Computational Modeling of Yeast Telomere Dynamics: Program Description

1) Biological model

1.1 Chromosome ends

- Cells have **32 chromosome ends** (N_ENDS=32).
- Each end has (in order from the telomere): **Telomere length** (int), **X** (one per end), **Y elements** (0...cap), and **C** (one per end).

1.2 Cell division, telomere erosion, senescence and recombination dynamics

- A cell is **senescent** if **any telomere < Ls**. A cell can divide if **all 32 telomeres ≥ Ls** (non-senescent).
- During a PD, the model builds the next generation by:
    - **Copying each existing cell once** into the next pool (N cells).
    - Repeatedly selecting **one parent** at a time from the current pool (≥N) to produce **one new child** (if the chosen parent is non-senescent), until pool size reaches 2N.
    - On **division** (non-senescent cells), both the parent and the daughter independently erode (lose) telomeric DNA at an average rate of --del-rate (nucleotides per telomere per division). To follow DNA replication biology, random half the ends erode via Poisson(2 × del_rate) to ensure per-end average deletion rate.
    - If the chosen parent cell is **senescent**, it will either:
        - **Die** with probability --p-sen-death, and is removed.
        - **Attempt recombination** (if --rec-model = 1) to repair its short ends. Recombination is applied one chromosome end with <Ls at a time.
- **Recombination breakpoint on the receptor end** (telomeres <Ls):
    - An end has 3 constant elements (**Tel**, **X**, **C**) plus **Y’ count** copies of Y.
    - Resection in short telomeres (<Ls) is assumed to be long, beyond the telomere repeat. The recombination breakpoint is chosen **uniformly** among these elements (probability = 1 / (3 + Y’ count)).
- **Donor chromosome end eligibility** is set by --rec-model:
    - 0 = off (no recombination)
    - 1 = recombination can occur in senescent cells. **Only chromosome ends with telomeres ≥ Ls in the same cell** can be donor; if none exist in this cell, no recombination.

- **Y’ recombination** when a receptor Y’ element becomes “recombinant breakpoint”. Donor chromosome end is either random (among those ≥ Ls) or optionally **weighted by donor chromosome end Y’ count** (if --rec-y-weighted flag is present). If donor has multiple Y’s, one is chosen at random. Recombination between Y’ receptor and Y’ donor also causes receptor chromosome end to capture all **donor terminal Y’ elements** (if any) **and donor telomere length**.
- **X and C recombination** when a receptor X or C element become “recombinant breakpoint”. Donor chromosome end is chosen at random (among those ≥ Ls in the same cell). Recombination at X causes receptor to capture **donor telomere length**. Recombination at C causes receptor to capture donor **Y’s, X and telomere length**.
- **Telomeric recombination (HR)**
    - **Donor telomere choice**: --donor-mode
        - 0 = **uniform** among eligible **donors** (>Ls)
        - 1 = weighted by **telomere length** among eligible **donors**
        - 2 = **max** telomere length among eligible **donors**
    - **Telomeric recombination modes** (--rec-tel-mode):

      Once the donor telomere is identified, recombination between telomeres can cause:
        - copy: receptor telomere (TelR’) becomes donor **TelD** (assumes recombination at the base of the telomere repeats; default).
        - rnd: receptor telomere becomes **TelR' = rR + (TelD - rD)** with random cut points on receptor telomere (rR) and donor telomere (rD).
        - end: like rnd but **rR = TelR** (receptor telomere uses its terminal end).
    - **Template switching (ALT)**. After telomeric HR, the receptor telomere (with its new length) can recombine again with a new telomere donor (template switching) with probability --prob-ts. Additional donor telomere choice follows --donor-mode. Receptor always uses --rec-tel-mode **end** for these secondary events. Successful ‘jumps’ allow further jumps (each with --prob-ts), up to 5.
    - **Circles (ALT: extrachromosomal templates)**
        - **Static circles**. --prob-circle gives a **constant per-cell** probability that a short telomere uses a t-circle instead of a chromosomal donor when chosen for recombination. If a t-circle is chosen as donor, it **adds** circle_len nucleotides to that telomere length.
        - **Dynamic circles.** When the --dynamic-circles flag is used, each cell carries its **own** prob-circle value (starts at 0).
            - Every time a resected telomere (<Ls) generates length ≥ --min-len-circle-generation by HR or HR+Template switching, it **adds** --prob-each-circle to this cell’s circle probability (prob-circle, clipped to 1).

