

Documentation of scripts used to build model

Nadiah Kristensen

December 11, 2025

This document refers to the contents of `examples/deterministic/used_to_build/`, which was used to build and check the formalisation of the model.

Contents

1	Generalise notation	2
2	Deterministic transitions	4
2.1	Model	4
2.1.1	First-author primary decision layer	4
2.1.2	Action-matrix updates and memory	5
2.1.3	Co-author primary decision layer	6
2.1.4	Summary of primary decision rules	6
2.1.5	Strategy implementation layer	7
2.2	Check implementation of deterministic transitions	7
3	Transitions with implementation errors	9
3.1	Model	9
3.1.1	Permitted implementation-errors	9
3.1.2	Transition probabilities	11
3.1.3	Stationary distribution	13
3.2	Check implementation of transitions with errors	14
3.2.1	Check counts of errors	14
3.2.2	Check stationary distributions	15

1 Generalise notation

Goal: single payoff function that can account for all variants of the model we've discussed, including weighting of the first-author's contribution to paper's influence, the option of the country not first-authoring any papers, and the option of co-authors receiving benefits from papers they didn't co-author.

Consider a single round of the paper-authorship game. Players $i \in \{1, \dots, n\}$ each play the role of first-author and potential co-author on papers, denoted with subscripts (f) and (c) respectively. Each player j first-authors one paper with location T_j in topic space, and every other player has the option of coauthoring the paper.

Let $a_{i,j}$ be an indicator function such that $a_{i,j} = 1$ if player i co-authors the paper first-authored by player j , and $a_{i,j} = 0$ otherwise, with $a_{i,i} = 1$. The authorship decisions of player i are vector $\mathbf{a}_i = (a_{i,1}, a_{i,2}, \dots, a_{i,n})$, and the authorship decisions of all players are a vector of vectors $\mathbf{a} = (\mathbf{a}_1, \mathbf{a}_2, \dots, \mathbf{a}_n)$.

Each player has a preference in topic space p_i . Define the alignment between player i 's preference and the paper first-authored by player j as

$$L(p_i, T_j) = \frac{1}{1 + D(p_i, T_j)} \quad (1)$$

distance between author i 's preference and topic proposed by j

Consider the paper first-authored player i on topic T_i . The influence the paper has is proportional to the number of authors it has, and b is its maximum influence

$$B_i(\mathbf{a}) = \frac{b(w_f + \sum_{j \neq i} a_{j,i})}{w_f + n - 1}, \quad B_i(\mathbf{a}) \in [0, b] \quad (2)$$

first author's weight
coauthors
normalisation

The payoff the first author receives:

$$\pi_i^{(f)} = a_{i,i} \left(\gamma_{i,i} B_i(\mathbf{a}) L(p_i, T_i) - c_{i,i} \right) \quad (3)$$

benefit = influence \times alignment
costs of first-authoring
placeholder (probably assume $\gamma_{i,i} = 1$)

In Eq. 3, I could have split the costs of first-authoring into a proposing and writing cost; however, that raises the possibility that a first-author might not even bother proposing, which adds another strategy. Considering this is a bit of a side issue, I figure we could just assume proposing is free.

In the role of co-author, two possibilities:

1. Excludable good: if co-authors do not receive benefits from papers they did not co-author:

$$\pi_i^{(c)} = \sum_{j \neq i} a_{j,j} a_{i,j} \left(\gamma_{i,j} B_j(\mathbf{a}) L(p_i, T_j) - c_{i,j} \right) \quad (4)$$

1 if i coauthors j 's paper; 0 otherwise
co-authorship benefit differs from first-authorship
costs of coauthoring per paper

2. Non-excludable good: if co-authors do receive benefits from papers they did not co-author:

$$\pi_i^{(c)} = \sum_{j \neq i} a_{j,j} \left(\gamma_{i,j} B_j(\mathbf{a}) L(p_i, T_j) - a_{i,j} c_{i,j} \right) \quad (5)$$

1 if i coauthors j 's paper; 0 otherwise

Could merge Eqs. 4 and 5 with something like

$$\pi_i^{(c)} = \sum_{j \neq i} a_{j,j} \left(P(a_{i,j}) \gamma_{i,j} B_j(\mathbf{a}) L(p_i, T_j) - a_{i,j} c_{i,j} \right) \quad (6)$$

Public-good indicator function

where the public-good indicator function:

$$P(a_{i,j}) = \begin{cases} 1 & \text{if papers are modelled as non-excludable public goods,} \\ a_{i,j} & \text{otherwise.} \end{cases} \quad (7)$$

In a single round, the total payoff i receives is the sum of payoffs from their roles as first-author and co-author

$$\pi_i = \pi_i^{(f)} + \pi_i^{(c)} = \sum_{j=1}^n a_{j,j} (P(a_{i,j}) \gamma_{i,j} B_j(\mathbf{a}) L(p_i, T_j) - a_{i,j} c_{i,j}) \quad (8)$$

2 Deterministic transitions

2.1 Model

This model examines how players make rational decisions about authorship and co-authorship based on cost-benefit analysis. I propose a two-tier decision-making framework:

1. **Primary decision layer:** Authors first evaluate whether the direct benefits of first-authoring or co-authoring outweigh the costs.
2. **Strategy implementation layer:** Only when direct benefits do not justify participation will authors implement their strategic policies (e.g., tit-for-tat, all-cooperate, etc.).

This framework provides a realistic representation of publishing behaviour: authors will always participate in papers with clear individual benefits, and only engage strategic considerations for papers where direct benefits alone are insufficient.

To evaluate whether the direct benefits of authoring/co-authoring outweigh the costs, players calculate their unilateral switching gain, from non-authorship to authorship, using the most recently implemented actions.

