

Supplementary Information

The Gravity of Love: Intrinsic Decay and Counteracting Effort in Romantic Relationships

Jorge Herrera de la Cruz¹, José-Manuel Rey^{2,3,*}

¹Department of Mathematics and Data Science, CEU San Pablo University, Madrid, Spain.

²Department of Psychology, Harvard University, Cambridge, MA, USA.

³Department of Economic Analysis, Complutense University of Madrid, Madrid, Spain.

Supplementary Note A. Computational solution and calibration

This supplementary note summarizes the computational procedures used to solve the dynamic game model and calibrate it to the observed marital satisfaction trajectory. The estimation framework consists of three main stages. First, the deterministic and stochastic versions of the relationship game are solved numerically using the RaBVIt-G algorithm. Second, the resulting equilibrium policies are used to generate model-implied trajectories of relationship satisfaction. Third, the Learning-by-Playing procedure iteratively updates the model parameters in order to minimize the discrepancy between simulated and observed trajectories.

A.1 Computing equilibrium relationship dynamics

The deterministic and stochastic versions of the dynamic game are solved numerically using the RaBVIt-G algorithm. The procedure iteratively computes feedback Nash equilibria and the associated value functions over a discretized state space.

Algorithm 1 Deterministic RaBVIt-G (d -RaBVIt-G)

Input: • State grid $\mathcal{X} = \{x^{(j)}\}_{j=1}^N \subset [x_{\min}, x_{\max}]$, step h ;

- Discount rates $\rho_1, \rho_2 > 0$;
- Utility functions $U_i(x)$, cost functions $D_i(c_i; c_i^*)$;
- Drift $f(x, c_1, c_2) = -rx + a_1c_1 + a_2c_2$;
- Effort bounds $[c_{\min}, c_{\max}]$, initial effort c^* ;
- Tolerances tol, tol_g , iteration caps $\text{itmax}, \text{itmax}_g$;
- Relaxation factor $\lambda \in (0, 1)$.

Output: Optimal feedback controls $c_i^\heartsuit(x)$ and value functions $v_i^\heartsuit(x)$, for $i = 1, 2$.

```

1: Initialisation:  $v_i^{(0)}(x) \leftarrow h U_i(x)$ ,  $c_i^{(0)}(x) \leftarrow c^*$  for all  $x \in \mathcal{X}$ ,  $i = 1, 2$ .
2: for  $t = 0, 1, \dots, \text{itmax}$  do
3:    $\Delta \leftarrow \text{tol} + 1$ 
4:   for each  $x \in \mathcal{X}$  do
5:      $u_1 \leftarrow c_1^{(t)}(x)$ ,  $u_2 \leftarrow c_2^{(t)}(x)$ . ▷ Inner game iteration
6:     for  $g = 1, \dots, \text{itmax}_g$  do
7:        $k_1 \leftarrow \arg \max_{c \in [c_{\min}, c_{\max}]} \{h[U_1(x) - D_1(c; c^*)] + (1 - \rho_1 h) \text{RBF}(x'; v_1^{(t)})\}$ ,
       where  $x' = \text{clip}(x + hf(x, c, u_2); x_{\min}, x_{\max})$ .
8:        $k_2 \leftarrow \arg \max_{c \in [c_{\min}, c_{\max}]} \{h[U_2(x) - D_2(c; c^*)] + (1 - \rho_2 h) \text{RBF}(x'; v_2^{(t)})\}$ ,
       where  $x' = \text{clip}(x + hf(x, u_1, c); x_{\min}, x_{\max})$ .
9:        $u_1 \leftarrow (1 - \lambda)u_1 + \lambda k_1$ ,  $u_2 \leftarrow (1 - \lambda)u_2 + \lambda k_2$ .
10:      if  $\max(|k_1 - u_1|, |k_2 - u_2|) < \text{tol}_g$  then break
11:      end if
12:    end for
13:     $c_i^{(t+1)}(x) \leftarrow u_i$ , for  $i = 1, 2$ .
14:    for  $i = 1, 2$  do ▷ Value iteration
15:       $x' \leftarrow \text{clip}(x + hf(x, c_1^{(t+1)}(x), c_2^{(t+1)}(x)); x_{\min}, x_{\max})$ .
16:       $\widehat{V}_i \leftarrow \text{RBF}(x'; v_i^{(t)})$ .
17:       $v_i^{(t+1)}(x) \leftarrow h[U_i(x) - D_i(c_i^{(t+1)}(x); c^*)] + (1 - \rho_i h) \widehat{V}_i$ .
18:    end for
19:     $\Delta \leftarrow \max\{\Delta, |c_1^{(t+1)}(x) - c_1^{(t)}(x)|, |c_2^{(t+1)}(x) - c_2^{(t)}(x)|\}$ .
20:  end for
21:  if  $\Delta < \text{tol}$  then break
22:  end if
23: end for
24: return  $c_i^\heartsuit = c_i^{(t+1)}$ ,  $v_i^\heartsuit = v_i^{(t+1)}$ , for  $i = 1, 2$ .

