

QCAD Recursive Math Model Simplified Explanation

Author: Travis Raymond-Charlie Stone

Date: July 10, 2025

What This Model Describes (In Simple Terms):

You're building a system that tracks how something grows, changes, or splits over time not just once, but recursively, layer by layer, based on patterns that emerge and evolve.

1. Starting Point and Growth:

$$x(t) = x * e^{\lambda t}$$

This means:

- Start with something (a value, a signal, a state).
- Over time, it either grows or shrinks depending on a factor called lambda (λ).
- If λ is positive, it grows. If negative, it shrinks.
- Like how bacteria multiply or how a rumor spreads (or fades).

2. Recursive Weighting:

sum of $e^{-k * P(x)}$

This means:

- As you go deeper (layer k), things get weaker or stronger depending on a probability function $P(x)$.
- Think of this as a filter: "How much should this layer matter?"
- It's like weighing older memories or levels of trust.

3. The Main Engine (Q):

$Q(x, t) = (\text{stuff})^{\alpha(t)}$

"Stuff" includes:

- How quickly things are changing (time derivative)
- How much the system bends or curves (spatial structure)
- How strong each piece is (weights)
- All of this is raised to a power that changes over time $\alpha(t)$

So it's like:

Take the changes, the shapes, and the strengths, and raise it to a time-sensitive power.

4. Why It Matters:

You're not just looking at motion. You're watching how the whole pattern builds recursively:

- It grows, bends, learns, and reacts to itself.
- It's intelligent structure behavior, not just numbers.

This describes:

- How thoughts grow
- How ecosystems evolve
- How societies change
- Or how a recursive AI might think

Summary (One Sentence):

This math describes a system that grows or collapses recursively, where every layer reacts to the last, influenced by time, space, probability, and feedback all wrapped in a structure that learns and changes shape over time.