If $\sigma \in \Sigma^i$, then $\sigma = \sigma \cdot \iota \in \Sigma^i\Sigma = \Sigma^{i+1}$. Thus $\Sigma^i \subseteq \Sigma^{i+1}$ for all $i > 0$. Since S_N is finite, it follows that $\Sigma^m = \Sigma^{m+1}$ for some $m > 0$. Observe that $\Sigma^i = \Sigma^m$ for all $i \geqslant m$. Let $\sigma, \tau \in \Sigma^m$. Then $\sigma\tau \in \Sigma^m\Sigma^m = \Sigma^m$. Since every element of S_N has finite order (the order of an element σ is the minimal $v > 0$ such that $\sigma^v = \iota$), it follows that Σ^m is a group.

(ii) There exists $\sigma \in \Sigma$, and σ has finite order k, so $\iota = \sigma^k \in \Sigma^k$.

(iii) By (ii), there exists $k > 0$ such that $\iota \in \Sigma^k$. Let $\Pi := \Sigma^k$. By (i), there exists $m > 0$ such that Π^m is a group. Take $\ell := km$. □

Define the *signature* of Σ to be the smallest positive integer s such that Σ^s is a group. Since $\Sigma^{s+s} = \Sigma^s$, we may define the *period* of Σ to be the smallest positive integer p such that $\Sigma^{s+p} = \Sigma^s$.

Theorem 2. *Let Σ be a non-empty subset of S_N with signature s and period p. For all integers $k \geqslant s$, there exists an integer i such that $0 \leqslant i < p$ and $\Sigma^k = \Sigma^{s+i}$.*

Proof. There exists an integer i such that $0 \leqslant i < p$ and $k - s \equiv i \mod p$. Now $k = s + mp + i$ for some $m \geqslant 0$, and $\Sigma^{s+pm} = \Sigma^s$, so $\Sigma^k = \Sigma^{s+i}$. □

Thus Σ^n converges precisely when $p = 1$, and the group it converges to is Σ^s.

Let $\langle\Sigma\rangle$ denote the subgroup of S_N generated by the elements of Σ (that is, the group comprising all finite products of elements in Σ). Clearly $\Sigma^m \subseteq \langle\Sigma\rangle$ for all $m > 0$. In particular, $\Sigma^s \subseteq \langle\Sigma\rangle$, but in general these need not be equal. For example, if $|\Sigma| = 1$, then $\Sigma^s = \{\iota\}$.

Proposition 3. *Let Σ be a non-empty subset of S_N with signature s and period p. If p divides s, then $\Sigma^s = \langle\Sigma^p\rangle$.*

Proof. Since $\iota \in \Sigma^s$ and $\Sigma^s\Sigma^p = \Sigma^s$, it follows that $\Sigma^p \subseteq \Sigma^s$. Thus $\langle\Sigma^p\rangle \subseteq \Sigma^s$. On the other hand, $s = pm$ for some integer m, so $\Sigma^s = (\Sigma^p)^m \subseteq \langle\Sigma^p\rangle$. □

Theorem 4. *Let Σ be a non-empty subset of S_N with signature s and period p. The following are equivalent.*

(i) $p = 1$.

(ii) $\Sigma^s = \langle\Sigma\rangle$

(iii) $\Sigma \subseteq \Sigma^s$.

(iv) $\Sigma^k = \Sigma^s$ *for all* $k \geqslant s$.

(v) $\Sigma^\ell \subseteq \Sigma^{\ell+1}$ *for some* $\ell > 0$.