  - On **division (post-senescent; all telomeres >Ls)**, the cell’s prob_circle is **split 50/50** between parent and daughter.

1.3 Cell passages and Limits to cell number

- The model can apply **subsampling** at specific PDs (--subsample-pds, --subsample-size) to mimic experimental cell passages and keep a manageable number of cells.
- The model applies **hard subsampling** (--hard-threshold, --hard-keep-fraction) to prevent **the number of cells from becoming too high**. --hard-threshold indicates the max number of cells before the next PD, and --hard-keep-fraction the fraction of cells to keep in the population. Note that the number of cells is determined before starting a round of PD; therefore, -hard-threshold 512000 implies a maximum number of cells of ~1 million.

2) Simulation phases

2.1 Initialization: Telomere lengths and Y’ number and distribution in original cell

- Telomere length across chromosome ends generated from --init-len (Poisson around mean init-len; clipped to ≥Ls) or from a file with specific lengths (--init-len-file, with 32 lines). Default --init-len 225.
- Number and distribution of Y’ elements across chromosome ends generated from the total number of Y’s in the cell --init-Ys (multinomial distributed across ends) or from a file with specific distribution (--init-Y-file, 32 lines). Default --init-Ys 0.
- Y counts per chromosome end are capped by --max-Ys (default 50).

2.2 Pre-evolution

- To add heterogeneity among cells and chromosome ends, exponential growth is applied for user-defined PDs (--pd-pre; default 10)
- In each pre-PD, cell and chromosome end dynamics as described in 1.2 (Cell division, telomere erosion, senescence and recombination dynamics) with the difference that telomeres in parent and children cells get symmetric changes in length, ±Poisson(del_rate) per end (50/50), clipped to ≥Ls (no senescent cells are generated).
- Applies **hard subsampling** if pertinent (see 1.3).

2.3 Replicates

- The same pre-evolved population is used for multiple replicates.
- From the pre-evolved cell pool, an **independent** sample of --init-n-cells (randomly chosen without replacement) per replicate is used to seed the main population.

2.4 Main evolution

- Starts with init-n-cells and follows exponential growth for –pd-max PDs.

- If --prob-circle > 0, all seeding cells and all their daughter cells are assigned prob_circle. If the --dynamic-circles flag is present, all seeding cells are assigned --prob-circle =0.
- In each PD, cell and chromosome end dynamics as described in 1.2 (Cell division, telomere erosion, senescence and recombination dynamics).
- Applies **subsampling** and **hard subsampling** if pertinent (see 1.3).

2.5 Stopping criteria

- If at any point, no dividing cells remain in the population, replicate stops early (before pd_max).
- If at any PD the fraction of senescent cells ≥ --max-freq-senesc (default 0.9999), the replicate stops and is recorded as fully senescent (fraction of senescent cells =1.0).

3) Implementation

Dependencies: numba ≥ 0.60, numpy ≥ 1.26. Pinning these versions is recommended for reproducible performance and JIT behavior.

JIT hot loops: Core population-doubling kernels are compiled with Numba in nopython mode. The first call triggers a short JIT warm-up; subsequent PDs run at native speed.

4) Command-line interface (key flags)

python YeastTelDynamics.V01.py

--num-replicates	# number of replicates (default 10)
--del-rate	# mean erosion rate per telomere per division (bp) (default 6)
--Ls	# telomere length senescence threshold (default 60)
--pd-max	# number of PDs (default 60)
--pd-pre	# number of PS in pre-evolution (seeding variance in telomere length) (default 10)
--init-len	# mean telomere length (bp) in initial cell (default 225), or
--init-len-file	# file with telomere length per chromosome end (32 lines)
--init-Ys	# total number of Y’ elements in initial cell (default 40), or
--init-Y-file	# file with number of Y’s per chromosome end (32 lines)
--init-n-cells	# initial cells drawn from pre-evolved population per replicate (default 10)
--hard-threshold	# maximum number of cells before next PD (default 512000; limiting cell number to <1,024,000)
--hard-keep-fraction	# fraction of cells to keep when cells reach –hard-threshold (default 0.1)
--max-freq-senesc	# early stop once fraction of senescent cells ≥ --max-freq-senesc (default 0.9999)
--subsample-pds	# PDs at which passages are applied (default 20 30 40 50 60)