2.1.1 First-author primary decision layer

Let us denote by $\mathbf{A}_{(i,j) \mapsto v}$ the matrix obtained from \mathbf{A} by replacing the element at position (i, j) with value v

$$\mathbf{A}_{(i,j) \mapsto v} := (a'_{k,l}) \text{ where } a'_{k,l} = \begin{cases} v & \text{if } (k, l) = (i, j) \\ a_{k,l} & \text{otherwise} \end{cases}$$

Then the unilateral decision by Player i to pursue authorship of paper j is represented by $\mathbf{A}_{(i,j) \mapsto 1}$, and the decision to forego authorship by $\mathbf{A}_{(i,j) \mapsto 0}$.

Consider Player i contemplating whether first-authorship in the next round would be worthwhile. They calculate their switching gain using the most recent information

$$\begin{aligned} & \text{payoff if } i \text{ had first-authored} \quad \text{payoff if } i \text{ had not first-authored} \\ & g_i^{(f)}(\mathbf{A}^t) = \pi_i^{(f)}(\mathbf{A}_{(i,i) \mapsto 1}^t) - \pi_i^{(f)}(\mathbf{A}_{(i,i) \mapsto 0}^t), \\ & \text{most-recent information} \uparrow \quad = \gamma_{i,i} B_i(\mathbf{A}^t) A_{i,i} - c_{i,i} \end{aligned} \quad (9)$$

If Player i 's unilateral first-authorship switching gain is positive, then according to their internal calculations, the benefits of first-authoring outweigh the costs; therefore, they will first-author in the next round

$$\begin{array}{c} \text{first-authoring would have} \\ \text{been worthwhile last round} \end{array} \quad \begin{array}{c} \text{first-author next round} \end{array} \quad \begin{array}{c} g_i^{(f)}(\mathbf{A}^t) > 0 \\ \implies \\ a_{i,i}^{t+1} = 1 \quad \forall i. \end{array} \quad (10)$$

However, if $g_i^{(f)}(\mathbf{A}^t) \leq 0$, Player i may nonetheless have other reasons for first-authoring, and therefore $a_{i,i}^{t+1}$ is calculated according to their first-authoring strategy.

2.1.2 Action-matrix updates and memory

The model must account for situations where potential first authors may choose not to publish in certain rounds. This raises the question of how potential co-authors should update their actions when no paper is published. The memory of past interactions is crucial for implementing conditional strategies.

For updating co-authorship actions, I implemented a memory-preserving approach: if a first author does not publish in a given round, then co-authors maintain their previous actions as 'memories'

$$\begin{array}{c} \text{first-author doesn't publish} \\ \downarrow \end{array} \quad \begin{array}{c} \text{co-author doesn't update action} \\ \downarrow \end{array} \quad \begin{array}{c} a_{j,j}^{t+1} = 0 \\ \implies \\ a_{i,j}^{t+1} = a_{i,j}^t \quad \forall i \neq j. \end{array} \quad (11)$$

Co-authors will only begin updating their actions again when first authors begin publishing again.

This approach, where action updates only occur when actions are actually implemented, ensures that all players' decisions are visible to all other players and can be incorporated into strategic calculations. For example, first authors can base their primary decision (Sect. 2.1.1) on the most recent actions of potential co-authors.

Example 1. Consider a 2-player game where: both co-authors' strategies is to do the opposite of what they did last, Player 2 always first-authors, and Player 1 stops first-authoring in the 3rd round and restarts in the 6th. The sequence of action matrices is shown below

$$\begin{array}{c} \text{first-authoring stops} \\ \downarrow \end{array} \quad \begin{array}{c} \text{first-authoring restarts} \\ \downarrow \end{array} \quad \begin{array}{c} \text{co-authorship memory fixed} \\ \uparrow \end{array} \quad \begin{array}{c} \text{co-authorship updating restarts} \\ \uparrow \end{array} \quad \begin{array}{c} \left(\begin{array}{cc} 1 & 0 \\ 1 & 1 \end{array} \right) \rightarrow \left(\begin{array}{cc} 1 & 1 \\ 0 & 1 \end{array} \right) \rightarrow \left(\begin{array}{cc} 0 & 0 \\ 0 & 1 \end{array} \right) \rightarrow \left(\begin{array}{cc} 0 & 1 \\ 0 & 1 \end{array} \right) \rightarrow \left(\begin{array}{cc} 0 & 0 \\ 0 & 1 \end{array} \right) \rightarrow \left(\begin{array}{cc} 1 & 1 \\ 1 & 1 \end{array} \right) \rightarrow \left(\begin{array}{cc} 1 & 0 \\ 0 & 1 \end{array} \right) \rightarrow \left(\begin{array}{cc} 1 & 1 \\ 1 & 1 \end{array} \right) \end{array}$$

2.1.3 Co-author primary decision layer

Players will contemplate co-authoring papers with all other players, both those who are currently first-authoring (i.e., players j such that $a_{j,j}^t = 1$) and those who are not; however, they will only implement their decisions and update their actions if the other player first-authors in the next round ($a_{j,j}^{t+1} = 1$). If a potential first-author does not publish in the next round, then the co-authors' cannot implement their decisions, and their actions do not update (Eq. 11).

