International Journal of Research and Technology (IJRT)

International Open-Access, Peer-Reviewed, Refereed, Online Journal

ISSN (Print): 2321-7510 | ISSN (Online): 2321-7529

| An ISO 9001:2015 Certified Journal |

Snake Optimization Algorithm (SOA): A Novel Approach for Blockchain Scalability and Carbon Footprint Minimization

[1]Harshit Rai Verma, [2]Ananya Gulati, [3]Aakash Jain

[1,2,3]Research Scholar, Netaji Subhas University of Technology, New Delhi, India

[4]Dr. Ankush Jain & [5]Dr. V.S.K.V Harish

[4,5]Supervision, Netaji Subhas University of Technology, New Delhi, India

Correspondence: harshit.verma.ug22@nsut.ac.in

ABSTRACT

While blockchain technology holds promise for decentralized applications, the Proof-of-Work (PoW) consensus protocol consumes significant amounts of energy and thus has a substantial carbon footprint. This paper presents and tests the use of the Snake Optimization Algorithm (SOA), a new swarm-intelligence metaheuristic algorithm, modeled on the behavior of snakes during their mating season, to optimize three key parameters influencing the operation of blockchains: the Tolerance Factor for additional PoW latency (TFT), the Green Node Preference Factor (GPF), and the Workload Throttling Factor (WTF), with the aim of minimizing the carbon footprint. A custom multi-objective fitness function is developed, which minimizes energy consumed per transaction, end-to-end latency, transaction fees, and reliability penalties with an emphasis on sustainability by assigning the most weight to minimize energy consumption (0.45). The results of the simulation under four different renewable-energy penetration scenarios (f_green = 0%, 20%, 50%, 100%) do show that SOA is able to converge within 500 iterations, and that power-per-transaction can be reduced by up to 90% (from 50 W to 5 W) as the availability of green nodes increases. With 50% green penetration, the effective green transaction fraction increases from 0.50 to 0.62 with SOA optimized scheduling. The statistical analysis show that the fitness landscape is always smooth and unimodal when renewable penetration is above 20%, leading to rapid convergence. The proposed framework is the first one that connects metaheuristic optimization to blockchain carbon-footprint minimization, which can be extended to Proof-of-Stake architectures, real-time carbon-intensity-aware scheduling, and multi-objective consensus parameter tuning.

Keywords: Blockchain, Carbon Footprint, Snake Optimization Algorithm, Proof-of-Work, Green Energy, Metaheuristic, Energy Efficiency, Sustainable Computing

1. INTRODUCTION

The revolutionary blockchain technology has been a game-changer in the 21st century, powering decentralized and tamper-proof record-keeping systems in various sectors such as cryptocurrency, supply-chain management, healthcare data integrity, decentralized finance (DeFi), and smart contracts. The fundamental security promise is based on consensus mechanisms, which are distributed protocols whereby a network of potentially untrusted nodes can reach a consensus on a canonical transaction history, with no central node. The Byzantine

fault tolerance mechanism most widely used in most major public blockchains (such as Bitcoin (BTC), Bitcoin Cash (BCH), Litecoin (LTC), etc.) is called Proof of Work (PoW), which makes all nodes (miners) performing computational power to solve a cryptographic puzzle. The winner of the puzzle is entitled to add the next block to the chain, and receives a block reward. PoW is beautiful in terms of its security attributes but it has a strong negative environmental externality because of competitive, arms-race character of puzzle-solving which encourages miners to use even more powerful and electricity-consuming hardware (CPUs, GPUs, ASICs) leading to the staggering level of energy consumption. At present (2024), it is estimated that the Bitcoin network consumes 120-150 TWh of electricity on an annual basis, which is the same amount of electricity used by the countries of Argentina or Norway. A large proportion of this is generated from fossil fuels, that results in considerable carbon dioxide equivalent (CO_2e) emissions. The total carbon footprint of the global blockchain industry represents a significant and increasingly unaddressed issue for international climate pledges, not least the Paris Agreement's goal of becoming "net zero" by 2050. A critical research question, which is answered in this paper, is whether the operational parameters of a PoW blockchain network can be tuned intelligently (without changing the consensus mechanism) to favor the routing of transaction validation work to renewable energy-powered miners, thereby lowering the effective carbon intensity of each transaction. We answer this positively by adapting the Snake Optimization Algorithm (SOA) which is recently proposed swarm-intelligence metaheuristic, for optimizing three important scheduling parameters namely green-node preference, latency tolerance and workload throttling.

1.1 Motivation

Academic and policy attention has been drawn to the environmental impact of blockchain operations. Proving that consumption of energy is unnecessary, other mechanisms like Proof of Stake (PoS), provide dramatic reductions, but the transition of existing PoW networks brings with it enormous technical, economic, and governance challenges – the Ethereum ‘Merge’ for instance took years to develop and engage with the community for consensus. An optimization-layer solution that optimizes the carbon profile without protocol-level changes is a pragmatic and immediately deployable pathway towards improving the carbon profile for networks that remain on PoW. Moreover, although the fields of metaheuristic optimization are extensive, especially in engineering design, power systems and scheduling problems, they have not yet addressed the optimization problems related to blockchain in a serious way. The Snake Optimizer (SO) introduced in Knowledge-Based Systems in 2022 by Hashim and Hussien has been tested on the CEC 2017 and CEC 2020 benchmark function suites, as well as on four constrained engineering problems, showing the superiority of its performance. However, it has not been used for blockchain scheduling or sustainability optimization. This is a new and current research gap.