```

The stochastic version introduces a diffusion term $\sigma(x) \geq 0$ and approximates expectations using a symmetric two-point quadrature:

$$x'_\pm = \text{clip}(x + hf(x, c_1, c_2) \pm \sqrt{h} \sigma(x); x_{\min}, x_{\max}), \quad \mathbb{E}[V(x')] \approx \frac{1}{2}(\text{RBF}(x'_+; V) + \text{RBF}(x'_-; V)).$$

Algorithm 2 Stochastic RaBVIt-G (*s*-RaBVIt-G)

Input: Same as Algorithm 1, with diffusion function $\sigma(x) \geq 0$.

Output: Optimal policies $c_i^\heartsuit(x)$ and value functions $v_i^\heartsuit(x)$ on $\mathcal{X}$.

```
1: Initialisation: As in Algorithm 1.
2: for  $t = 0, 1, \dots, \text{itmax}$  do
3:    $\Delta \leftarrow \text{tol} + 1$ .
4:   for each  $x \in \mathcal{X}$  do
5:      $u_1 \leftarrow c_1^{(t)}(x), \quad u_2 \leftarrow c_2^{(t)}(x)$ . ▷ Inner stochastic game iteration
6:     for  $g = 1, \dots, \text{itmax}_g$  do
7:        $k_1 \leftarrow \arg \max_{c \in [c_{\min}, c_{\max}]} \left\{ h[U_1(x) - D_1(c; c^*)] + (1 - \rho_1 h) \widehat{V}_1(x; c, u_2) \right\}$ 
8:        $k_2 \leftarrow \arg \max_{c \in [c_{\min}, c_{\max}]} \left\{ h[U_2(x) - D_2(c; c^*)] + (1 - \rho_2 h) \widehat{V}_2(x; u_1, c) \right\}$ 
9:       where  $\widehat{V}_i(x; c_1, c_2) = \frac{1}{2}(\text{RBF}(x'_+; v_i^{(t)}) + \text{RBF}(x'_-; v_i^{(t)}))$ 
10:      and  $x'_\pm = \text{clip}(x + hf(x, c_1, c_2) \pm \sqrt{h} \sigma(x); x_{\min}, x_{\max})$ .
11:       $u_1 \leftarrow (1 - \lambda)u_1 + \lambda k_1, \quad u_2 \leftarrow (1 - \lambda)u_2 + \lambda k_2$ .
12:      if  $\max(|k_1 - u_1|, |k_2 - u_2|) < \text{tol}_g$  then break
13:      end if
14:    end for
15:     $c_1^{(t+1)}(x) \leftarrow u_1, \quad c_2^{(t+1)}(x) \leftarrow u_2$ . ▷ Feedback update
16:    for  $i = 1, 2$  do ▷ Value iteration
17:      Compute  $x'_\pm = \text{clip}(x + hf(x, c_1^{(t+1)}(x), c_2^{(t+1)}(x)) \pm \sqrt{h} \sigma(x); x_{\min}, x_{\max})$ .
18:      Compute expected continuation value:  $\widehat{V}_i = \frac{1}{2}(\text{RBF}(x'_+; v_i^{(t)}) + \text{RBF}(x'_-; v_i^{(t)}))$ .
19:      Update value function:  $v_i^{(t+1)}(x) \leftarrow h[U_i(x) - D_i(c_i^{(t+1)}(x); c^*)] + (1 - \rho_i h) \widehat{V}_i$ .
20:    end for
21:     $\Delta \leftarrow \max\{\Delta, |c_1^{(t+1)}(x) - c_1^{(t)}(x)|, |c_2^{(t+1)}(x) - c_2^{(t)}(x)|\}$ .
22:  end for
23:  if  $\Delta < \text{tol}$  then break
24:  end if
25: end for
26: return  $c_i^\heartsuit \leftarrow c_i^{(t+1)}, \quad v_i^\heartsuit \leftarrow v_i^{(t+1)}$  for  $i = 1, 2$ .
```

