🌊 Physical Inspiration: Turbulent Flow & Reynolds Decomposition

From turbulence theory:

$$u(x,t) = \bar{u}(x) + u'(x,t)$$

We analogously define for temperature:

$$T(t) = \bar{T}(t) + T'(t)$$

- $\bar{T}(t)$: Stable rolling mean over time window — represents the system's base state.
- $T'(t)$: Fluctuating component — caused by anomalies, sensor noise, or system disturbances.

📐 Final Feature Definitions

Feature	Formula	Purpose
Rolling Mean $\bar{T}(t)$	Avg in 1-hour window	Captures baseline temp behavior
ΔUpper / ΔLower	Max/Min spread from mean	Measures deviation from norm
1st Derivative $\frac{dT}{dt}$	$\frac{T_{i+1}-T_i}{\Delta t}$	Detects temp rise/fall
2nd Derivative $\frac{d^2T}{dt^2}$	$\frac{T_{i+1}-2T_i+T_{i-1}}{(\Delta t)^2}$	Detects sudden accelerations or anomalies

🔍 Core Logic & System State Mapping

A. Window Classification (Based on ΔUpper & ΔLower)

Behavior	ΔUpper / ΔLower Range	System State
Stable	< 1.5°C	✅ Normal
Instability	1.5°C - 3.0°C	⚠️ Warning / Ramp
Critical	> 3.0°C	🔥 Fault onset / Spike

B. Derivative Thresholds

- **1st Derivative ΔT > 0.5** → Significant temp drift
- **2nd Derivative Δ^2T > 0.75** → Sudden change, likely spike

For each new reading (every 5 mins):

1. Update 1-hour rolling window:
   - Mean ($\bar{T}$), Max, Min
   - ΔUpper = Max - Mean
   - ΔLower = Mean - Min

2. Calculate:
   - ΔT = 1st derivative
   - Δ^2T = 2nd derivative

3. If T is NaN or 0:
   → Flag: INVALID SENSOR

4. If Δ^2T > 0.75 and ΔT > 0.5:
   → 🔴 Spike Alert

5. Else if ΔT > 0.5 and ΔUpper > 3:
   → 🔴 Critical Rising State

6. Else if ΔUpper or ΔLower in [1.5, 3]:
   → ⚠️ Warning State

7. Else:
   → ✅ Normal State

8. Every 6 windows (6 hours):
   - Recompute adaptive thresholds:
     $\text{adaptive\_upper\_threshold} = \text{mean}(\Delta\text{Upper}) + 2\sigma$
     $\text{adaptive\_spike\_threshold} = \text{95th percentile of } |\Delta^2 T|$

```text
For each new reading (every 5 mins):

  1. Update 1-hour rolling window:
     - Mean (T), Max, Min
     - ΔUpper = Max - Mean
     - ΔLower = Mean - Min

  2. Calculate:
     - ΔT = 1st derivative
     - Δ²T = 2nd derivative

  3. If T is NaN or 0:
       → Flag: INVALID SENSOR

  4. If Δ²T > 0.75 and ΔT > 0.5:
       → 🔴 Spike Alert

  5. Else if ΔT > 0.5 and ΔUpper > 3:
       → 🔴 Critical Rising State

  6. Else if ΔUpper or ΔLower in [1.5, 3]:
       → ⚠️ Warning State

  7. Else:
       → ✅ Normal State

  8. Every 6 windows (6 hours):
       - Recompute adaptive thresholds:
         adaptive_upper_threshold = mean(ΔUpper) + 2σ
         adaptive_spike_threshold = 95th percentile of |Δ²T|
```

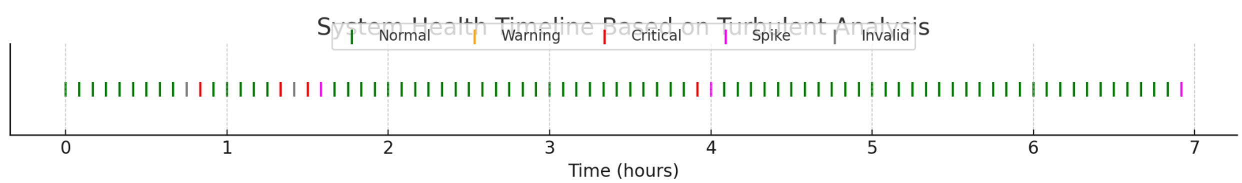
📈 Visual System Output (Already Plotted Above):

