

Scheduling Dashain Tika

Exact bounds on sitting and standing costs

Apoorva Lal

Krabbs

2026-10-21

1 The scheduling problem

During Dashain, elders give tika and blessings to younger family members. Consider a gathering with the following rule: **every person gives tika to everyone younger than them**, one pair at a time. A recipient sits to receive tika and eventually stands to leave. Repeating this for each elder can mean many sit–stand cycles. The scheduling question is whether changing the order can reduce that movement while completing every required tika.

Assume distinct ages and index the $N \geq 2$ people from youngest, 1, to eldest, N . Each older person gives exactly once to each younger person, so the required directed dyads are

$$\mathcal{E} = \{(i, j) : 1 \leq j < i \leq N\}, \quad M = |\mathcal{E}| = \binom{N}{2}.$$

In (i, j) , person i gives and person j receives. A schedule is a permutation $\pi = ((g_t, r_t))_{t=1}^M$ of $\mathcal{E}$. Interactions occur sequentially. Age determines the direction of each interaction. The schedule determines when it happens.

Let e be the combined cost of sitting down and subsequently standing up once. If each individual action costs e , replace this cycle cost by $2e$.

1.1 Eldest-first policy

Person N gives tika to $N - 1, N - 2, \dots, 1$, in that order. Person $N - 1$ then gives to $N - 2, \dots, 1$, and so on until person 2 gives to person 1. Thus the dyads are processed as

$$(N, N - 1), (N, N - 2), \dots, (N, 1), (N - 1, N - 2), \dots, (2, 1).$$

In this baseline, **the recipient sits and stands for every tika**, even when their next interaction immediately follows. The number of recipient cycles is

$$(N - 1) + (N - 2) + \dots + 1 = M, \quad C_{\text{recipient}} = Me.$$

If the giver also sits, each giver can remain seated throughout their block. Counting both roles then gives $M + N - 1$ cycles and cost $(M + N - 1)e$. These two accounting conventions lead to the respective comparisons below.

i Optimal sitting and standing costs

For a fixed giver position and recipient position, let $c_G, c_R \geq 0$ be the respective costs of a seated episode. Each episode includes entry, consecutive interactions in that role, and exit. The exact minimum is

$$C^* = (N - 1) \max(c_G, c_R) + (M - N + 2) \min(c_G, c_R).$$

Batch the more expensive role and retain the other occupant at each block boundary. The schedules below attain this bound.

The physical arrangement determines the relevant special case:

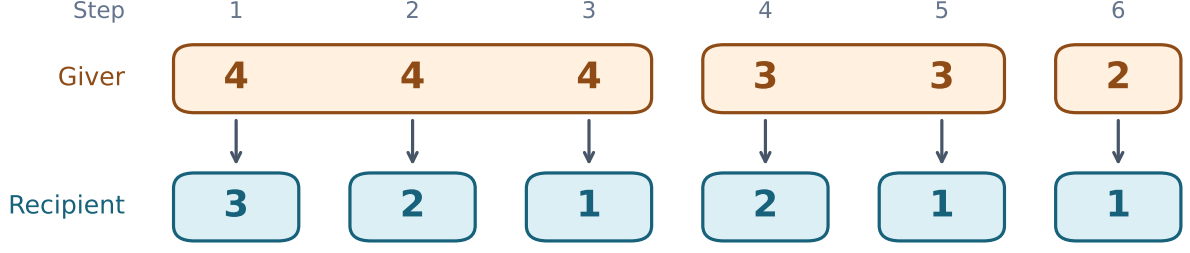
Arrangement	Minimum total sit–stand cost
One recipient sits; givers can approach with zero charged seating cost	$(N - 1)e$
One giver seat and one recipient seat; both occupants' cycles cost e	$(M + 1)e$
Everyone can remain seated and all pairs can interact from their seats	Ne

The first two cases follow from the boxed formula. The fully seated case is a separate arrangement with enough seats and access between every pair. All costs include initial sitting and final standing. Every arrangement still requires M tikas.