Consider Player i contemplating whether co-authoring Player j 's paper would be worthwhile in the next round $t + 1$. They calculate their unilateral switching gain in a similar manner to first authors

$$g_{i,j}^{(c)}(\mathbf{A}^t) = \overset{\text{payoff if } i \text{ had co-authored } j\text{'s paper}}{\pi_i^{(c)}(\mathbf{A}_{(i,j) \mapsto 1}^t)} - \overset{\text{payoff if } i \text{ had not co-authored } j\text{'s paper}}{\pi_i^{(c)}(\mathbf{A}_{(i,j) \mapsto 0}^t)},$$

$$\overset{\text{most-recent information}}{\uparrow} = \gamma_{i,j} A_{i,j} \{P(1) B_j(\mathbf{A}_{(i,j) \mapsto 1}^t) - P(0) B_j(\mathbf{A}_{(i,j) \mapsto 0}^t)\} - c_{i,j} \quad (12)$$

where

$$B_j(\mathbf{A}_{(i,j) \mapsto v}^t) = \left(\frac{b}{w_f + n - 1} \right) \left(w_f + v + \sum_{\substack{k \neq j \\ k \neq i}} a_{k,j}^t \right)$$

If $g_{i,j}^{(c)}(\mathbf{A}^t) > 0$, then Player i will be interested in co-authoring with Player j in the next round; however, they will only do so if Player j first-authors in the next round

$$\overset{\text{co-authoring would have been worthwhile last round}}{g_{i,j}^{(c)}(\mathbf{A}^t) > 0} \text{ and } \overset{\text{first-author is publishing}}{a_{j,j}^{t+1} = 1} \implies \overset{\text{co-author next round}}{a_{i,j}^{t+1} = 1} \quad \forall i. \quad (13)$$

In the case where $a_{j,j}^{t+1} = 0$, the memory-preserving rule given in Eq. 11 applies.

Provided $a_{j,j}^{t+1} = 1$, similar to first-authors, if $g_{i,j}^{(c)}(\mathbf{A}^t) \leq 0$, then Player i will make their co-authorship decision based on their co-authorship strategy (e.g., all-c, all-d, pavlov, etc.).

2.1.4 Summary of primary decision rules

First author:

$$a_{i,i}^{t+1} = \begin{cases} 1 & \text{if } g_i^{(f)}(\mathbf{A}^t) > 0, \\ \text{use strategy} & \text{otherwise.} \end{cases} \quad (14)$$

Co-author:

$$a_{i,j}^{t+1} = \begin{cases} a_{i,j}^t & \text{if } a_{j,j}^{t+1} = 0, \\ 1 & \text{if } a_{j,j}^{t+1} = 1 \text{ and } g_{i,j}^{(c)}(\mathbf{A}^t) > 0, \\ \text{use strategy} & \text{otherwise.} \end{cases} \quad (15)$$

2.1.5 Strategy implementation layer

In the event where the direct benefits of authoring do not outweigh the costs, a player's authorship decision is made based on their strategic policy.

For first authors, I have implemented two simple strategies:

1. `always`: always first-author
2. `never`: never first-author

Note that first-authors with a 'never first-author' policy will still first-author in situations where the direct benefits outweigh the costs. If we are to focus on co-authorship decisions only, I would make 'never first-author' the default.

For co-authors, we have discussed both unconditional and conditional strategies. I have implemented the following unconditional co-authorship strategies:

1. `all-c`: always co-author
2. `all-d`: never co-author.

Conditional strategies like tit-for-tat or Pavlov are designed for situations where players can only either cooperate or defect in the previous round — there is no third option. Therefore, they do not apply to situations where one or both players were forced to opt out because the other did not first-author in the previous round. In that case, players should have some fall-back strategy. This is particularly important for players pursuing a purely co-authorship-based strategy and never first-authoring.

I have implemented one reciprocal strategy with two fallback options (note: the rules below assume the non-focal player j is first-authoring, $a_{j,j}^{t+1} = a_{j,j}^t = 1$)

- `pavlov-c`: Player i uses Pavlov strategy if they published last round; otherwise, they always cooperate

$$a_{i,j}^{t+1} = \begin{cases} 1 & \text{if } a_{i,i}^t = 1 \text{ and } ((a_{i,j}^t, a_{j,i}^t) = (0, 0) \text{ or } (a_{i,j}^t, a_{j,i}^t) = (1, 1)) , \\ 0 & \text{if } a_{i,i}^t = 1 \text{ and } ((a_{i,j}^t, a_{j,i}^t) = (0, 1) \text{ or } (a_{i,j}^t, a_{j,i}^t) = (1, 0)) , \\ 1 & \text{if } a_{i,i}^t = 0 . \end{cases}$$

if i first-authoring, i uses Pavlov
if i not first-authoring, i always cooperates

- `pavlov-d`: Player i uses Pavlov strategy if they published last round; otherwise, they always defect

$$a_{i,j}^{t+1} = \begin{cases} 1 & \text{if } a_{i,i}^t = 1 \text{ and } ((a_{i,j}^t, a_{j,i}^t) = (0, 0) \text{ or } (a_{i,j}^t, a_{j,i}^t) = (1, 1)) , \\ 0 & \text{if } a_{i,i}^t = 1 \text{ and } ((a_{i,j}^t, a_{j,i}^t) = (0, 1) \text{ or } (a_{i,j}^t, a_{j,i}^t) = (1, 0)) , \\ 0 & \text{if } a_{i,i}^t = 0 . \end{cases}$$

2.2 Check implementation of deterministic transitions

To check the logic of the deterministic transitions was implemented correctly, I plotted the deterministic transitions between all action states in a two-player game (Fig. 1). I chose the parameter

values such that co-authors always had a negative unilateral switching gain, but first-authors had a positive unilateral switching gain with 1 co-author and a negative switching gain otherwise. All transition graphs generated matched my expectations.