Proof. First, (i) implies (ii) by Proposition 3. Clearly (ii) implies (iii). Note that if $\Pi \subseteq \Sigma^s$, then $\Pi\Sigma^s = \Sigma^s$ since Σ^s is a group. Thus (iii) implies (iv). Clearly (iv) implies (v). Lastly, suppose that (v) holds. Now $\Sigma^k \subseteq \Sigma^{k+1}$ for all $k \geqslant \ell$. By Theorem 2, there exists $k \geqslant \ell$ such that $\Sigma^k = \Sigma^s$, and it follows that $\Sigma^s \subseteq \Sigma^{s+1} \subseteq \cdots \subseteq \Sigma^{s+p} = \Sigma^s$. Thus $\Sigma^{s+1} = \Sigma^s$ and (i) holds. □

Proposition 5. *Let Σ be a non-empty subset of S_N with period p. If $\iota \in \Sigma$, then $p = 1$.*

Proof. Let s be the signature of Σ. If $\iota \in \Sigma$, then $\Sigma \subseteq \Sigma^s$, so $p = 1$ by Theorem 4. □

The converse of Proposition 5 is not true in general. For example, if σ is a transposition and τ is a 3-cycle that is disjoint from σ, then $\{\sigma, \tau\}$ has signature 5 and period 1 but does not contain the identity.

Recall that any permutation σ in S_N may be written as a product of transpositions; the permutation σ is even (resp. odd) if this product has an even (resp. odd) number of transpositions [10].

Proposition 6. *Let Σ be a non-empty subset of S_N with signature s and period p. If Σ consists only of odd permutations, then s and p are even.*

Proof. If Σ consists only of odd permutations, then for all $i \geqslant 0$, the set Σ^{2i} consists only of even permutations, while Σ^{2i-1} consists only of odd permutations. Since $\iota \in \Sigma^s$ and ι is even, it follows that s and p are even. □

The converse of Proposition 6 is not true in general. For example, if σ and τ are disjoint transpositions, then $\{\sigma, \sigma\tau\}$ has signature 2 and period 2 but contains the even permutation $\sigma\tau$.

Concatenation as a Markov chain

Our main results in this section are Theorem 8, giving a geometric equivalence to convergence, and Theorem 10, showing that converging concatenations approach uniform random selection of permutations.

Let $\mathcal{G}$ be the Cayley graph of the group $G = \langle\Sigma\rangle$ generated by Σ; that is, the graph with vertex set G and directed edges $v \to w$ for those $v, w \in G$ for which there exists $\sigma \in \Sigma$ with $\sigma v = w$. (For our purposes we include loops $v \to v$ in $\mathcal{G}$ if the identity $\iota \in \Sigma$.) From the above we know $\Sigma^n \subseteq G$, with equality for large n if and only if $p = 1$, so any permutation state of an n-fold concatenation is a vertex in $\mathcal{G}$. This lets us view n-fold concatenation as an n-step random walk on $\mathcal{G}$.

Let p_σ be the probability that the basic AFN will apply permutation $\sigma \in \Sigma$. Now, starting from the identity map $X_0 = \iota$, let $X_n \in G$ be the random state selected by an n-fold concatenation. Then $X_{n+1} = \sigma X_n$ with probability p_σ, so the transition matrix $P_{gh} = \mathbb{P}(X_{n+1} = h | X_n = g)$ for X_n as a random walk on $\mathcal{G}$ reads

$$P_{gh} = \begin{cases} p_\tau & \text{if } \tau = hg^{-1} \in \Sigma, \\ 0 & \text{otherwise.} \end{cases} \tag{3}$$

Note that for fixed $g \in G$, $\tau = hg^{-1}$ is distinct for every $h \in G$ and therefore hits every element of Σ once, and vice-versa for fixed h. This guarantees that $\mathbf{P} = [P_{ij}]$ is