1.2 Comparing the dyads

For four people, every schedule contains the same six older-to-younger pairs. Figure 1 shows their execution order and the resulting seated episodes. Recipient batching reduces the recipient count from six to three; both optimized orders attain the two-seat minimum of seven total cycles. The giver-block construction also preserves eldest-first priority among givers.

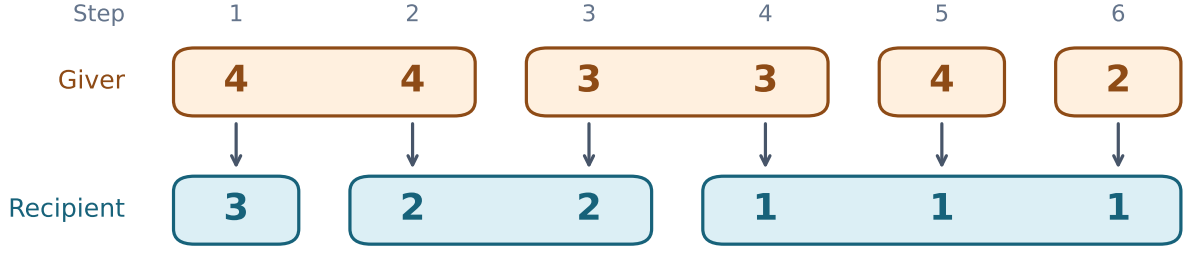
Eldest first · recipient resets



Recipient only: 6e

Two seats: 3e + 6e = 9e

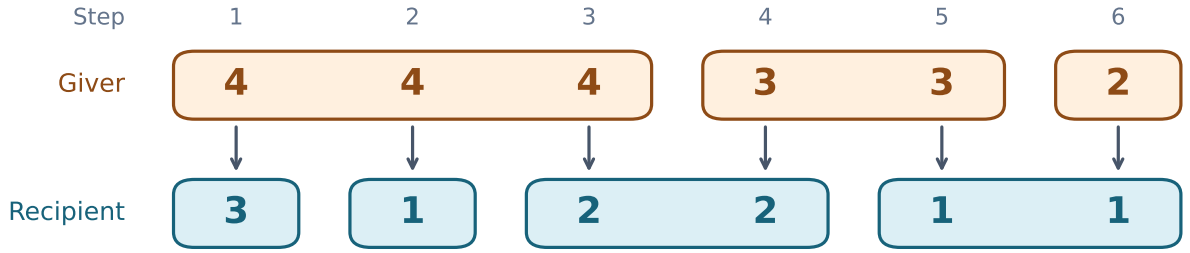
Optimal recipient batching



Recipient only: 3e

Two seats: 4e + 3e = 7e

Optimal two-seat giver blocks



Recipient only: 4e

Two seats: 3e + 4e = 7e

Figure 1: The same six directed dyads in three schedules, read from left to right. Each downward arrow is one tika, from the numbered giver to the numbered recipient. Each continuous band is one seated episode, including its eventual stand-up. The baseline resets the recipient after every tika. Recipient batching is optimal for recipient-only costs; both lower schedules are optimal for equal-cost two-seat service. All costs use e per sit–stand cycle.

2 A setup-cost model

There is an active giver position and an active recipient position. Consecutive interactions can reuse an occupant in the **same role**. Changing roles starts a new role episode. Under the two-seat interpretation, the chairs have fixed roles; the participant gets up to change chairs.

Let $c_G \geq 0$ be the cost of one giver episode and $c_R \geq 0$ the cost of one recipient episode. A seated episode includes entry, any number of consecutive interactions in that role, and exit. Count episodes by

$$G(\pi) = 1 + \sum_{t=1}^{M-1} \mathbf{1}\{g_{t+1} \neq g_t\}, \quad R(\pi) = 1 + \sum_{t=1}^{M-1} \mathbf{1}\{r_{t+1} \neq r_t\}.$$

For any schedule, keep the occupant seated across consecutive uses of the same role. Additional stand–sit resets only increase its cost. The objective is

$$\min_{\pi} C(\pi), \quad C(\pi) = c_G G(\pi) + c_R R(\pi).$$

This model describes two different practical cases. Setting $(c_G, c_R) = (0, e)$ allows givers to approach a seated recipient, with their movement uncharged. Setting $(c_G, c_R) = (e, e)$ charges both people at a two-seat station. Walking, waiting, tray transfers, and reaching costs are excluded. They can change which arrangement is attractive.