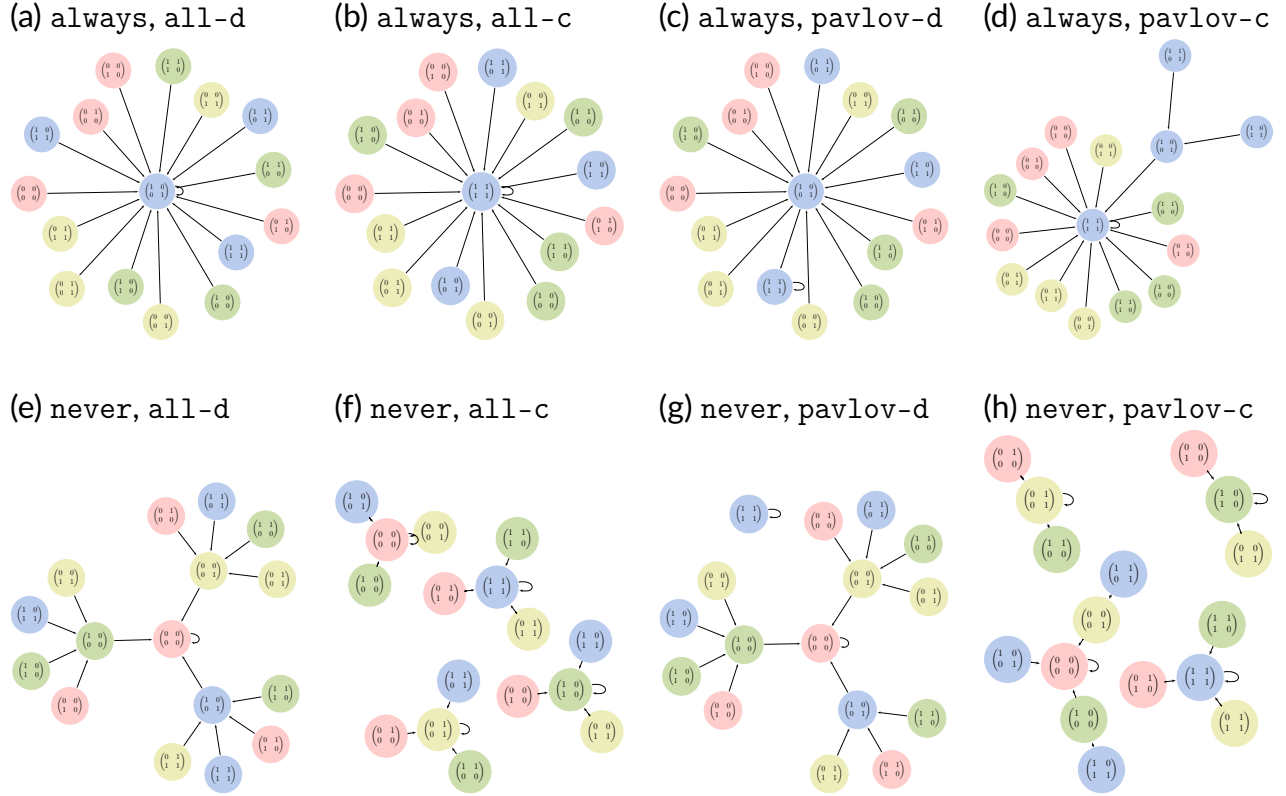


Figure 1: Deterministic transitions between states when all players pursue the same strategies. Nodes represent an action-state A , which in this 2-player game is a 2×2 binary matrix, and directed edges indicate how the actions of all players change at each round of the game. Strategies in each panel are labelled: first-author, co-author.

3 Transitions with implementation errors

3.1 Model

To determine the long-term payoffs players receive given their own and others' strategies, we take into account that players may make rare implementation errors. We modify the deterministic transition matrix D to obtain a transition matrix P that accounts for errors, find P 's stationary distribution, and calculate each player's payoffs at that expected distribution.

3.1.1 Permitted implementation-errors

Consider a predecessor A_p , its deterministic successor A_d ($D_{p,d} = 1$), and the ultimate successor A_u that is actually reached. The ultimate successor may differ from the deterministic successor due to implementation errors. Because implementation errors do not apply to co-authorship memories (i.e., memory-preserving assumption), then the set of ultimate successors is restricted to those whose co-authorship memories match the memory or action in the predecessor.

Example 2. Consider the following hypothetical transitions from predecessor A_p with deterministic successor A_d to ultimate successor A_u . Because Player 1 in A_u is not first-authoring, the error transition below is not permitted (not-permitted co-authorship marked in red).

$$\underbrace{\begin{pmatrix} 0 & 0 \\ 0 & 0 \end{pmatrix}}_{A_p} \rightarrow \underbrace{\begin{pmatrix} 1 & 1 \\ 1 & 1 \end{pmatrix}}_{A_d} \rightarrow \underbrace{\begin{pmatrix} 0 & 1 \\ 1 & 1 \end{pmatrix}}_{A_u}$$

◁

Specifically, denote the set of all players who are not first-authoring in the ultimate successor by \mathcal{J}_{A_u}

$$\mathcal{J}_{A_u} = \{j \in \mathcal{N} \mid (A_u)_{j,j} = 0\}.$$

Then all elements $(A_u)_{i,j}$ with $i \neq j$ and $j \in \mathcal{J}_{A_u}$ represent coauthorship memories. These memories must match the previous memory or co-authorship action in A_p . Therefore, the ultimate successor must satisfy

$$(A_u)_{i,j} = (A_p)_{i,j} \quad \forall i \neq j, \quad \forall j \in \mathcal{J}_{A_u}.$$

The condition $\forall i \neq j$ ensures that only coauthorship memories are constrained and not first-author actions; first-author actions are free to change either deterministically or through error. Allowing first-author actions to change ensures that all non-publishing states can communicate with all other states. For example, a state with $(A_p)_{j,j} = 0$ and with a deterministic successor with $(A_d)_{j,j} = 0$ can still ultimately reach states with $(A_u)_{j,j} = 1$ by Player j erroneously first-authoring (e.g., Fig. 2b).

The number of co-authorship memories in successor A_u is equal to the number of potential co-authors of players who are not first-authoring

$$m_m = (n - 1) \sum_{j=1}^n [(A_u)_{j,j} = 0],$$

where the bold square brackets are Iverson brackets (i.e., if Q is true, $[Q] = 1$; otherwise $[Q] = 0$).