A.2 Trajectory simulation and scoring

Given equilibrium feedback controls, model-implied trajectories are generated at a monthly frequency and then annualized to match the observed data. Model fit is evaluated using the mean absolute error between observed and simulated annual trajectories.

Algorithm 3 Trajectory simulation (d -RaBVIt-G or s -RaBVIt-G)

Input: • Initial state x_0 ;

- Optimal feedbacks $S_1^\heartsuit, S_2^\heartsuit$;
- Time step $h = 1/12$ (monthly), total horizon T (in years);
- Drift $f(x, c_1, c_2) = -rx + a_1c_1 + a_2c_2$;
- Bounds $[x_{\min}, x_{\max}]$, diffusion $\sigma \geq 0$.

Output: Annual predictions $\hat{Y} = \{\hat{y}_t\}_{t=1}^T$.

- 1: Build RBF interpolants for $S_1^\heartsuit$ and $S_2^\heartsuit$ over $\mathcal{X}$.
 - 2: Set $x_1 \leftarrow x_0$.
 - 3: **for** $\tau = 1, \dots, 12T - 1$ **do**
 - 4: Evaluate controls: $c_{i,\tau} \leftarrow S_i^\heartsuit(x_\tau)$, for $i = 1, 2$.
 - 5: **If deterministic:** $x_{\tau+1} \leftarrow \text{clip}(x_\tau + h f(x_\tau, c_{1,\tau}, c_{2,\tau}); x_{\min}, x_{\max})$.
 - 6: **If stochastic:** $x_{\tau+1} \leftarrow \text{clip}(x_\tau + h f(x_\tau, c_{1,\tau}, c_{2,\tau}) + \sqrt{h} \sigma \xi_\tau; x_{\min}, x_{\max})$, $\xi_\tau \sim \mathcal{N}(0, 1)$.
 - 7: **end for**
 - 8: Annualise: $\hat{y}_t \leftarrow x_{1+12(t-1)}$, for $t = 1, \dots, T$.
 - 9: **return** $\hat{Y} = \{\hat{y}_t\}_{t=1}^T$.
-

Algorithm 4 Model scoring

Input: Observed trajectory $Y = \{y_t\}_{t=1}^T$, predicted trajectory $\hat{Y} = \{\hat{y}_t\}_{t=1}^T$.

Output: $\text{Err} = \sum_{t=1}^T |y_t - \hat{y}_t|$.

A.3 Learning by Playing calibration algorithm

Figure 2 of the main manuscript illustrates the overall Learning by Playing procedure. The present section provides the computational details of its implementation. Each iteration alternates between a *Playing* stage, in which RaBVIt-G computes the equilibrium trajectory under the current parameter configuration, and a *Learning* stage, in which the parameters are updated to improve the fit to the observed data. The detailed implementation is presented in Algorithm 5. The model parameters to be calibrated, together with the fixed discount factor ρ , are summarized in Supplementary Table 1.

As in previous applications of the RaBVIt-G framework, the Playing stage uses a time step of $h = 1/12$ and a relaxation factor of $\lambda = 0.05$ to smooth the updating of feedback policies during the search for equilibrium [1].

Rather than optimizing all parameters simultaneously, the Learning stage adopts a block-coordinate (Gauss–Seidel) strategy [2]. Given the equilibrium policies computed during the Playing stage, the parameters are updated sequentially while holding the remaining parameters fixed. Specifically, given the equilibrium feedback controls $(S_1^{(k)}, S_2^{(k)})$ computed by RaBVIt-G for $\theta^{(k)}$, each *Learning* iteration consists of two steps:

(a) Holding the equilibrium feedback controls, the grid-fixed parameters (A_i, β_i) , and the relationship-dynamics parameters fixed (r, a_1, a_2, σ) , the remaining utility and disutility parameters

(α_i, c_i^*) are re-optimized to improve the fit to the observed trajectory.