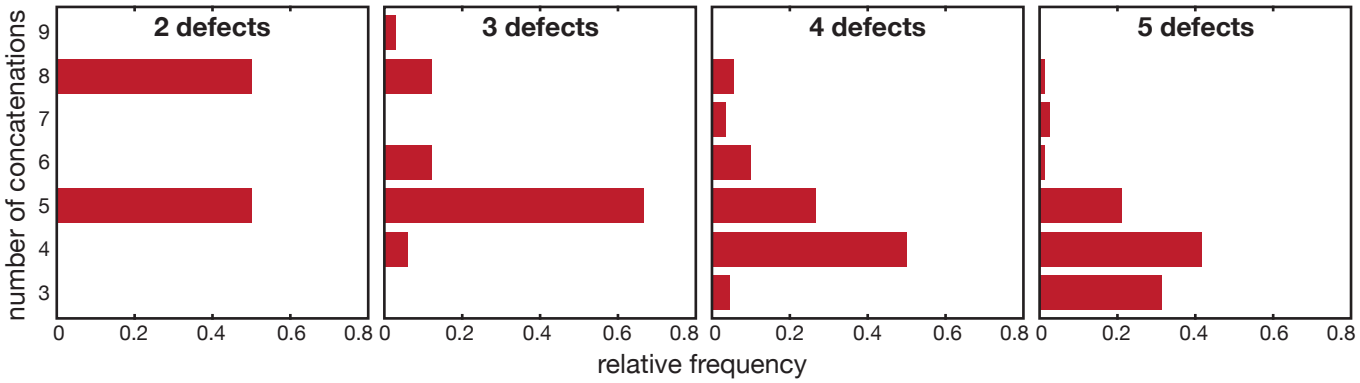

FIG. S1. Distributions of the number of concatenations required to generate S_5 using the relevant lattices in Fig. 3 of the main text, for S_5-generating lattices with between 2 and 5 crossover defects. As the number of defects increases, the number of concatenations required (the lattice signature) tends to fall amongst S_5-generating networks.

a doubly-stochastic matrix (that is, a matrix whose rows and columns each sum to 1) and so has a left eigenvector $(11\cdots1)$ with eigenvalue 1.

Lemma 7. *The Markov chain on G with transition matrix given by Eq.* (3) *is irreducible. Equivalently, the Cayley graph $\mathcal{G}$ is strongly connected.*

Proof. Suppose $X_n = g$ and we wish to reach state h in a finite number of steps. This is the statement that there exists a finite sequence of generators $\sigma_i, \sigma_j, \ldots, \sigma_k$ such that $\sigma_i\sigma_j\cdots\sigma_k = hg^{-1}$. By closure we have $hg^{-1} \in G$, and by definition of G all elements of G are finite products of $\sigma_i \in \Sigma$. Therefore this sequence exists. □

We can now give a geometric condition for convergence of concatenation to a group.

Theorem 8. *Let Σ be a non-empty subset of S_N with period p. Then $p = 1$ if and only if the Cayley graph $\mathcal{G}$ generated by Σ is aperiodic; that is, the g.c.d. of the lengths of all directed cycles in $\mathcal{G}$ is* 1.

Proof. The set Σ has $p = 1$ if and only if $\Sigma^n = G$ for all sufficiently large n by Theorem 4. This is equivalent to $\mathbb{P}(X_n = g) > 0$ for all $g \in G$ and all sufficiently large n, which is the statement that the Markov chain is aperiodic. Now, the possible return times for a state $g \in G$ correspond to the lengths of all directed cycles containing g in $\mathcal{G}$. Since the chain is irreducible (Lemma 7), all states have the same period [11] given by the g.c.d. of all possible return times for all states. The result follows. □

Instead of considering the entire graph $\mathcal{G}$, we can often simply read off aperiodicity directly from Σ, per the following. (Recall that the order of an element σ is the minimal $v > 0$ such that $\sigma^v = \iota$.)

Proposition 9. *Let Σ be a non-empty subset of S_N with period p. Let v_σ be the order of $\sigma \in \Sigma$. Then* $\gcd\{v_\sigma : \sigma \in \Sigma\} = 1$ *implies $p = 1$.*

Proof. We will establish that the chain is aperiodic, which implies $p = 1$ per Theorem 8. By Lemma 7 we need only establish aperiodicity of one state to imply aperiodicity of the chain [11]. Suppose $X_n = \iota$. Then $\mathbb{P}(X_{n+v_\sigma} = \iota) > 0$ by applying $\sigma \in \Sigma$ repeatedly v_σ times, so each generator $\sigma \in \Sigma$ implies a possible return time of v_σ. Thus the g.c.d. of all return times is at most $\gcd\{v_\sigma : \sigma \in \Sigma\}$, implying ι is aperiodic when this equals 1. □

Note that the condition of Proposition 9 is sufficient but not necessary for aperiodicity. Consider, for example, $\Sigma = \{(123), (234)\}$ (in group-theoretic cycle notation). Then both elements of Σ have order 3, but Σ^n converges because $\mathcal{G}$ contains cycles with the coprime lengths 3 and 4. In this case, $G = A_4$.