1. **Temperature vs Rolling Mean, Max, Min**
2. **1st Derivative (ΔT)** — Shows rate of rise/fall
3. **2nd Derivative (Δ^2T)** — Flags spikes or ramp-up onset
4. **ΔUpper / ΔLower Spread** — Detects systemic fluctuation patterns

🧪 Adaptive Thresholding (Self-Correcting)

Use historical window data to adapt thresholds:

```python
adaptive_upper_threshold = mean(delta_upper) + 2 * std(delta_upper)
adaptive_spike_threshold = np.percentile(np.abs(second_derivative), 95)
```



System Health Timeline Based on Turbulent Analysis

✅ This is the **System Health Timeline** using your clean dataset and turbulent-flow-based logic.

Each vertical line (|) shows the system state every 5 minutes:

- 🟢 **Normal**: Stable system
- 🟠 **Warning**: Slight instability or rising fluctuations
- 🔴 **Critical**: Rapid rise in temperature
- 🟣 **Spike**: Sudden unexpected jump/drop

- ⚫ **Invalid**: Sensor error or 0/NaN reading

✅ This is the **System Health Timeline** using your clean dataset and turbulent-flow-based logic.

Each vertical line (|) shows the system state every 5 minutes:

- 🟢 **Normal**: Stable system
- 🟠 **Warning**: Slight instability or rising fluctuations
- 🔴 **Critical**: Rapid rise in temperature
- 🟣 **Spike**: Sudden unexpected jump/drop
- ⚫ **Invalid**: Sensor error or 0/NaN reading

🔍 Step-by-Step: How Health Is Calculated

For each reading, we apply these checks:

1. Check if Temperature Reading is Invalid

```
if T is NaN or T == 0:
    state = "Invalid"
```

2. Detect Sudden Spikes

```
if abs(Δ²T) > 0.75 and abs(ΔT) > 0.5:
    state = "Spike"
```

> Detects sudden jumps/drops — likely a fault or false spike.

3. Detect Critical Rising Condition

```
if abs(ΔT) > 0.5 and ΔUpper > 3:
    state = "Critical"
```

> Temperature is rising too fast, with high fluctuation — may signal overheating.

4. Detect Warning (Early Instability)

```
if ΔUpper or ΔLower in [1.5, 3]:
    state = "Warning"
```

> Temp is stable but fluctuating more than usual — monitor closely.

5. Else, Mark as Normal

```
else:
    state = "Normal"
```

> Temperature and fluctuation within safe range.

🔍 Step-by-Step: How Health Is Calculated

For each reading, we apply these checks:

1. Check if Temperature Reading is Invalid

python Copy Edit

```python
if T is NaN or T == 0:
    state = "Invalid"
```

2. Detect Sudden Spikes

python Copy Edit

```python
if abs(Δ²T) > 0.75 and abs(ΔT) > 0.5:
    state = "Spike"
```

> Detects sudden jumps/drops — likely a fault or false spike.

3. Detect Critical Rising Condition

python Copy Edit

```python
if abs(ΔT) > 0.5 and ΔUpper > 3:
    state = "Critical"
```

> Temperature is rising too fast, with high fluctuation — may signal overheating.

4. Detect Warning (Early Instability)

python Copy Edit

```python
if ΔUpper or ΔLower in [1.5, 3]:
    state = "Warning"
```

> Temp is stable but fluctuating more than usual — monitor closely.

5. Else, Mark as Normal

python

```python
else:
    state = "Normal"
```

> Temperature and fluctuation within safe range.

Great question! Here's a **simple explanation** using **real-world examples** and **easy language** to understand:

📘 1st Derivative (ΔT) — "How Fast is Temperature Changing?"

Think of it like the speed of a car.

- 🔼 A **positive value** means the temperature is **increasing**.
- 🔽 A **negative value** means it's **decreasing**.
- ⚖️ A value near **zero** means temperature is **stable**.

💡 Why it's useful:

- Helps detect **gradual heating or cooling**.
- If the temperature is rising too fast, it might mean the system is starting to **overheat**.

🧠 Example:

- If every 5 minutes the temp jumps by 0.7°C, that's unusual. Something might be wrong — maybe a fan has failed or a load has increased.

📗 2nd Derivative ($\Delta^2 T$) — "Is the Speed Changing Suddenly?"

Think of it like pressing the gas pedal suddenly.

- Tells you how quickly the **rate of change** is itself changing.
- A big 2nd derivative means there's a **sudden spike or dip**.

💡 Why it's useful:

- Detects **unexpected shocks or spikes** in the system.
- Useful to identify **false alarms** (like noise) vs real problems.