(b) Holding all other parameters fixed, the relationship-dynamics parameters (r, a_1, a_2, σ) are re-optimized against the observed trajectory.

Supplementary Table 1: Model parameters used in the calibration procedure. The parameter vector $P = (r, a_1, a_2, \sigma, \theta_1, \theta_2)$ consists of relationship-dynamics parameters and behavioral parameters. The discount factor ρ_i , included for completeness, is held fixed throughout.

Parameter	Description	Interpretation
<i>Relationship-dynamics parameters</i>		
r	Gravity of love	Intrinsic rate of relationship decline
a_i	Effort effectiveness	Effectiveness of partner i 's maintenance effort
σ	Volatility	Magnitude of stochastic fluctuations in relationship dynamics
<i>Behavioral parameters (partner i)</i>		
A_i	Utility scale	Overall contribution of relationship quality to well-being
α_i	Utility curvature	Degree of diminishing marginal utility of relationship quality
β_i	Effort sensitivity	Curvature of the effort-disutility function
c_i^*	Preferred effort level	Effort level minimizing effort-related dissatisfaction
$\rho_i = 0.1$	Discount factor	Individual time preference parameter governing the valuation of future well-being

The updated parameter vector is then passed to a new Playing stage, and the procedure is repeated until convergence. The final calibrated model is determined by the parameter vector that minimizes the discrepancy between the model-generated and observed marital-satisfaction trajectories.

Algorithm 5 Learning by Playing (deterministic or stochastic)

Input: Observed data $Y = \{y_t\}_{t=1}^T$; state grid $\mathcal{X}$; candidate grids $\mathcal{A}, \mathcal{B}$ for A_i, β_i ; initial parameter vector $\theta^{(0)}$; tolerance $\varepsilon > 0$.

Output: Estimated parameters $\hat{\theta}$, equilibrium policies $S_i^\heartsuit(x)$, and fitting error.

```
1: for  $A \in \mathcal{A}$  do
2:   for  $\beta \in \mathcal{B}$  do
3:     Initialise  $\theta^{(0)}$ , set  $k \leftarrow 0$ ,  $\Delta \leftarrow \infty$ .
4:     while  $\Delta > \varepsilon$  do
5:       Playing: Compute equilibrium policies  $(v^{(k)}, S_1^{(k)}, S_2^{(k)}) \leftarrow (d/s)\text{-RABVIT-G}(A, \beta; \theta^{(k)}, \mathcal{X})$ .
6:       Simulation: Generate predicted trajectory  $\hat{Y}^{(k)} = \text{SIMULATE}(S_1^{(k)}, S_2^{(k)}; \theta^{(k)})$ .
7:       Learning (block-coordinate update):
8:         (a) re-optimize  $(\alpha_i, c_i^*)$ , with  $(r, a_1, a_2, \sigma)$ , and the policies fixed;
9:         (b) re-optimize  $(r, a_1, a_2, \sigma)$  to minimize  $\text{Score}(Y, \hat{Y})$ , holding  $(\alpha_i, c_i^*)$  and the policies fixed;
10:        set  $\theta^{(k+1)}$  to the values produced by (a)–(b).
11:       Update:  $\Delta \leftarrow \|\theta^{(k+1)} - \theta^{(k)}\|$ ,  $k \leftarrow k + 1$ .
12:     end while
13:     Record  $(A, \beta, \hat{\theta}, \text{Score})$ .
14:   end for
15: end for
16: return best-performing configuration.
```

Supplementary Note B. Additional estimation results and robustness analyses

This supplementary note presents additional analyses of parameter identification and robustness. We examine the sensitivity of the calibration procedure to variations in parameter values, assess the stability of the estimated parameters under bootstrap perturbations of the observed trajectory, and evaluate the robustness of the estimated decay parameter r , which quantifies the gravity of love, under alternative behavioral specifications.

B.1 Parameter identification and estimation uncertainty

Supplementary Figure 1 reports the marginal error profiles associated with the estimated parameters. The dashed vertical lines indicate the parameter values selected by the calibration procedure. Overall, the deterministic and stochastic models display very similar patterns. Several parameters are identified within relatively broad regions of low error, indicating that multiple nearby parameter configurations provide comparably good fits to the observed marital satisfaction trajectory. This feature is consistent with the heterogeneity analyses reported in the main text, which show that different parameter combinations may generate similar relationship trajectories.