We conclude by observing that convergent concatenations select permutation elements uniformly at random in the large-n limit.

Theorem 10. *Let X_n be states of the Markov chain on G with transition matrix given by Eq.* (3)*. If the generating set Σ has period $p = 1$, then $\mathbb{P}(X_n = g) \to 1/|G|$ as $n \to \infty$ for all $g \in G$.*

Proof. Because the chain is finite, irreducible (Lemma 7) and aperiodic (Theorem 8), it is positive recurrent (as at least one state must be positive recurrent, and all other states are finitely reachable from such a state). Therefore it has a unique limiting distribution $P^n_{ij} \to \pi_j$ as $n \to \infty$ where $\pi_j = \pi_i P_{ij}$ and $\sum_j \pi_j = 1$ [11]. We remarked above that $\mathbf{P}$ has a left eigenvector $(11\cdots1)$ with eigenvalue 1, which must correspond to $\pi_j = 1/|G|$ by uniqueness. The result follows. □

Generating S_5 with small lattices

In the main text (Fig. 3), we show that creating permutation devices from the concatenation of small 3×5 hexagonal lattices with a few local crossover defects can

generate the full symmetric group S_5. For applications it may not be sufficient to know simply if it is possible, but rather how many networks are required and whether there is a trade-off necessary between network topological complexity and repetitions. In fact, as the number of defects increases and the probability of generating S_5 rises, the number of concatenations required (the signature s) typically falls. Fig. S1 visualises the signature distributions of the S_5-generating networks with between 2 and 5 crossover defects, showing the trend toward networks of lower signature as the number of defects rises.

* f.g.woodhouse@damtp.cam.ac.uk
† Present address: Department of Mathematics, Imperial College, London SW7 2BZ, U.K.

[1] J. D. Horton, A polynomial-time algorithm to find the shortest cycle basis of a graph. *SIAM J. Comput.* **16**, 358–366 (1987).

[2] D. J. Earl, M. W. Deem, Parallel tempering: Theory, applications, and new perspectives. *Phys. Chem. Chem. Phys.* **7**, 3910–3916 (2005).

[3] E. Bittner, A. Nußbaumer, W. Janke, Make life simple: Unleash the full power of the parallel tempering algorithm. *Phys. Rev. Lett.* **101**, 130603 (2008).

[4] The GAP Group, *GAP – Groups, Algorithms, and Programming, Version 4.8.3* (2016). `http://www.gap-system.org`.

[5] F. G. Woodhouse, A. Forrow, J. B. Fawcett, J. Dunkel, Stochastic cycle selection in active flow networks. *Proc. Natl. Acad. Sci. USA* **113**, 8200–8205 (2016).

[6] F. G. Woodhouse, J. Dunkel, Active matter logic for autonomous microfluidics. *Nat. Commun.* **8**, 15169 (2017).

[7] P. Denissenko, V. Kantsler, D. J. Smith, J. Kirkman-Brown, Human spermatozoa migration in microchannels reveals boundary-following navigation. *Proc. Natl. Acad. Sci. U.S.A.* **109**, 8007–8010 (2012).

[8] S. Redner, *A Guide to First-Passage Processes* (Cambridge University Press, New York, 2001).

[9] A. Borst, F. E. Theunissen, Information theory and neural coding. *Nat. Neurosci.* **2**, 947–957 (1999).

[10] J. Rotman, *An Introduction to the Theory of Groups* (Springer-Verlag, New York, 1995).

[11] J. R. Norris, *Markov Chains* (Cambridge University Press, New York, 1997).