--subsample-size	# number of cells subsampled in passages (default 10000)
--p-sen-death	# probability senescent cell dies when selected (default 0.1)
--max-Ys	# cap on Y’ elements per chromosome end (default 50)
--rec-model	# 0= no recombination; 1=recombination, donor must be ≥Ls (default 1)
--donor-mode	# telomere donors, 0=uniform, 1=weighted by length, 2=max length (default 2)
--rec-tel-mode	# mode of recombination between telomeres (copy|rnd|end) (default copy)
--rec-y-weighted	# (optional) chromosome end Y’ donors weighted by their Y’ count
--prob-ts	# probability of template switch per recombination event at telomeres (default 0)
--prob-circle	# static probability of recombination between a telomere and t-circle, (ignored if --dynamic-circles is present) (default 0)
--circle-len	# nucleotides added to a telomere if recombination between a telomere and a t-circle (default 2000)
--dynamic-circles	# (optional) enables per-cell circle probability dynamics
--min-len-circle-generation	# length in extended resected telomeres that enables generation of t-circles (default 120)
--prob-each-circle	# increment to per-cell prob-circle upon rescue (default 0.001)
--out-prefix	# output files prefix
--seed	# (optional) allows reproducing results (--auto-seed is on by default).

5) Outputs

Two files are written with --out-prefix (*.summary.tsv and *.replicates.tsv):

***.summary.tsv** — mean parameters across replicates at each PD

First lines include program, version, command, and wall time (seconds).

Columns:

- PD
- Number of cells
- Fraction of senescent cells
- Telomere length parameters (mean, 5%, 10%, 50%, 90%, 95%)
- Y’ number parameters (mean, 5%, 10%, 50%, 90%, 95%)

***.replicates.tsv** — results for each replicate; allows investigating variances between replicates

For each metric above, a block with rows = PD (0..max), columns = Rep1..RepK.

6) Example workflows

6.1 Erosion and Senescence without Recombination

> python YeastTelDynamics.V01.py --rec-model 0 --pd-max 50 --out-prefix YTelDynamics_norec

Defaults applicable to no recombination: --auto-seed; --num-replicates 10; --init-len 225; --del-rate 6; --Ls 60; --pd-pre 10; --init-n-cells 10; --hard-threshold 512000; --hard-keep-fraction 0.1; --subsample-pds 20 30 40; --subsample-size 10000; --p-sen-death 0.1; --max-freq-senesc 0.9999

6.2 Erosion, Senescence and HR Recombination

> python YeastTelDynamics.V01.py --rec-model 1 --init-len 200 --pd-max 60 --init-Ys 40 --rec-y-weighted --out-prefix YTelDynamics_rec

Defaults applicable to HR recombination: -auto-seed; --del-rate 6 --Ls 60 --donor-mode 2 --rec-tel-mode copy --circle-len 2000 --prob-ts 0 --p-sen-death 0.1 --subsample-pds 20 30 40 --subsample-size 10000 --hard-threshold 512000 --hard-keep-fraction 0.1 --init-n-cells 10

6.3 Erosion, Senescence and HR Recombination with t-circles present (ALT-like Type II survivors)

> python YeastTelDynamics.V01.py --rec-model 1 --init-len 200 --pd-max 70 --init-Ys 40 --rec-y-weighted --prob-circle 0.01 --subsample-pds 20 30 40 50 --out-prefix YTelDynamics_rec_circles

Defaults applicable to HR recombination with t-circles present: -auto-seed; --del-rate 6 --Ls 60 --donor-mode 2 --rec-tel-mode copy --circle-len 2000 --prob-ts 0 --p-sen-death 0.1 --subsample-size 10000 --hard-threshold 512000 --hard-keep-fraction 0.1 --init-n-cells 10 --circle-len 2000

7) Reproducibility

- --seed allows fixing global RNG. Otherwise --auto-seed is on by default (runs will differ).
- The *.summary.tsv output shows the command line to allow reproducing results.

8) Performance tips

- Use --hard-threshold, --hard-keep-fraction, --subsample-pds and --subsample-size to limit number of cells and control memory/time.
- Set a sensible --max-freq-senesc (e.g., 0.999, 0.9999), particularly when studying very large populations.

9) Citation / acknowledgement

If you use this simulator in a publication, please cite the original article (Tsai et al. 2026) and this github repository.