If each tika itself takes time τ , total duration is $\tau M + C(\pi)$ when the setup costs are measured in time. The number of tikas stays quadratic in N , even when movement costs become linear.

3 Three lower bounds and an exact optimum

Every person $2, \dots, N$ must give, and every person $1, \dots, N - 1$ must receive. Therefore

$$G(\pi) \geq N - 1, \quad R(\pi) \geq N - 1.$$

Two consecutive interactions are distinct, so at least one of their giver and recipient identities changes. The first interaction starts two episodes; the remaining $M - 1$ interactions each start at least one. Consequently,

$$G(\pi) + R(\pi) \geq M + 1.$$

These three inequalities establish the lower bound in the introductory box.

Lower bound. Suppose $c_R \geq c_G$. Then

$$\begin{aligned} C(\pi) &= c_G \{G(\pi) + R(\pi)\} + (c_R - c_G)R(\pi) \\ &\geq c_G(M + 1) + (c_R - c_G)(N - 1) \\ &= c_R(N - 1) + c_G(M - N + 2). \end{aligned}$$

The case $c_G \geq c_R$ follows by interchanging the roles. The constructions in the next section attain the bound. This proves global optimality over all $M!$ schedules. No probabilistic approximation is required.

The rule is to **batch the more expensive role**, keeping its occupant in place for all their interactions, and arrange the block boundaries to retain the other occupant too.

4 Constructing optimal schedules

The following constructors generate the schedules illustrated below.

4.1 Recipient batching: the linear-cost case

Seat person $N - 1$ and have every older person give them tika. Then seat person $N - 2$ and repeat, continuing to person 1. Every recipient sits once:

$$R = N - 1, \quad C = (N - 1)e \quad \text{when} \quad c_G = 0, c_R = e.$$

Every recipient must sit at least once, so this already attains the lower bound. Compared with Me , the saving is

$$Me - (N - 1)e = \frac{(N - 1)(N - 2)}{2}e, \quad \frac{Me}{(N - 1)e} = \frac{N}{2}.$$

For $N = 10$, there are still 45 tikas, while recipient sit-stand cycles fall from 45 to 9. Givers must be able to approach the current recipient.