Example 3. Consider a 3-player game and Player 2's first-authorship status. Fig. 2 illustrates all possible transitions from the predecessor to deterministic successor and ultimate successor states. For each transition, we consider how the second column of the action state changes (i.e., the first-authorship status of Player 2 and the statuses of its potential coauthors), and we count: the number of co-authorship memories (m_m), the maximum number implementation errors possible, and the minimum number of correct implementations possible.

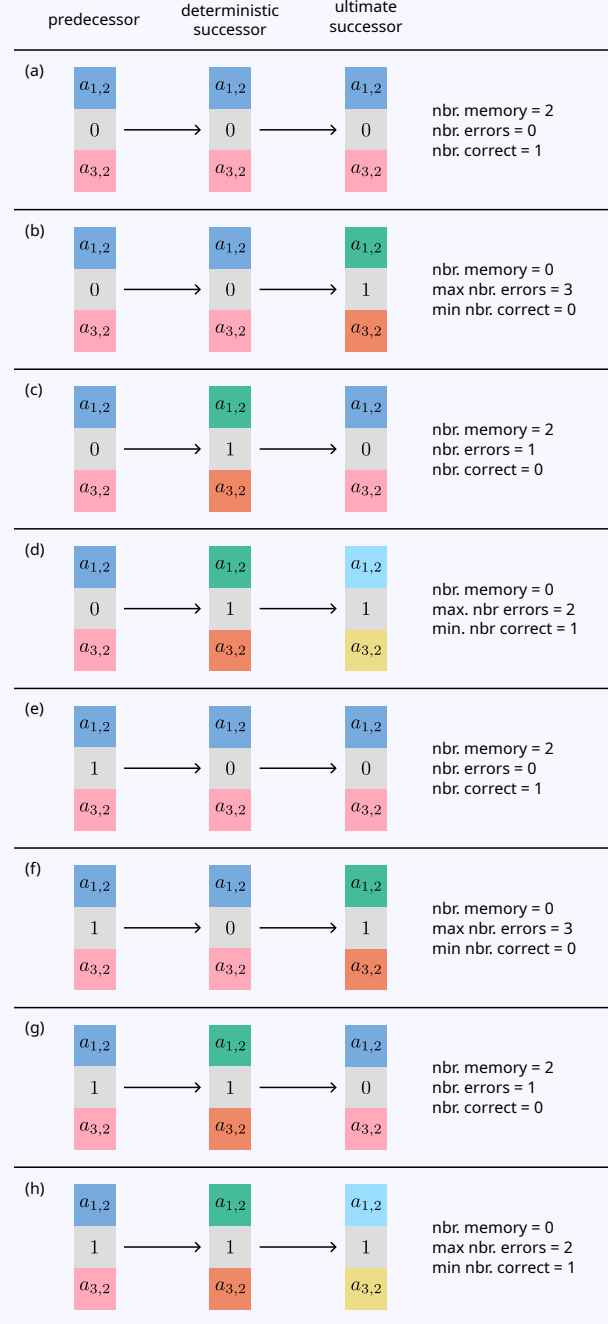


Figure 2: The possible transitions in Player 2's first-authorship status from the predecessor state to the deterministic successor and ultimate successor. Each column represents the second column of the action state. Player 2 has two potential co-authors, and situations where their co-authorship status has the potential to change are highlighted by different colours.



3.1.2 Transition probabilities

The probability of transitioning from \mathbf{A}_p to the ultimate successor \mathbf{A}_u instead of the deterministic successor \mathbf{A}_d is a function of the number of errors that separate \mathbf{A}_u from \mathbf{A}_d . Specifically, if ε is the probability of a single implementation error, m_e is the number of implementation errors, and m_c is the number of correct implementations, then the probability of transition from \mathbf{A}_p to \mathbf{A}_u is

$$\mathbb{P}[\mathbf{A}^{t+1} = \mathbf{A}_u \mid \mathbf{A}^t = \mathbf{A}_p; \mathbf{A}_d] = (1 - \varepsilon)^{m_c} \varepsilon^{m_e}. \quad (16)$$

Note that the implementation-error calculation in Eq. 16 does not count memory states because memory states are not actions implemented. In particular, memory states (which do not change) do not count towards the number of actions correctly implemented m_c .

Example 4. Consider the hypothetical transitions below, from predecessor \mathbf{A}_p with deterministic successor \mathbf{A}_d to ultimate successor \mathbf{A}_u . Errors are marked in red, correct implementations in blue, and memories in grey.

$$\underbrace{\begin{pmatrix} 0 & 0 \\ 0 & 1 \end{pmatrix}}_{\mathbf{A}_p} \rightarrow \underbrace{\begin{pmatrix} 0 & 0 \\ 0 & 0 \end{pmatrix}}_{\mathbf{A}_d} \rightarrow \underbrace{\begin{pmatrix} 1 & 0 \\ 1 & 0 \end{pmatrix}}_{\mathbf{A}_u} \quad \triangleleft$$

The number of errors is the sum of implementation errors committed by first authors and co-authors

$$m_e(\mathbf{A}_p, \mathbf{A}_d, \mathbf{A}_u) = m_e^{(f)}(\mathbf{A}_d, \mathbf{A}_u) + m_e^{(c)}(\mathbf{A}_p, \mathbf{A}_d, \mathbf{A}_u), \quad (17)$$

where number of errors committed by first authors is

$$m_e^{(f)}(\mathbf{A}_d, \mathbf{A}_u) = \sum_{j=1}^n [(\mathbf{A}_u)_{j,j} \neq (\mathbf{A}_d)_{j,j}], \quad (18)$$

and the number of errors by committed co-authors is

$$m_e^{(c)}(\mathbf{A}_p, \mathbf{A}_d, \mathbf{A}_u) = \sum_{j \in \mathcal{N} \setminus \mathcal{J}_{\mathbf{A}_u}} \sum_{i=1}^n [(\mathbf{A}_u)_{i,j} \neq (\mathbf{A}_d)_{i,j}]. \quad (19)$$

