Real-time Physical Material Aging and Weathering Simulation

TNM084 Project

Shaoxuan Yin

Linköping University

January 17, 2025

Project Overview

- Initial goal: Create a real-time material aging simulation
- Focus on:
  - Procedural generation of aging effects
  - Real-time performance
  - Interactive parameter control
- Original plan: Implement in Vulkan
- Final implementation: OpenGL Compute Shaders

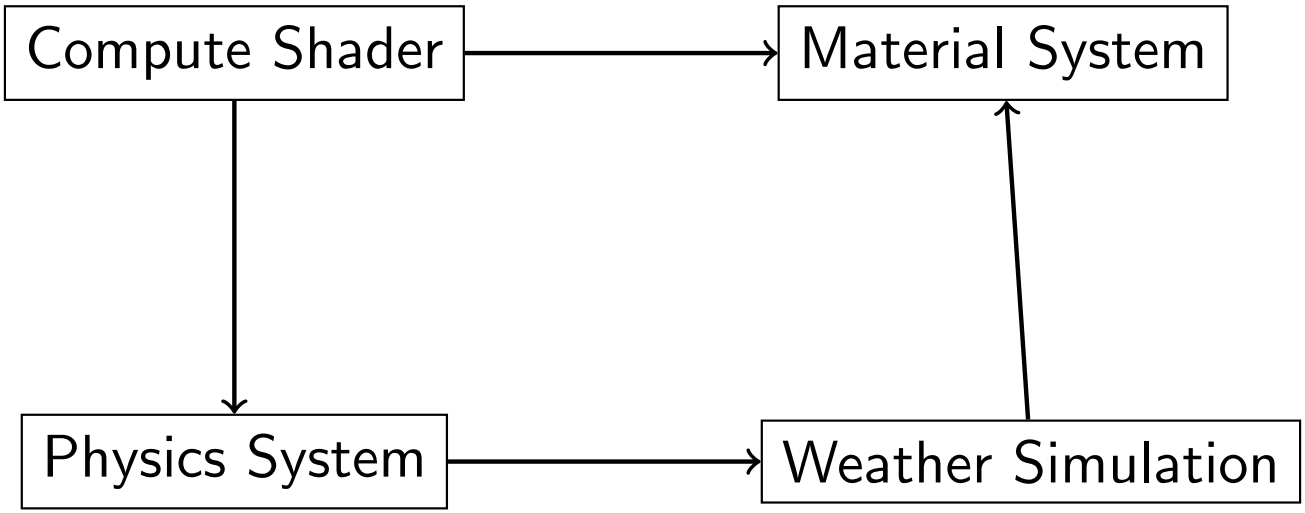
Initial Approach: Vulkan

- Attempted Vulkan implementation
- Challenges faced:
  - Complex memory management
  - Buffer synchronization issues
  - Multiple object handling complications
- Decision to pivot after 13,000 lines of code

Successful Pivot: OpenGL

- Switched to OpenGL Compute Shaders
- Benefits:
  - Faster development cycle
  - Easier resource management
  - Better debugging capabilities
- Maintained original project goals

System Architecture



- Real-time ray tracing with compute shaders
- Physically-based material aging simulation
- Dynamic weather influences material degradation
- Interactive physics for realistic object behavior

Rust Formation Process - Noise Functions

- Base Hash Function:

$$hash(p) = fract(p \cdot vec3(443.897, 397.297, 491.187))$$

- 3D Value Noise:

$$noise(p) = lerp(lerp(lerp(h_{000}, h_{100}, f_x), lerp(h_{010}, h_{110}, f_x), f_y),$$