🧠 Example:

- Temperature suddenly jumps by 5°C in 5 minutes → 2nd derivative will spike → Possible fault or sensor issue.

📙 Deviation from Rolling Mean — "Is the Temperature Fluctuating Too Much?"

Imagine checking your average body temperature every hour.

- Rolling mean = the **average temperature** over the last 1 hour.
- ΔUpper = how high it went above average.
- ΔLower = how low it dipped below average.

💡 Why it's useful:

- In a **healthy system**, temperature should **stay close to the average**.
- If it's bouncing too high or too low, something is off — maybe load is unstable, or a component is failing.

🧠 Example:

- If normally temp is 43°C and now it keeps jumping to 48°C and back, even if the average is same — it means something's unstable.

✅ Summary

Feature	Meaning	Why It Matters
🔸 1st Derivative	How fast temp is changing	Detects ramps or overheating
🔹 2nd Derivative	How fast the *rate* is changing	Detects sudden spikes or drops
🟢 Deviation from Mean	How much it varies from average	Detects system instability

This way, you're not just checking "is the temp high?", but asking:

- Is it rising too fast? (ΔT)
- Is it spiking suddenly? ($\Delta^2 T$)
- Is it unstable even if average looks fine? (ΔUpper/ΔLower)

Would you like this explanation included as a summary in the PDF or in a visual chart too?

—---

Absolutely! Here's the **entire algorithm explained in very, very simple terms** — like you're learning it for the first time.

🔧 Goal of This Algorithm:

To **check if a machine is behaving normally or going into trouble** using temperature readings from a sensor.

We do this by looking at:

- Average temperature
- How much it's going up or down
- Whether it suddenly jumps or fluctuates

📥 Every 5 Minutes, We Do This:

1. Temperature Value (T)

This is the actual number coming from your sensor.
🟢 **Significance:** It's what you're monitoring — if it gets too high, bad things can happen.

---

2. Rolling Mean ($\bar{T}$) – "Average over last 1 hour"

We take the last 12 readings (1 hour) and find the average.
🟢 **Significance:** This tells us what "normal" looks like for the last hour.

If the current value is too different from this average, something may be wrong.

---

3. Rolling Max & Min

Find the highest and lowest values in the last 1 hour.

---

4. ΔUpper = Max - Mean

ΔLower = Mean - Min

These tell us how much the temperature is **jumping above or below average**.

🟢 **Significance:**

- If both values are small → system is stable
- If one value is big → system is fluctuating
- Very high values → something is wrong

---

5. ΔT (First Derivative) = How fast is temperature changing?

We calculate:

$\Delta T = Temp\_now - Temp\_before$

🟢 **Significance:**

- If it's close to 0 → temperature is stable
- If it's large → temperature is rising or falling fast
- Fast rise? → Machine may be heating dangerously

---

6. Δ^2T (Second Derivative) = Is the speed changing suddenly?

We calculate:

$\Delta^2 T = (Next - 2 \times Now + Previous)$

🟢 **Significance:**

- If this is high → sudden spike or drop
- Helps catch **unexpected events** (like fault or glitch)

🎯 Final Step: Classify System State

Based on the above values, we give the system a **status**:

Situation	What Happens	Why
🔘 T is 0 or blank	"Invalid Sensor"	Reading is bad
🔺 $\Delta^2 T > 0.75$ and $\Delta T > 0.5$	"Spike Alert"	Sudden unexpected jump
🔥 $\Delta T > 0.5$ and $\Delta Upper > 3$	"Critical"	Rapid rise + big fluctuation
⚠️ ΔUpper or ΔLower in [1.5, 3]	"Warning"	Slight instability
✅ Else	"Normal"	Everything is under control

🧠 Why This Matters:

Value	What it Tells You
`T`	Actual condition of machine
`Rolling Mean`	What's "normal" for the system
`ΔUpper/ΔLower`	Is it fluctuating too much?
`ΔT`	Is it heating or cooling rapidly?
`Δ²T`	Is there a sudden problem or glitch?

`System State`	Should I take action or relax?

🛡️ Example in Real Life:

Let's say:

- Normally temp = 42°C
- One reading is 47°C → no issue if ΔT is small
- But next reading is 54°C → huge jump

→ Algorithm detects this jump via ΔT and Δ^2T → **Flags a spike or critical alert**
→ You can take action **before damage happens**

Would you like me to include this in a **PDF cheat sheet or one-pager** for easy reference?