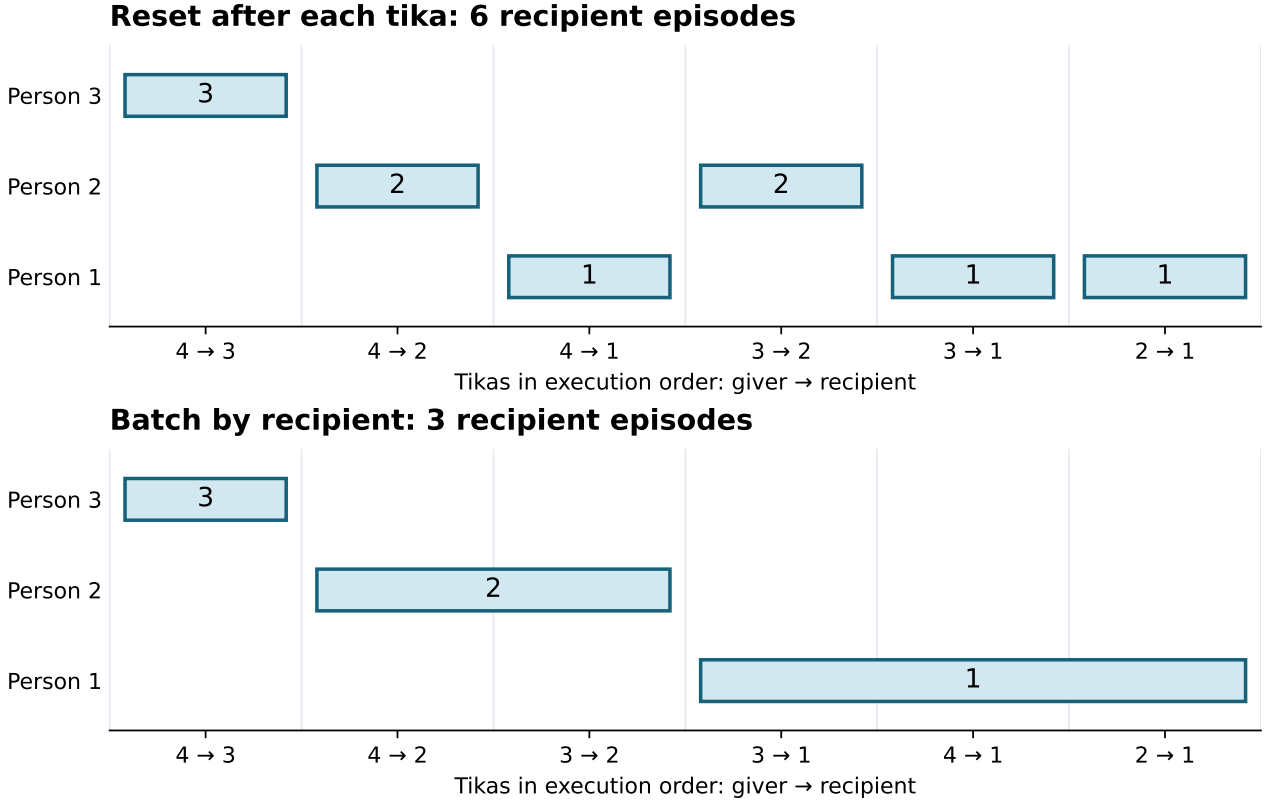


Figure 2: Recipient-only seating costs, illustrated with four people and six tikas. Each blue band is one sit–stand episode: it begins when a person sits and ends when they stand. The upper policy resets the recipient after every tika; the lower policy lets each recipient remain seated through all their elders. Giver movement is uncharged.

4.2 Giver blocks with a shared recipient at each boundary

When giver episodes are at least as expensive as recipient episodes, preserve the eldest-first giver blocks. In the block for giver $i \geq 3$, use this recipient order:

$$i - 1, \quad i - 3, i - 4, \dots, 1, \quad i - 2.$$

The middle list is empty when $i = 3$. Giver 2 visits recipient 1. Thus giver i finishes with recipient $i - 2$, who stays seated while giver $i - 1$ takes over and also starts with $i - 2$.

For $N = 4$, the schedule is

$$\underbrace{(4, 3), (4, 1), (4, 2)}_{\text{giver 4}}; \underbrace{(3, 2), (3, 1)}_{\text{giver 3}}; (2, 1).$$

Each boundary changes the giver alone. Within a block, only the recipient changes. There are $N - 1$ giver episodes and $M + 1$ total episodes:

$$G = N - 1, \quad R = M - N + 2.$$

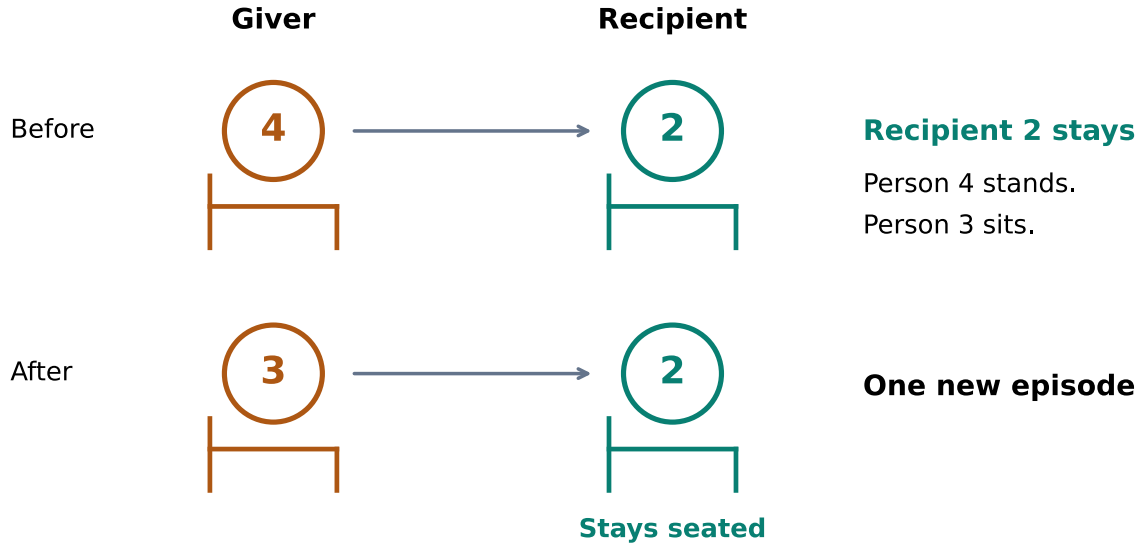


Figure 3: A boundary between giver blocks in the four-person schedule: (4,2) is followed by (3,2). Recipient 2 stays seated while giver 4 is replaced by giver 3. The full six-tika schedule uses three giver episodes and four recipient episodes, attaining the two-seat minimum of seven.

4.3 Recipient blocks with a shared giver at each boundary

For $c_R \geq c_G$, use the symmetric construction. Recipient $N - 1$ first receives from giver N . For each subsequent recipient $j = N - 2, \dots, 1$, the giver order is

$$j + 2, \quad N, N - 1, \dots, j + 3, \quad j + 1.$$

The middle list can be empty. Recipient j ends with giver $j + 1$, who also starts the block for recipient $j - 1$. The same giver stays in place across each boundary. The executable construction obtains this schedule by reversing the giver-block schedule, age labels, and role labels.

For $N = 4$, this gives

$$(4, 3) ; \underbrace{(4, 2), (3, 2)}_{\text{recipient 2}} ; \underbrace{(3, 1), (4, 1), (2, 1)}_{\text{recipient 1}}.$$

The recipient blocks proceed in descending age order, so each person receives from all their elders before giving to anyone younger. The episode counts are