A notable exception is the decay parameter r . For both the deterministic and stochastic models, the marginal error profile of r exhibits a well-defined convex shape with a minimum at the selected

estimate. This contrasts with the flatter profiles observed for several other parameters and indicates that the data provide comparatively strong information about the magnitude of the underlying decay process acting on relationship satisfaction.

By comparison, the broadest low-error region is observed for the parameter β_i , suggesting that the available data contain limited information for distinguishing among alternative specifications of effort-related costs. Similarly, the stochasticity parameter σ is only weakly identified, reflecting the fact that aggregate satisfaction trajectories provide relatively little information about the magnitude of underlying emotional fluctuations.

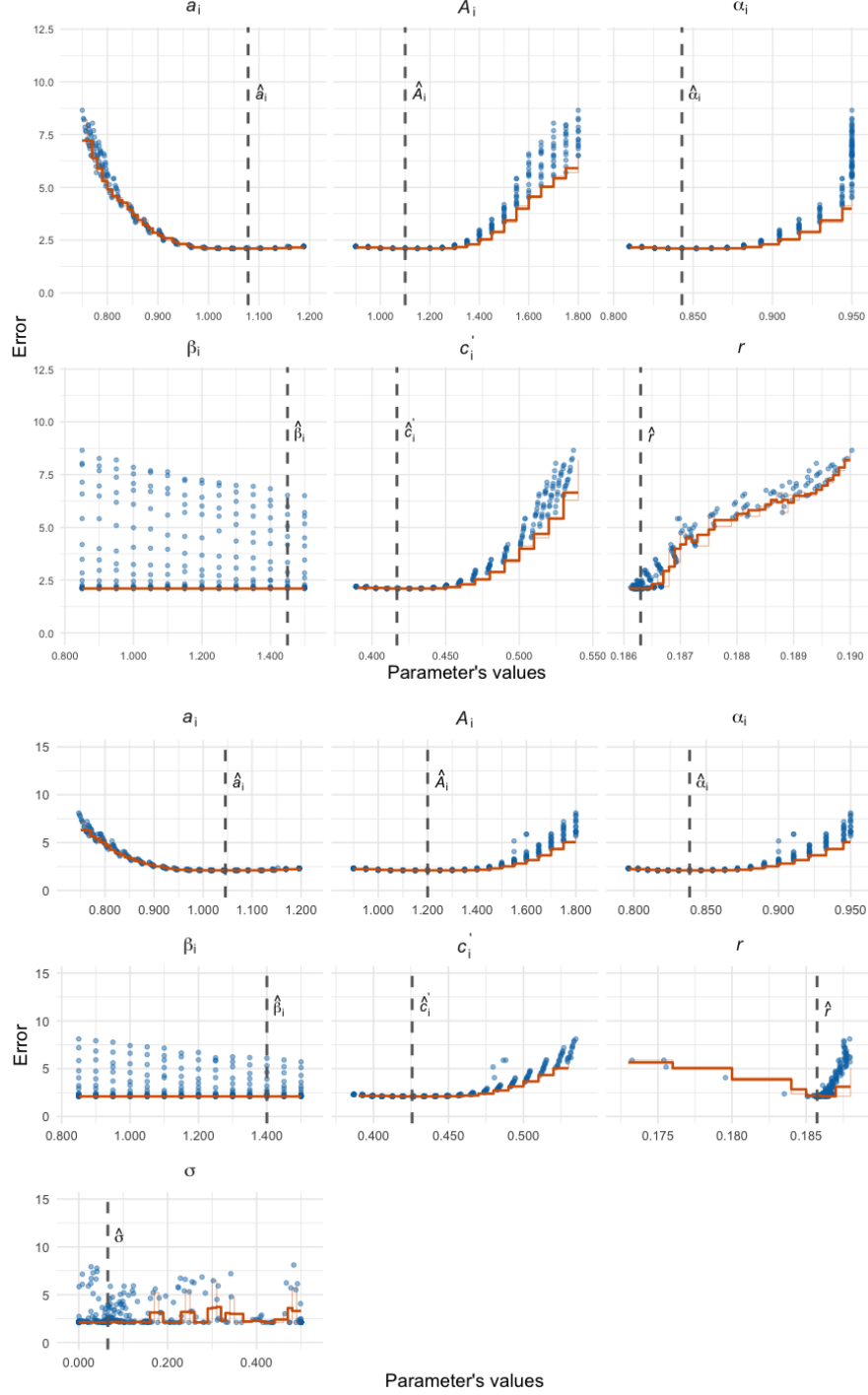
The analysis indicates that, although some individual parameters are only partially identifiable, the estimate of r is comparatively well determined by the data and remains a robust feature of the calibration.

