Mathematical Overview: Multiple Outcomes and Chained Races

AccumulatR notes

February 8, 2026

1 Goal

This note summarizes what must be added mathematically to extend the current race framework to:

1. multiple outcomes per trial (generation and likelihood), and
2. chained races where later races start when earlier races finish, with possibly unobserved intermediate outcomes.

2 Current single-response race model

2.1 Accumulator-level timing

For accumulator i, define a defective finishing time

$$T_i = \begin{cases} o_i + t0_i + X_i, & \text{with probability } 1 - q_i, \\ \infty, & \text{with probability } q_i, \end{cases}$$

where X_i has base density/CDF (g_i, G_i) on $(0, \infty)$.

Hence

$$f_i(t) = (1 - q_i)\, g_i(t - o_i - t0_i)\, \mathbf{1}\{t \geq o_i + t0_i\},$$
$$S_i(t) = q_i + (1 - q_i)\left[1 - G_i(t - o_i - t0_i)\right].$$

2.2 Expression-level events

Let E_j denote expression events built from accumulators/pools/guards.

OR / first-of If $E = \bigvee_{j=1}^{m} E_j$ (first completion among independent branches),

$$f_E(t) = \sum_{j=1}^{m} f_{E_j}(t) \prod_{k \neq j} S_{E_k}(t), \qquad S_E(t) = \prod_{j=1}^{m} S_{E_j}(t).$$

AND / all-of If $E = \bigwedge_{j=1}^{m} E_j$ (all finished by t),

$$F_E(t) = \prod_{j=1}^{m} F_{E_j}(t), \qquad f_E(t) = \sum_{j=1}^{m} f_{E_j}(t) \prod_{k \neq j} F_{E_k}(t).$$

Guard For $E = \text{guard}(\text{blocker} = B, \text{reference} = R)$,

$$f_E(t) = f_R(t)\, S_B^{\text{eff}}(t), \qquad F_E(t) = \int_0^t f_R(u)\, S_B^{\text{eff}}(u)\, du.$$

k-of-n pool If $N(t) = \sum_{i=1}^{n} \mathbf{1}\{T_i \le t\}$, then for the k-th order event:

$$S_{k:n}(t) = \mathbb{P}(N(t) < k), \qquad f_{k:n}(t) = \frac{d}{dt}\left[1 - S_{k:n}(t)\right].$$

2.3 Current single observed outcome

With observed $(R = r,\ rt = t)$ and competitor set $C(r)$:

$$p(r,t) = d_r(t), \quad d_r(t) = f_{\nu(r)}(t) \prod_{c \in C(r)} S_{\nu(c)}(t).$$

With component mixture m and weights w_m:

$$p(r,t) = \sum_m w_m\, d_{r,m}(t).$$

This is the current one-event-per-trial target.

3 A. Multiple outcomes

3.1 A1. Multiple labels for one latent event

If one latent event e is mapped to a vector label y via map h:

$$p(Y = y,\ t) = \sum_{e:\ h(e)=y} d_e(t).$$

This is a latent-label summation only (no extra time convolution).

3.2 A2. Multiple observed event outcomes within trial

Suppose trial observation is an ordered sequence

$$(r_1, t_1), \ldots, (r_K, t_K), \qquad 0 < t_1 < \cdots < t_K.$$

Let $\mathcal{H}_{k-1}$ be the history/forced-state after the first $k-1$ events. Define conditional event hazard term:

$$\lambda_{r_k}(t_k \mid \mathcal{H}_{k-1}) = f_{r_k}(t_k \mid \mathcal{H}_{k-1}) \prod_{c \in C_{r_k}(\mathcal{H}_{k-1})} S_c(t_k \mid \mathcal{H}_{k-1}).$$

Then

$$p(r_{1:K}, t_{1:K}) = \mathbf{1}\{t_1 < \cdots < t_K\} \prod_{k=1}^{K} \lambda_{r_k}(t_k \mid \mathcal{H}_{k-1}).$$

If times are missing/latent:

$$p(r_{1:K}) = \int_{0<t_1<\cdots<t_K<\infty} \prod_{k=1}^{K} \lambda_{r_k}(t_k \mid \mathcal{H}_{k-1})\, dt_{1:K}.$$

If intermediate outcomes are latent too, also sum over latent labels.

Key new object The core new integral for multi-outcome likelihood is an ordered-simplex integral over latent event times.

4 B. Chained races

4.1 Two-stage chain

Let stage 1 output (Z_1, T_1). Stage 2 starts at

$$B_2 = T_1 + \delta(Z_1),$$

and has relative finish U_2, absolute finish $T_2 = B_2 + U_2$.

If both stages observed:

$$p(z_1, t_1, z_2, t_2) = d^{(1)}_{z_1}(t_1)\, d^{(2)}_{z_2|z_1}(t_2 - t_1 - \delta(z_1))\, \mathbf{1}\{t_2 > t_1 + \delta(z_1)\}.$$

If stage 1 is unobserved:

$$p(z_2, t_2) = \sum_{z_1} \int_0^{t_2 - \delta(z_1)} d^{(1)}_{z_1}(u)\, d^{(2)}_{z_2|z_1}(t_2 - u - \delta(z_1))\, du.$$

This is a sum of convolutions over hidden stage-1 time.

4.2 Many chained stages

Use forward recursion (semi-Markov style):

$$\alpha_s(z, t) = \sum_{z'} \int_0^t \alpha_{s-1}(z', u)\, k_s(z, t \mid z', u)\, du,$$

where k_s is the stage-s conditional race kernel.

5 Likelihood framework placement

Your low-level event mathematics can remain mostly unchanged. The main extension is at trial likelihood composition:

- current: one (R, rt) contribution per trial;
- extended: event-sequence contribution, with product form when fully observed, and forward integral/sum recursion when partially observed.

6 Computational implications

Case	New math cost	Efficient strategy
All event times observed	Mostly repeated densities/survivals	cache node evals by state/time
Hidden event times	ordered simplex integrals	forward recursion + 1D quadrature
Hidden chained stage(s)	convolutions	adaptive 1D quadrature or FFT on grid
Many latent branches	large sums of tiny terms	log-sum-exp accumulation

Practical recommendation For small/moderate latent dimension, adaptive Gauss-Kronrod recursion is natural. For repeated convolutions on fixed grids, FFT methods can reduce cost from $O(N^2)$ to $O(N \log N)$ per convolution.

7 Takeaway

Mathematically, both requested extensions require replacing the single-event trial target with a *path probability* over event histories:

(products of conditional hazards) $\times$ (integrals over latent times) $\times$ (sums over latent outcomes).

Your current event/pool/guard primitives are already close to what this requires; the major change is trial-level composition and inference recursion.