**Abstract**

Wireless Sensor Networks (WSNs) are constrained by limited energy resources, making network lifetime a critical performance metric. Traditional duty cycling approaches rely on fixed schedules or residual energy alone, which often leads to premature node depletion and reduced stability. This paper proposes a **context-aware duty cycling mechanism** that dynamically adjusts node sleep–wake schedules using three lightweight parameters: residual energy, data entropy, and local traffic load. The entropy measure captures the variability of sensed data, ensuring nodes remain active when information is more valuable, while traffic awareness reduces congestion and balances relay burdens. The duty cycle controller combines these parameters into a simple adaptive rule without relying on machine learning or computationally heavy optimization techniques. Simulation results on a synthetic sensing model demonstrate significant improvements in First Node Death (FND), Half Node Death (HND), and Last Node Death (LND) metrics compared to baseline fixed-duty approaches. The proposed method achieves enhanced stability and extended lifetime with minimal overhead, making it suitable for resource-constrained WSN deployments.

**Pseudocode**

Algorithm: Context-Aware Duty Cycling (CA-DC)

Input:

N ← number of nodes

E0 ← initial energy per node

Dmin, Dmax← minimum and maximum duty cycle bounds

W ← entropy window size

B ← number of bins for entropy

Tmax ← normalization factor for traffic load

βE, βH, βT← weighting coefficients

η, δ ← smoothing and hysteresis parameters

rounds ← maximum number of simulation rounds

Output:

FND ← round when first node dies

HND ← round when half nodes die

LND ← round when last node dies

------------------------------------------------------------

Initialization:

For each node i in {1,…,N} do

Place node i randomly in the sensing field

Set energy E[i] = E0

Set duty cycle D[i] = 0.1

Initialize entropy window X[i] = empty

Set generated packets G[i] = 0

Set forwarded packets F[i] = 0

End For

------------------------------------------------------------

For round r = 1 to rounds do

alive\_nodes = count(E[i] > 0)

Update FND, HND, LND markers if thresholds are reached

For each node i with E[i] > 0 do

// --- Context Computation ---

1. Residual Energy:

En = E[i] / E0

2. Data Entropy:

Sense new value x

Update window X[i] with x (keep last W values)

Compute probability distribution p over B bins

H = -Σ p \* log(p)

Normalize: Hn = H / log(B)

3. Traffic Load:

G[i] = 1 (self-generated packet)

F[i] = number of forwarded packets in last round

Tl = F[i] + α \* G[i]

Tn = min(1, Tl / Tmax)

// --- Duty Cycle Controller ---

S = βE \* (1 - En) + βH \* Hn + βT \* Tn

D\* = clamp(Dmin + γ \* S, Dmin, Dmax)

If |D\* - D[i]| > δ then

D\_next[i] = (1 - η) \* D[i] + η \* D\*

Else

D\_next[i] = D[i]

End If

// --- Energy Consumption ---

Compute distance d to Base Station

If d < d0 then

Etx = E\_elec \* k + E\_fs \* k \* d^2

Else

Etx = E\_elec \* k + E\_mp \* k \* d^4

End If

Erx = E\_elec \* k \* F[i]

E[i] = E[i] - (Etx + Erx) \* D[i]

If E[i] < 0 then E[i] = 0

End For

// Update duty cycles

For each node i

D[i] = D\_next[i]

End For

End For

------------------------------------------------------------

Return: FND, HND, LND

* Nodes adjust how long they stay awake (duty cycle) depending on:  
   **How much energy is left**  
   **How important the sensed data is (entropy)**  
   **How much traffic they handle**
* This avoids wasting energy on nodes with little to contribute.
* Result → **longer lifetime, delayed first node death, balanced energy use**.