B.2 Bootstrap stability analysis

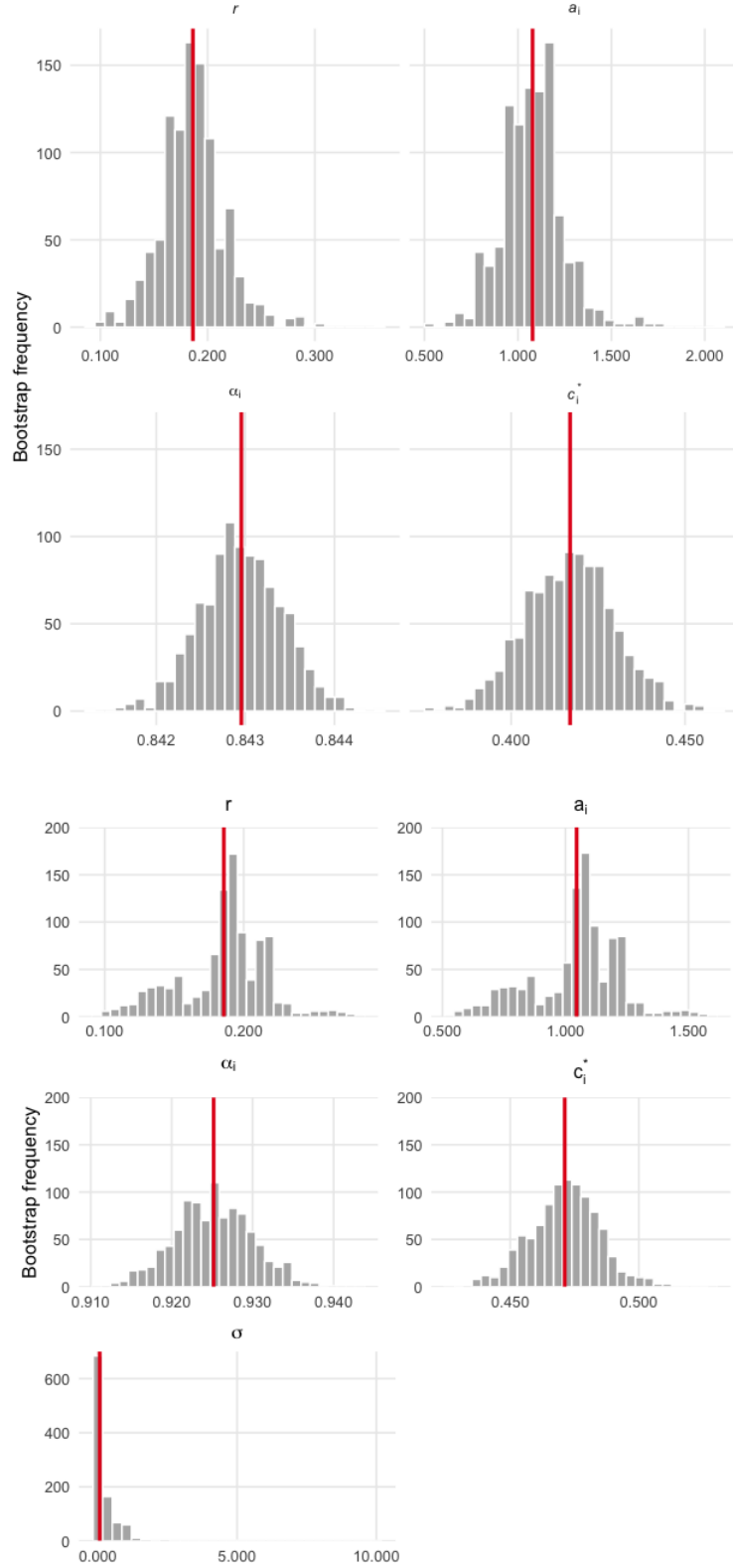
Supplementary Figure 2 reports the results of 1,000 bootstrap replications of the calibration procedure. The bootstrap samples are generated by perturbing the fitted feeling trajectory using residuals from the baseline estimation and re-running the Learning by Playing algorithm on each synthetic perturbed trajectory. The purpose of this exercise is not to provide formal inferential statements, but rather to assess the stability of the estimated parameters with respect to plausible perturbations of the observed marital satisfaction trajectory.