The number of correct implementations is

$$m_c(\mathbf{A}_p, \mathbf{A}_d, \mathbf{A}_u) = n + \underbrace{(n-1) |\mathcal{N} \setminus \mathcal{J}_{\mathbf{A}_u}|}_{\text{nbr. co-author actions implemented}} - m_e(\mathbf{A}_p, \mathbf{A}_d, \mathbf{A}_u). \quad (20)$$

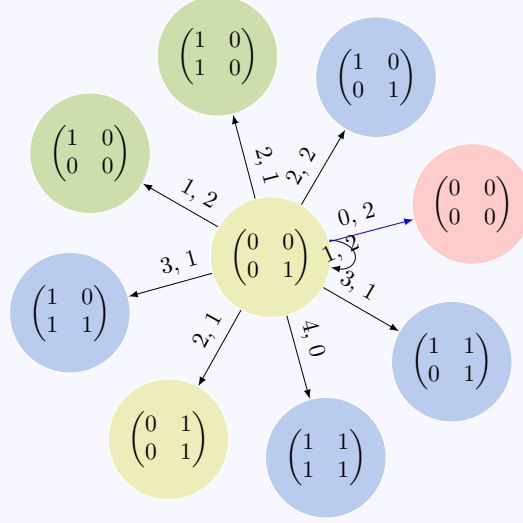
The transition probabilities are encoded in a transition matrix

$$\mathbf{P}_{p,u} = \mathbb{P}[\mathbf{A}^{t+1} = \mathbf{A}_u \mid \mathbf{A}^t = \mathbf{A}_p; \mathbf{A}_d] \quad (21)$$

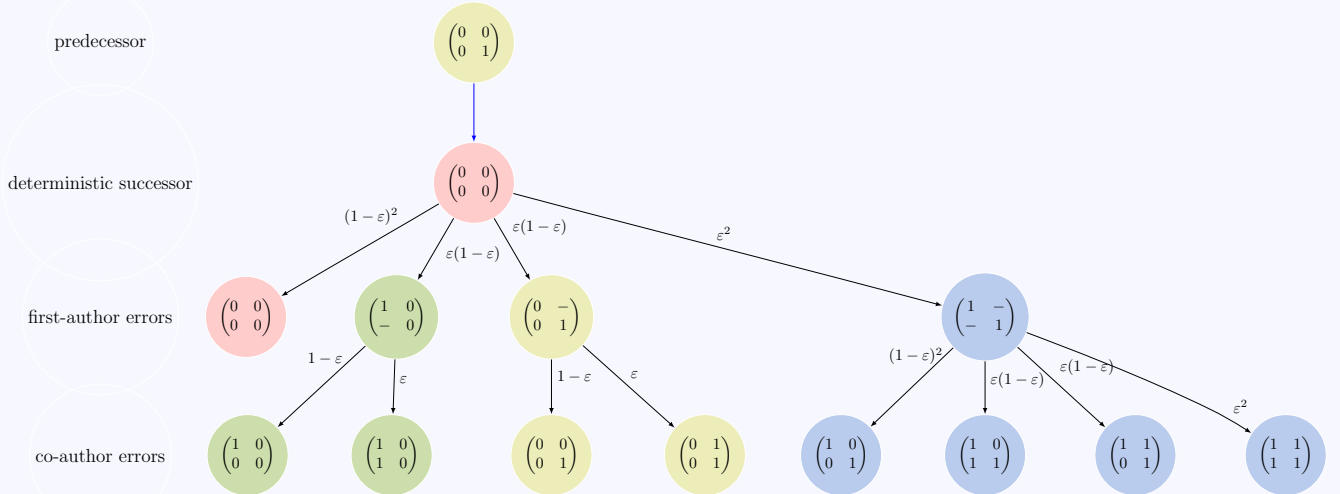
Example 5. Consider the predecessor and deterministic successor

$$\mathbf{A}_p = \begin{pmatrix} 0 & 0 \\ 0 & 1 \end{pmatrix} \text{ and } \mathbf{A}_d = \begin{pmatrix} 0 & 0 \\ 0 & 0 \end{pmatrix}.$$

The diagram below illustrates all of possible ultimate successors, \mathbf{A}_u , with the edge labels indicating the number of errors and correct actions, (m_e, m_c) , respectively



The probability of transition from \mathbf{A}_p to each \mathbf{A}_u can be calculated by counting the number of errors committed by first authors, and then the number of errors committed by co-authors given the first-author errors.



After the first-author-errors stage, co-author actions of non-publishing first-authors are constrained to match their values in the predecessor (Sect. 3.1.1), whereas co-author actions of publishing first-authors may take any value. The latter are marked – in the first-author errors stage, and their possible outcomes realised in the row below.

One can verify that the product of errors along each path from \mathbf{A}_p to each \mathbf{A}_u matches Eq. 16 with the m_e and m_c values given in the earlier diagram. \triangleleft

3.1.3 Stationary distribution

Index the possible action states A_i for $i \in \{1, \dots, 2^{n^2}\}$. We are interested in calculating the stationary distribution

$$\mathbf{v} = \mathbf{v} \mathbf{P}, \quad (22)$$

where $\mathbf{v} = (v_1, \dots, v_{2^{n^2}})$ and each v_i is the long-term expected time spent in action state A_i .

A typical approach to calculating \mathbf{v} is to set ε to some arbitrarily small number and then solve for \mathbf{v} numerically (eigenvector). However, it is possible to obtain an analytic solution for $\varepsilon \rightarrow 0$.

To obtain the analytic stationary distribution, I used the Python module SymPy to encode \mathbf{P} symbolically and then solved Eq. 22. The example in [this blog post](#) illustrates how the solution is found. In short, first we choose a maximum ε order to consider, and write \mathbf{v} as a polynomial in ε to that order. Then we match same- ε -order terms Eq. 22, and attempt to solve for the ε^0 coefficients (solving a system of linear equations). If the solution cannot be found, the maximum ε order is increased until the solution is found.