$$lerp(lerp(h_{001}, h_{101}, f_x), lerp(h_{011}, h_{111}, f_x), f_y), f_z)$$

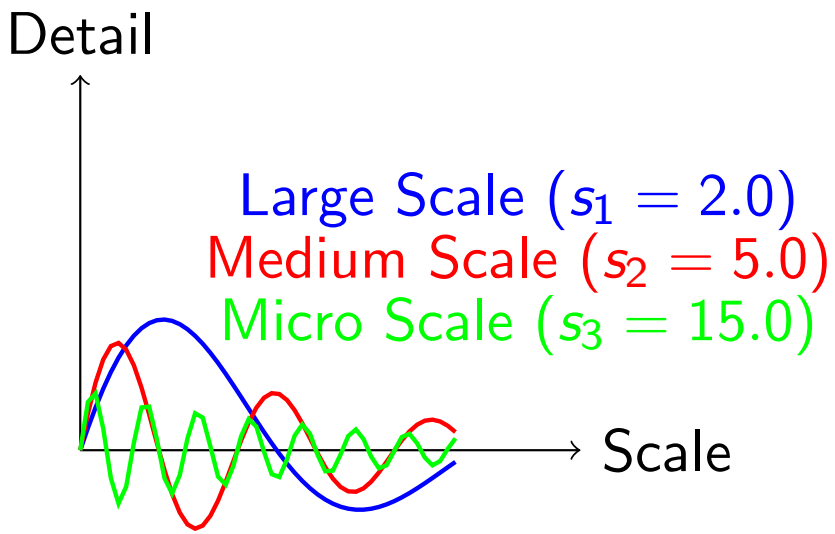
where h_{ijk} are hash values and f_{xyz} are fractional coordinates

- Multi-octave Composition:

$$R(p) = \sum_{i=1}^{3} w_i \cdot noise(p \cdot s_i)$$

with weights: $w_1 = 0.5$, $w_2 = 0.3$, $w_3 = 0.15$

Rust Formation - Multi-scale Detail



- Each scale contributes unique details:
  - Base pattern: $noise(p \cdot 2.0) \cdot 0.5$
  - Medium detail: $noise(p \cdot 5.0 + base) \cdot 0.3$
  - Fine detail: $noise(p \cdot 15.0 + medium) \cdot 0.15$

Advanced Rust Formation Features

- Edge Detection and Weathering:

$$E(p) = (1 - |\hat{p} \cdot \hat{n}|)^3$$

- Moisture Influence:

$$M(p) = noise(p \cdot 25.0) \cdot moisture$$

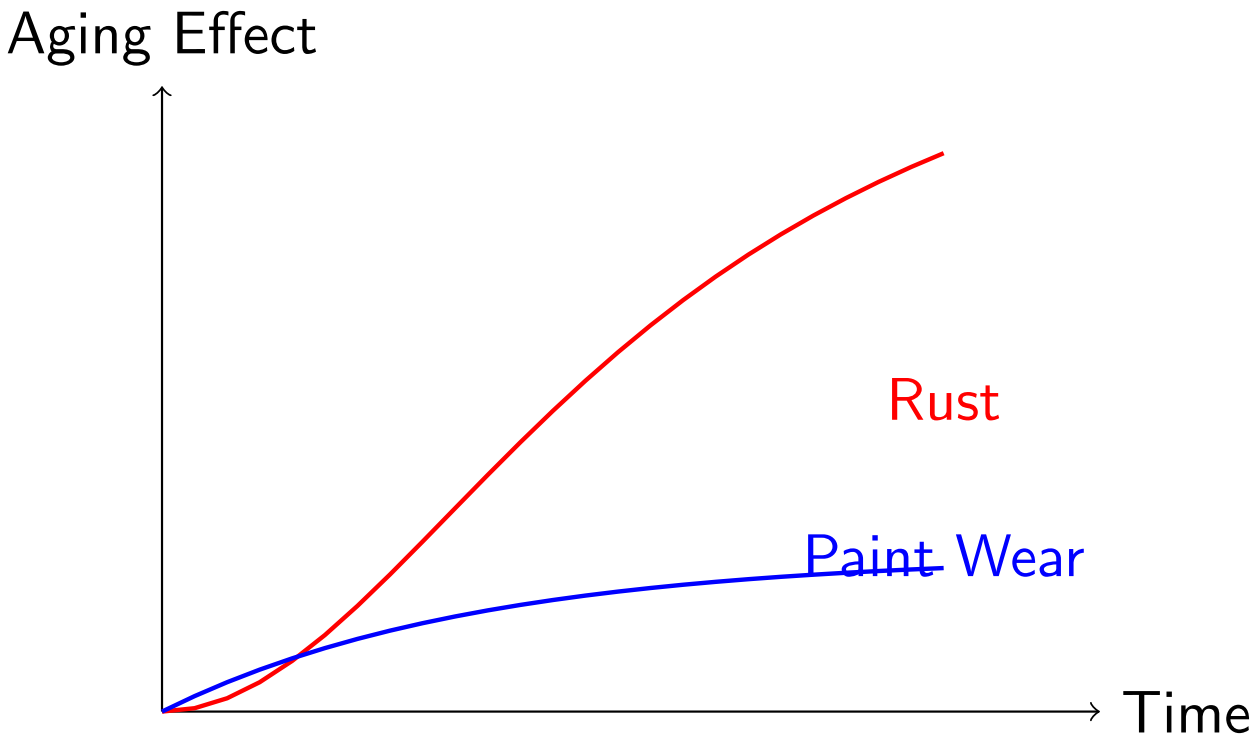
- Combined Rust Pattern:

$$rust = R(p) \cdot E(p) \cdot (1 + M(p))$$

- Material Layer System:
  - Deep rust: $smoothstep(0.3, 0.7, rust)$
  - Surface rust: $smoothstep(0.1, 0.4, rust)$
  - Light rust: $smoothstep(0.0, 0.3, rust)$
- Normal Map Generation:

$$N(p) = normalize(n + \nabla rust \cdot depth)$$

Material Evolution System



- Different materials age at varying rates
- Rust: Accelerating growth with saturation
- Paint: Exponential decay pattern
- Interactive control of aging progression

Painting Aging System - Mathematical Model

- Base Texture Sampling:

$$T(uv) = texture(paintingTexture, uv)$$

- Aging Effects:
  - Cracking Pattern:

$$C(p) = noise(p \cdot (20.0 + age \cdot 30.0))$$

$$crack = smoothstep(0.6, 0.7, C(p)) \cdot age$$

  - Paint Peeling:

$$P(p) = noise(p \cdot (5.0 + age \cdot 10.0) + age \cdot 2.0)$$

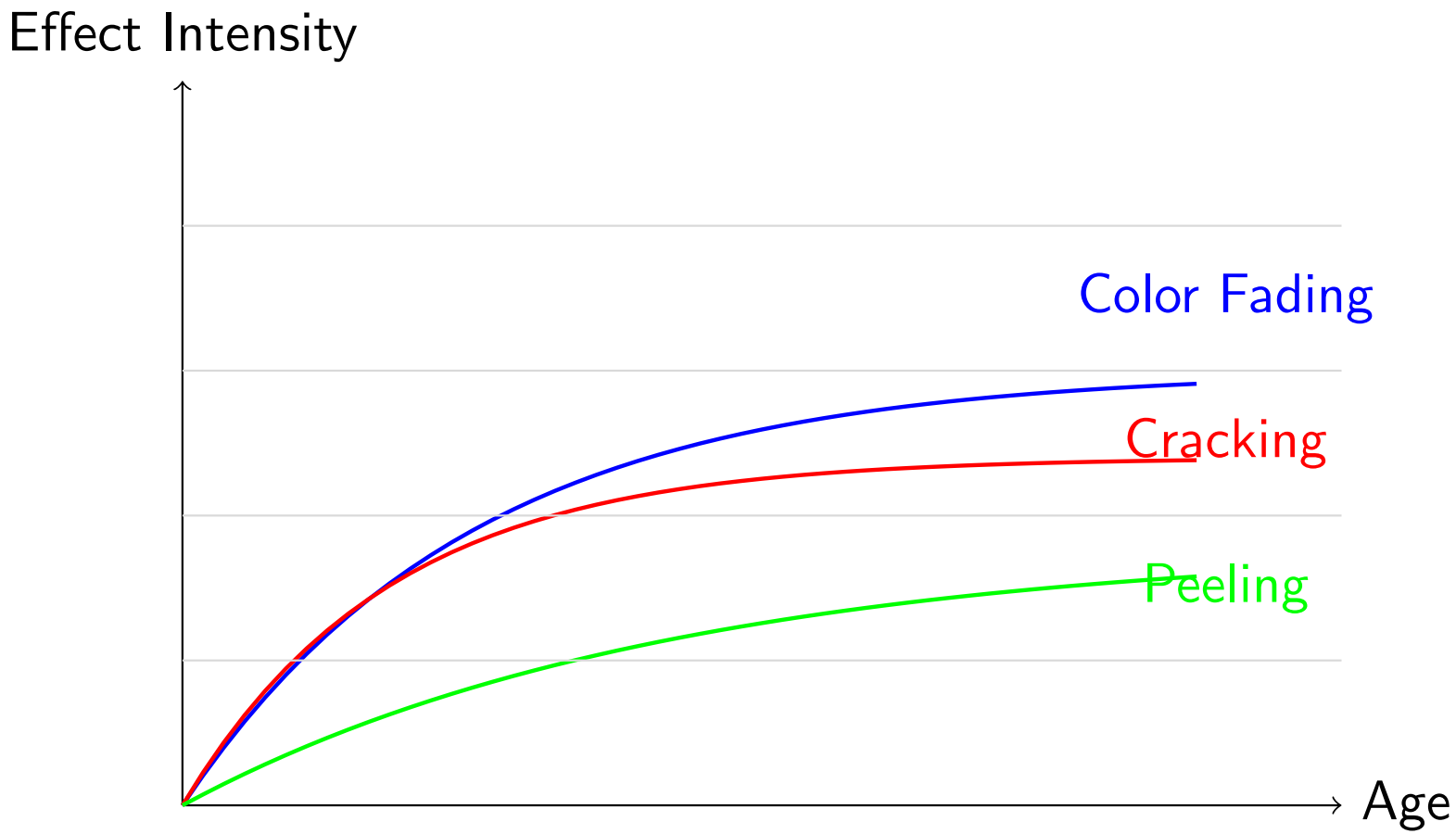
$$peel = smoothstep(0.7, 0.8, P(p)) \cdot age$$

  - Color Aging:

$$color_{aged} = mix(T(uv), T(uv) \cdot 0.7, age \cdot 0.5)$$

$$color_{yellow} = mix(color_{aged}, color_{aged} \cdot yellowTint, age \cdot 0.3)$$

Painting Aging System - Effect Progression



- Color fading occurs most rapidly in early stages
- Cracking develops at moderate rate
- Peeling is a slower, progressive effect

Painting Material Properties

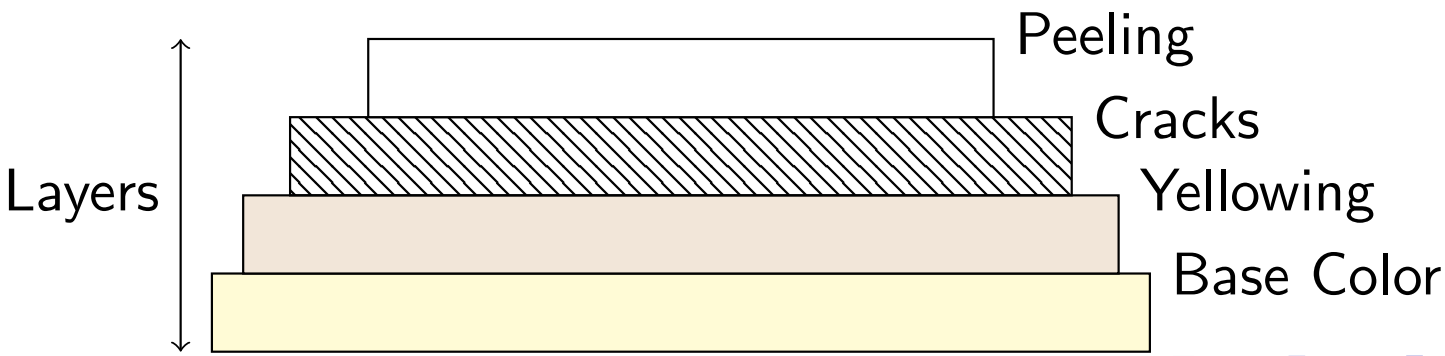
- Normal Map Generation:

$$N(p) = normalize(n + vec3(crack, crack, 0.0) \cdot age \cdot 0.2 + \text{vec3(peel, peel, 0.0)} \cdot age \cdot 0.3)$$

- Material Properties Evolution:
  - Roughness: $mix(0.2, 0.8, age)$
  - Metallic: Always 0.0 (non-metallic)
  - IOR: 1.5 (constant)
- Final Color Composition:

$$color_{final} = mix(color_{yellow}, vec3(0.2), crack \cdot 0.5)$$

$$color_{final} = mix(color_{final}, vec3(0.1), peel)$$



Ground Procedural Generation

- Height Field Generation:

$$H(x,z) = -\sum_{i=1}^{3} C_i(x,z) + N(x,z)$$

where:

- $C_i(x,z)$ are crater functions:

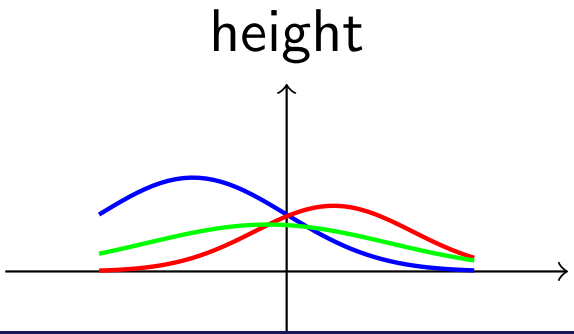
$$C_1(x,z) = e^{-||(x,z)+(1.0,0.5)||\cdot 1.5}$$

$$C_2(x,z) = 0.7 \cdot e^{-||(x,z)-(2.0,-1.0)||\cdot 2.0}$$

$$C_3(x,z) = 0.5 \cdot e^{-||(x,z)-(-1.5,1.0)||\cdot 1.0}$$

- $N(x,z)$ is terrain roughness:

$$N(x,z) = 0.2 \cdot noise(p \cdot 2.0)$$



Combined Crater Profile

Dynamic Ground Features

- Normal Calculation for Lighting:

$$\vec{N}(x,z) = normalize \begin{pmatrix} \frac{\partial H}{\partial x} \\ 1.0 \\ \frac{\partial H}{\partial z} \end{pmatrix}$$

- Dynamic Puddle System:

$$P(x,z,t) = depth(x,z) \cdot moisture \cdot (1 + R(x,z,t))$$

where:

  - $depth(x,z) = smoothstep(0.1, 0.3, -H(x,z))$
  - $R(x,z,t)$ is the ripple function:

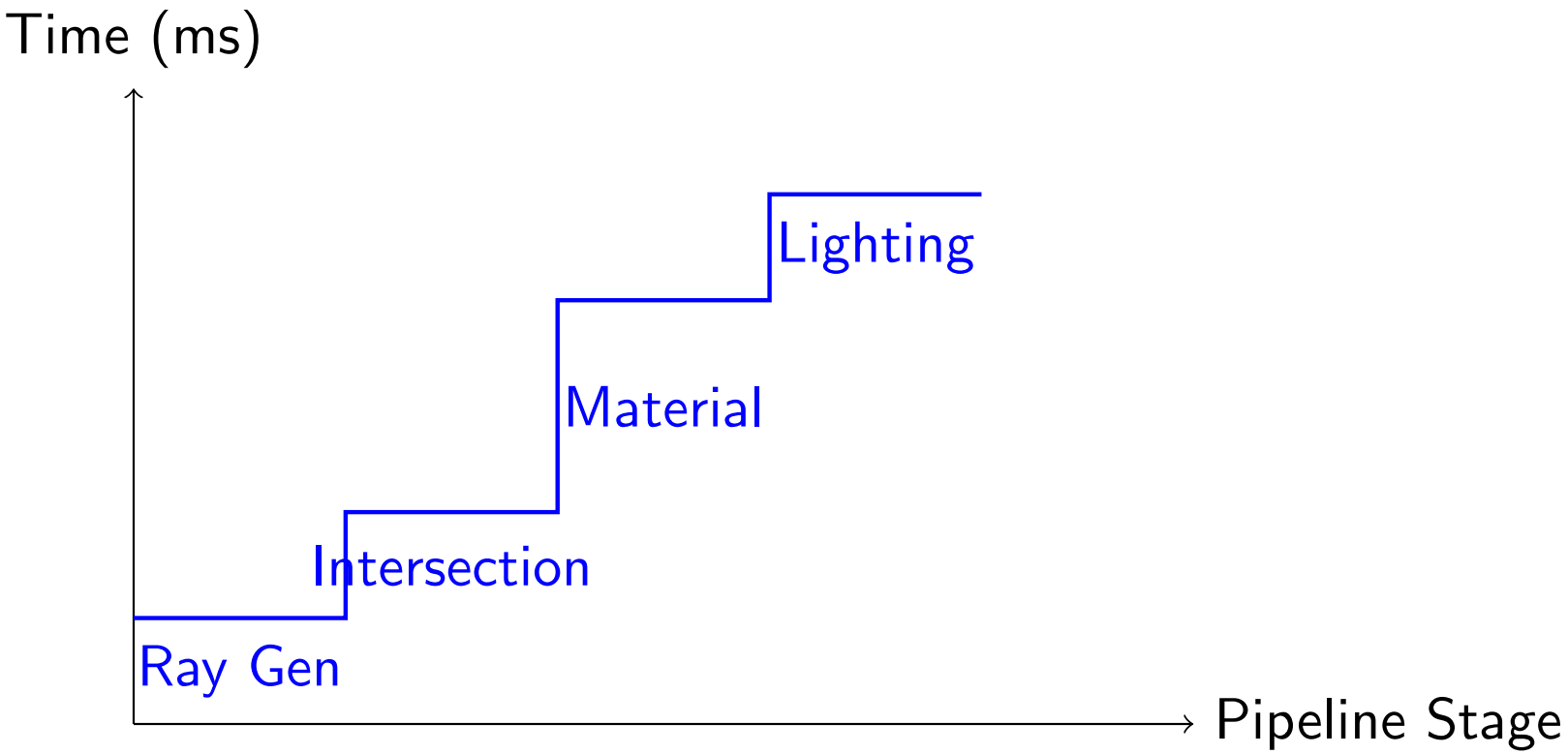
$$R(x,z,t) = \sum_{i=1}^{5} w_i \cdot sin(d_i \cdot 8.0 - t \cdot 3.0) \cdot e^{-d_i \cdot 1.5}$$

  - d_i is distance to ripple center i

- Material Blending:

$$M(p) = lerp(base, puddle, P(x,z,t))$$

Ray Tracing Pipeline



- Efficient compute shader-based ray tracing
- Custom material system for aging effects
- Real-time performance with complex materials
- Dynamic lighting adaptation to weather

Physics Integration

- Verlet Integration for Object Motion:

$$x(t + \Delta t) = 2x(t) - x(t - \Delta t) + a(t)\Delta t^2$$

- Collision Response:

$$v' = v - (1 + e)(v \cdot n)n$$

where e is restitution coefficient

- Object-Environment Interaction:

$$F_{total} = F_{gravity} + F_{collision} + F_{weather}$$

Weather System

- Dynamic Weather Components:
  - Cloud Coverage:

$$C(t) = noise(p \cdot 5.0 + t \cdot 0.1) \cdot moisture$$

  - Light Intensity:

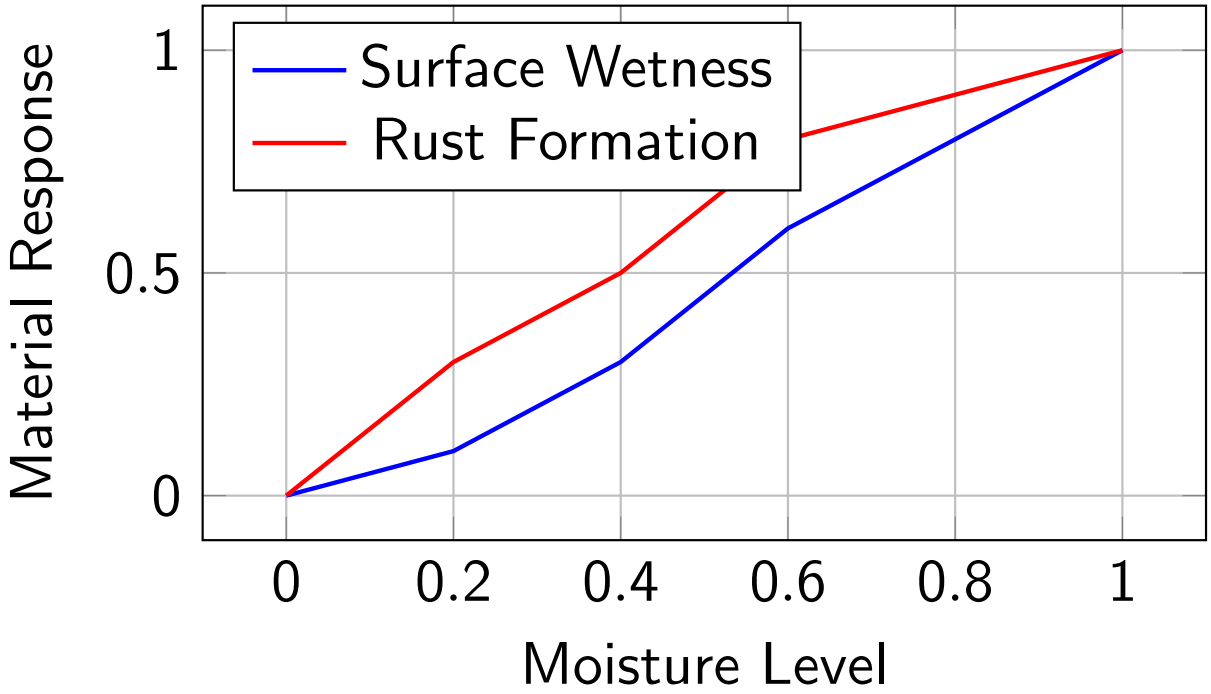
$$I(t) = I_{base} \cdot (1.0 - moisture \cdot 0.7)$$

  - Puddle Formation:

$$P(x, z) = smoothstep(0.1, 0.3, D(x, z)) \cdot moisture$$

- Surface Effects:
  - Dynamic puddle ripples
  - Wet surface reflections
  - Cloud shadows
- Environmental Response:
  - Material wetness adaptation
  - Light scattering through clouds
  - Surface moisture accumulation

Weather Impact on Materials



- Weather-Material Interaction:
  - Enhanced rust formation in wet conditions
  - Dynamic surface reflectivity
  - Moisture-dependent aging rates
- Visual Effects:
  - Real-time puddle formation
  - Dynamic cloud shadows
  - Wet surface sheen

Interactive Features

- Real-time Parameter Control:
  - Adjust moisture levels (M/N keys)
  - Control aging rate (R/F keys)
  - Dynamic weather conditions
- Physical Interactions:
  - Object collisions affect wear patterns
  - Surface impacts create aging hotspots
  - Environment-aware material aging
- Visual Feedback:
  - Immediate material response to changes
  - Progressive aging visualization
  - Weather effect demonstration

Demo Time!

Controls:

WASD	Movement
Mouse	Look around
Space	Jump
R/F	Adjust rust/age
M/N	Control moisture

Watch for:

- Dynamic rust formation
- Paint deterioration