Because the estimation procedure is explicitly conditioned on fixed values of (A_i, β_i) selected on a predefined grid, bootstrap distributions cannot be recovered for these parameters. The analysis therefore focuses on the remaining estimated parameters, namely $(r, a_i, \alpha_i, c_i^*)$ and, in the stochastic specification, the diffusion parameter σ .

The resulting distributions reveal a heterogeneous degree of dispersion across parameters. The estimates of a_i display a comparatively larger spread across bootstrap replications, indicating greater sensitivity to data perturbations, whereas the distributions of α_i and c_i^* are substantially more concentrated, suggesting stronger identification. Particular attention should be paid to the bootstrap distribution of r . Despite the variability induced by the resampling procedure, the estimated values remain tightly concentrated around the baseline estimate reported in Table 1 of the main manuscript. This result provides additional evidence that the characteristic timescale of relationship decline identified in the main analysis is a robust feature of the calibration.



Supplementary Figure 1: Parameter identification and marginal error profiles. The first two rows (six panels) report the marginal ℓ_1 calibration error for the parameters of the deterministic model, whereas the three lower rows (seven panels) show the corresponding profiles for the stochastic specification, including the diffusion parameter σ . Each point represents a parameter configuration evaluated during the calibration procedure, and the dashed vertical lines indicate the parameter values selected by the Learning by Playing algorithm. Sharper minima indicate stronger parameter identification. In both specifications, the decay parameter r exhibits a pronounced convex profile with a minimum at the selected estimate, indicating that its value is comparatively well determined by the data. By contrast, broader low-error regions are observed for β_i and, in the stochastic model, for σ .



Supplementary Figure 2: Stability of parameter estimates under bootstrap perturbations. Histograms of parameter estimates obtained from 1,000 bootstrap replications of the Learning-by-Playing calibration procedure applied to perturbed versions of the observed marital-satisfaction trajectory.

B.3 Robustness to alternative behavioral specifications

To assess the robustness of the estimate of r , we repeated the calibration procedure using the utility and disutility functions proposed by Goudon and Lafitte [3]. Supplementary Table 2 reports the resulting estimates.

Supplementary Table 2: Robustness of the estimate of r under alternative utility and disutility specifications. The table reports the estimated decay coefficient r , volatility σ , correlation ρ between observed and fitted trajectories, and the implied half-life $\ln(2)/r$ of relationship satisfaction. In contrast to the specification adopted in the main text, where the behavioral parameter vector is $\theta_i = (A_i, \alpha_i, \beta_i, c_i^*)$, here we use the utility and disutility functions proposed by [3], $U_i(x) = A \ln(x + 1)$ and $D_i(c_i) = \frac{1}{2}(c_i - c^*)^2$, so that a reduced parameterization involving only two behavioral parameters, (A, c^*) , is estimated in addition to (r, a_1, a_2, σ) .

Utility specification	Model	r	σ	Corr.	Half-life (years)
Present study	Deterministic	0.186	0.00	0.8150	3.73
Present study	Stochastic	0.186	0.065	0.8148	3.73
Goudon and Lafitte [3]	Deterministic	0.187	0.00	0.8122	3.70
Goudon and Lafitte [3]	Stochastic	0.188	0.46	0.8119	3.68

The estimated half-life remains remarkably stable across specifications, around 3.7 years, despite substantial differences in the functional forms and parameterizations of the utility and disutility functions.

References

- [1] J. Herrera de la Cruz and J.-M. Rey. Controlling forever love. *PLoS ONE*, 16(12):e0260529, 2021.
- [2] J. Nocedal and S. J. Wright. *Numerical Optimization*. Springer, New York, 2 edition, 2006.
- [3] T. Goudon and P. Lafitte. The lovebirds problem: why solve Hamilton–Jacobi–Bellman equations matters in love affairs. *Acta Appl. Math.*, 136(1):147–165, 2015.