3.2 Check implementation of transitions with errors

3.2.1 Check counts of errors

Fig. 3 shows the results of the m_e and m_c calculations for an arbitrary sampling of scenarios. The values match my expectations.

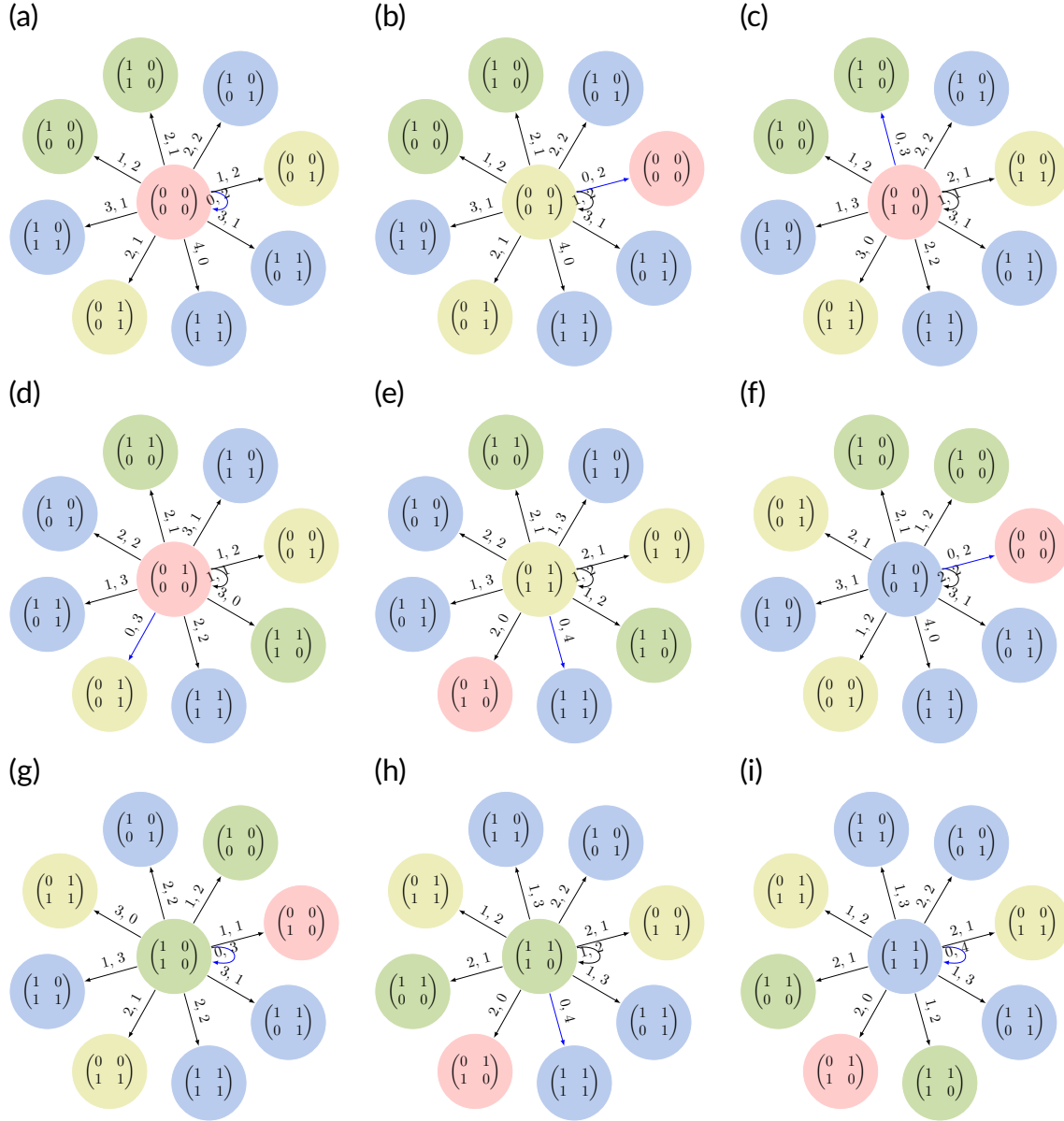


Figure 3: Transitions from selected predecessors (central node) to valid ultimate successors (outer nodes) and the number of erroneous and correct implementations between them (edge labels). The transition to the deterministic successor is highlighted in blue, and the edge labels indicate the number of errors and correct actions (m_e, m_c), respectively.

3.2.2 Check stationary distributions

Check 1: co-authors use `pavlov-c`, first authors use `never`

Using the analytic approach, I found the solution to the stationary distribution at order ε^1 , and the stationary distribution distribution was a single state with probability 1: $((0, 0), (0, 0))$.

To verify this was correct, I took the deterministic transition diagram (Fig. 4a) and created an implementation-errors transition diagram by including all transitions of order ε^1 (Fig. 4b). The deterministic dynamics revealed three absorbing states; however, when transitions caused by single-action errors are included (red edges, Fig. 4b), it is evident from inspection that $((0, 0), (0, 0))$ is indeed the stationary state.