$$R = N - 1, \quad G = M - N + 2.$$

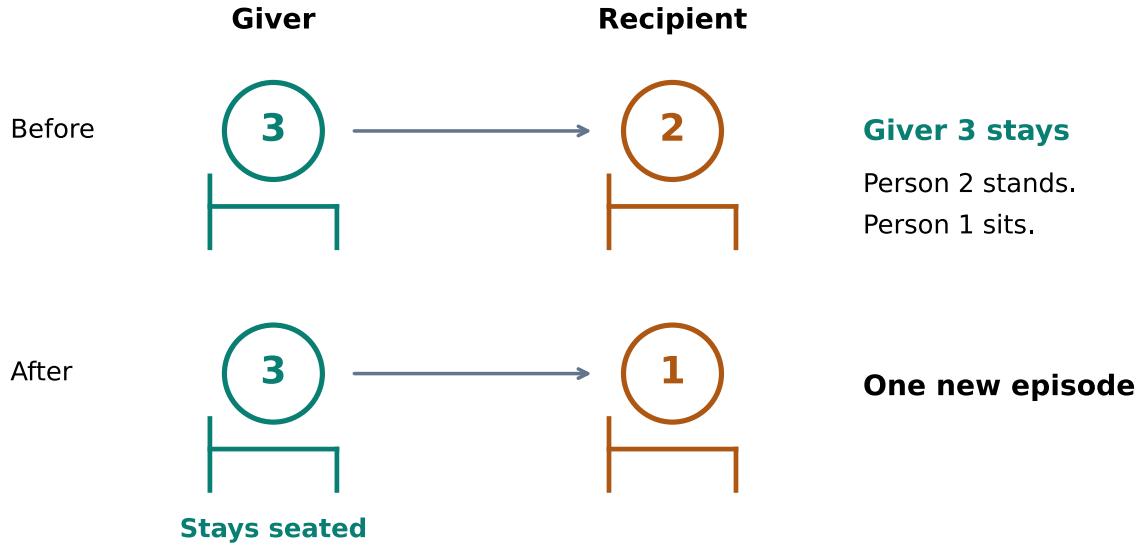


Figure 4: A boundary between recipient blocks: (3,2) is followed by (3,1). Giver 3 stays seated while recipient 2 is replaced by recipient 1. The full four-person schedule uses four giver episodes and three recipient episodes. This construction is optimal when recipient episodes are at least as costly as giver episodes.

The two constructions satisfy the lower bound for their respective cost ordering. They each enumerate all M required interactions in $O(M)$ time.

4.4 What the two-seat restriction permits

With $c_G = c_R = e$, either construction achieves

$$C^* = (M + 1)e.$$

This is also optimal when the two chairs have interchangeable roles. The first pair needs two seated people; every distinct subsequent pair needs at least one new person to sit. The construction attains that physical lower bound while keeping the chair roles fixed.

To compare with the original policy, charge its giver visits too. If every recipient stands after each tika, the eldest-first policy costs $\{M + (N - 1)\}e$. Reusing recipient 1 across its final two giver blocks saves one episode for $N \geq 3$, yielding $(M + N - 2)e$ for the same interaction order. The optimized order costs $(M + 1)e$.

For $N = 10$, these are **54**, **53**, and **46** total episodes, respectively. The two-seat restriction allows a modest improvement. The reduction to nine recipient episodes applies when giver movement has zero charged cost.

4.5 Everyone remains seated

With enough seats and access between every pair, all N people can sit at the start and stand at the end. Execute the M tikas sequentially while everyone stays in place. This costs Ne , which is optimal because each participant must be seated for at least one interaction. The arrangement assumes that distance and reach impose no extra standing requirements.

All four people sit once

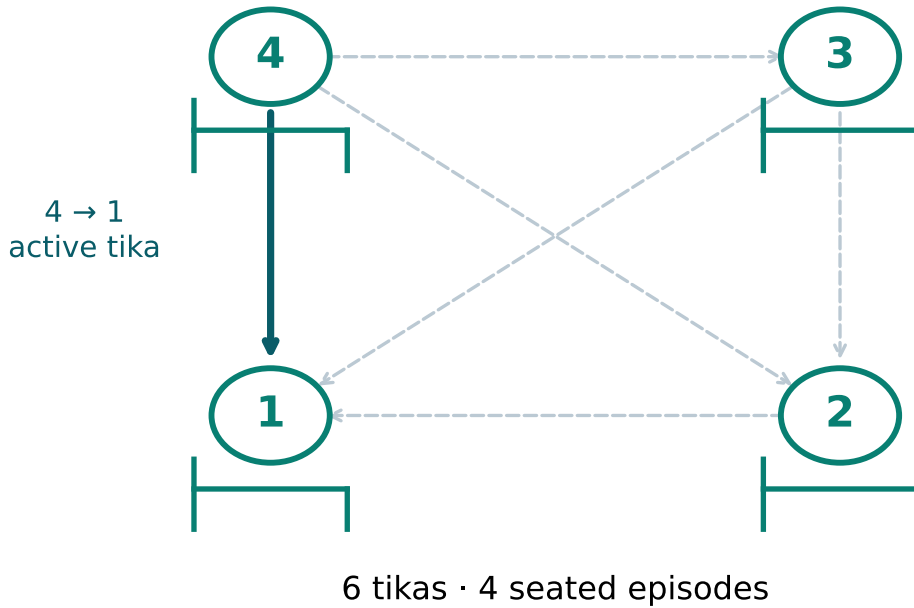


Figure 5: Four people remain seated for all six required interactions. Arrows mark older-to-younger dyads; the highlighted arrow shows a single active tika, 4 to 1. Each person has one seated episode, giving a total cost of 4e. This arrangement requires physical access between every pair.

5 A worked schedule

The cells below represent the same 15 interactions for six people. Numbers give their execution order. Giver blocks run along rows; recipient blocks run along columns. The altered order at block boundaries permits an occupant to remain seated.

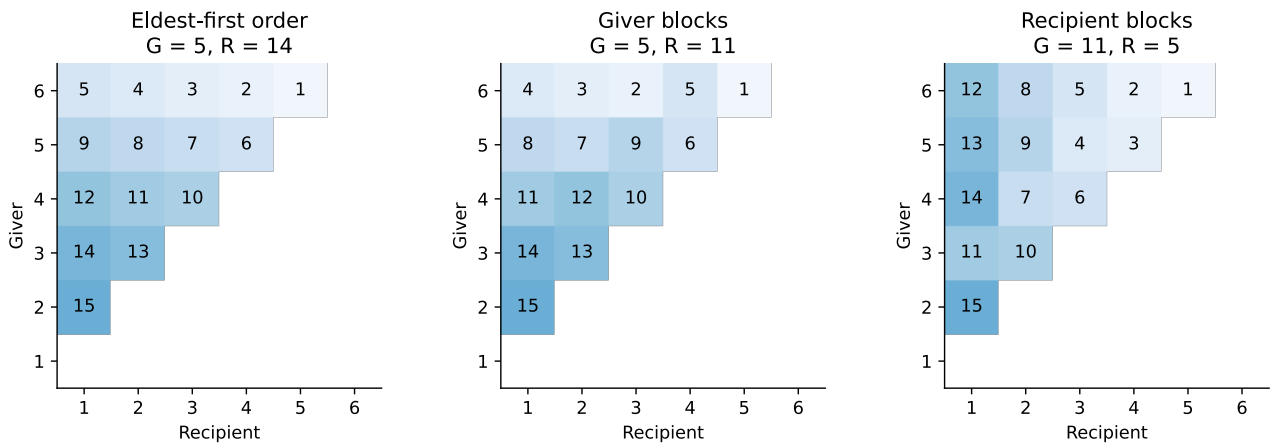


Figure 6: Interaction order for six people. G and R count giver and recipient episodes with consecutive reuse. The specified reset-after-each-tika baseline would add one recipient episode to the first panel.

6 Verification and scope

The optimization is a sequence-dependent setup problem. One can represent each required interaction as a vertex, connecting two vertices when their interactions share the giver or share the recipient. The block constructions give a Hamilton path in this graph, so every transition changes just one role. Age gives a triangular set of allowed giver-recipient pairs, which makes the path construction particularly simple.

The following exact subset dynamic program independently checks the optimum for small N . Its state is a set of completed interactions and the final interaction; it considers every possible next interaction. The general proof above applies to every $N \geq 2$.

20 exact optimization checks pass for $N = 2-5$; both constructions also pass coverage, cost, and age-precedence checks for $N = 2-30$.

Several extensions change the optimization problem:

- **Participant-specific discomfort.** In the recipient-only model, costs e_j give optimum $\sum_{j=1}^{N-1} e_j$: seat every recipient once. With costs on both sides, minimizing the number of episodes alone can miss an important distributional objective.
- **Fixed giver priority.** Eldest-first giver blocks are preserved by the giver construction. Recipient batching allows younger givers to act before the eldest has finished all their tikas. Both implemented schedules ensure that each person receives from every elder before starting to give.
- **Space and movement.** A single recipient seat, a two-seat station, and a fully seated group have different feasible schedules. Reaching and walking costs need to be included when those actions are consequential.
- **Parallel service or omitted dyads.** Multiple simultaneous tikas introduce resource conflicts; incomplete family interaction graphs can eliminate the simple triangular construction. Those are different scheduling problems.

For the complete sequential ceremony, the cost model has an exact solution: batch the expensive role and retain one participant across each block boundary. The required social interactions are unchanged.