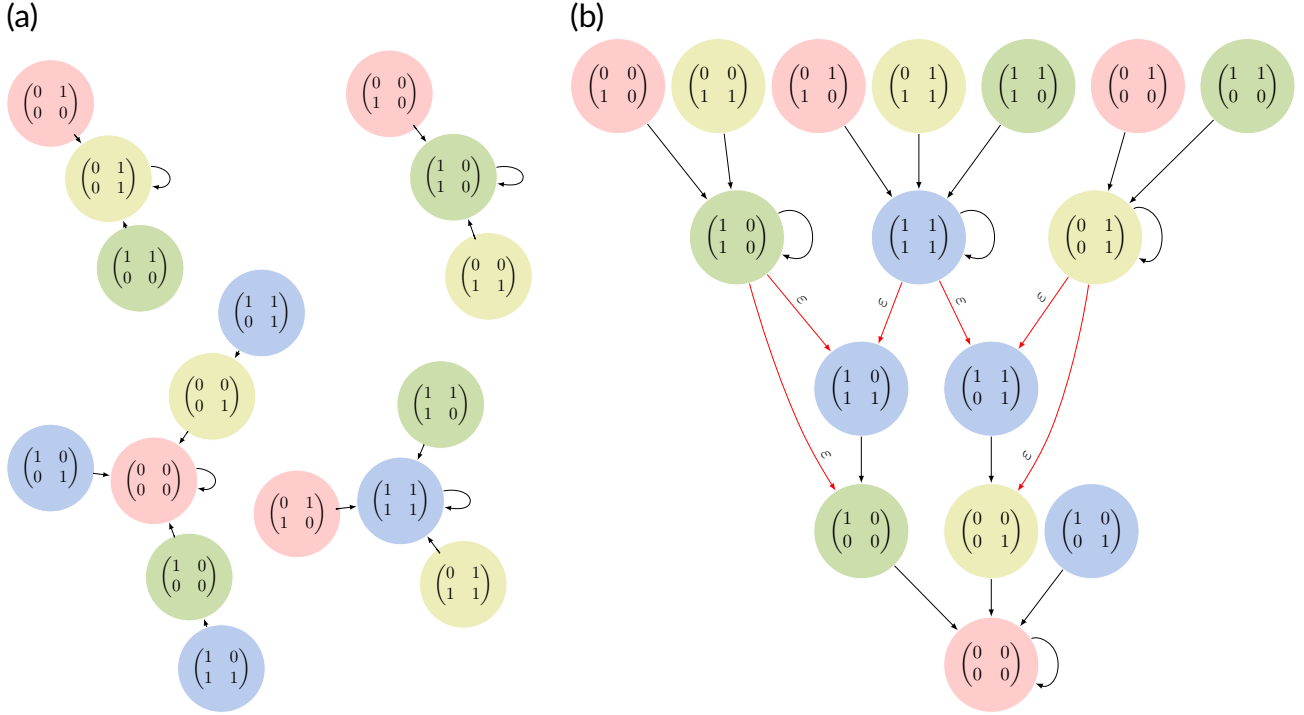


Figure 4: Dynamics when co-authors use `pavlov-c` and first authors use `never`, (a) the deterministic dynamics, and (b) the dynamics with implementation errors included (red edges) For simplicity, only single-action errors that transition the game between different basins of attraction are shown.

Check 2: co-authors use `all-c`, first authors use `never`

Using the analytic approach, I found the solution to the stationary distribution at order ε^1 , and the stationary distribution was a single state with probability 1: $((0, 0), (0, 0))$. I verified that it was correct by drawing the transition diagram with the ε^1 errors included (Fig. 5b).

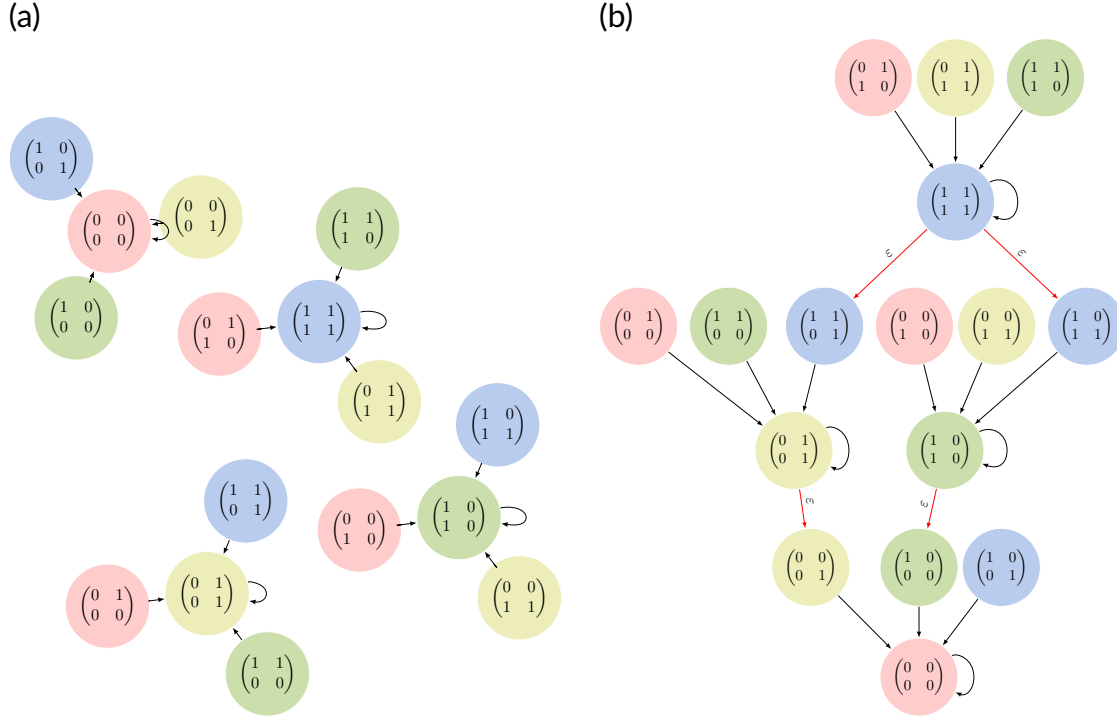


Figure 5: Dynamics when co-authors use `all-c` and first authors use `never`, (a) the deterministic dynamics, and (b) the dynamics with implementation errors included (red edges). For simplicity, only single-action errors that transition the game between different basins of attraction are shown.