1.2 Contributions

The specific contributions of this paper are:

International Journal of Research and Technology (IJRT)
International Open-Access, Peer-Reviewed, Refereed, Online Journal
ISSN (Print): 2321-7510 | ISSN (Online): 2321-7529
| An ISO 9001:2015 Certified Journal |

- Development of a multi-objective blockchain fitness function to capture the power consumption per transaction, latency, transaction fees, and uptime reliability in one normalized optimization objective, weighted for sustainability.
- Optimization of three blockchain scheduling parameters, namely Tolerance Factor for Latency (TFT), Green Node Preference Factor (GPF), and Workload Throttling Factor (WTF), for various penetration scenarios of renewables.
- Analytical derivation of a new model of the probability that a renewable-energy miner will be selected to solve the PoW puzzle in a window extended with TFT, giving a principled foundation for the calculation of the green-fraction of the fitness function.
- Simulation results for the convergence behavior and optimum parameter values of SOA and the resulting carbon-footprints reductions for f_green ∈ {0%, 20%, 50%, 80%, 100%}.
- Design of a novel sustainable framework for blockchain that serves as the basis for subsequent phases of carbon-intensity-aware scheduling, variants of PoS, and multi-objective Pareto-front optimization.

1.3 Paper Organization

The problem statement and network model is presented in Section 2. Research objectives are presented in section 3. Section 4 reviews the related work. The SOA algorithm is described in Section 5. The proposed methodology along with fitness function derivation is presented in section 6. The results of the simulations and analysis are presented in Section 7. Novelty, limitations, and future directions are discussed in Section 8. The paper is rounded off by section 9.

2. PROBLEM STATEMENT

Let N be a PoW blockchain network with n_total nodes such that f_green of them utilize renewable energy sources (wind/solar/hydroelectric etc.) and (1 - f_green) are using conventional fossil fuel energy from the grid. Each node i has its own PoW solving time t_i (a random variable that depends on the node's hash rate), power consumption P_i (P_green for renewable nodes, P_regular for conventional nodes) and uptime u_i ∈ [0, 1]. As soon as a new transaction comes in all nodes start solving the PoW puzzle at the same time. In the normal PoW, the transaction is assigned to the first node that provides a correct solution, such as node A, that can solve the transaction in t_A. The carbon footprint of the resulting transaction validation depends on which node wins: a green node winner produces emissions close to zero, and a conventional node winner produces emissions that are proportional to P_regular. The central optimization problem is: find a scheduling policy, with parameters (TFT, GPF, WTF), that minimizes the expected carbon footprint per transaction, while maintaining constraints on maximum latency, minimum miner incentive fairness, and network uptime. Formally:

Minimize **f(x) = w_lat × lat_norm + w_fee × fee_norm + w_power × power_norm + w_uptime × uptime_penalty**

Subject to: latency ≤ lat_max, uptime ≥ uptime_min, TFT ∈ [0, TFT_max], GPF ∈ [0, 1], WTF ∈ [0, 1]

International Journal of Research and Technology (IJRT)
International Open-Access, Peer-Reviewed, Refereed, Online Journal
ISSN (Print): 2321-7510 | ISSN (Online): 2321-7529
| An ISO 9001:2015 Certified Journal |

where x = [N_total, t_pow_ms, TFT, P_green, P_regular, uptime] is the decision vector, and w_lat = 0.30, w_fee = 0.25, w_power = 0.45, w_uptime = 1.00 are the objective weights prioritizing sustainability (power) and reliability (uptime).

3. RESEARCH OBJECTIVES

The following measurable objectives guide the design and evaluation of the proposed SOA-based framework:

1. Create a probabilistic model to the effective green transaction fraction F_green that is achievable by tuning TFT and GPF, based on the Poisson distribution, and obtain the closed form expression of power-per-transaction as a function of f_green.
2. Design an appropriate multi-objective fitness function normalized so that it can be fairly compared between different network configurations and scaling scenarios.
3. Optimize (TFT, GPF, WTF) for five representative renewable energy penetration levels (f_green = 0%, 20%, 50%, 80% and 100%) using SOA and characterize the optimal parameter configurations.
4. Achieve at least 50% reduction in power-per-transaction for f_green = 50% without using any other optimization techniques except intelligent scheduling.
5. Converge to SOA within 1,000 iterations for all the scenarios tested, thereby validating the algorithmic efficiency.
6. Show the extensibility of the framework by showing that the fitness function and the space of values of the SOA parameters generalizes for different network sizes (N_total ranging from 100 to 10000 nodes).

4. LITERATURE REVIEW

4.1 Blockchain Energy Consumption and Carbon Footprint

There is a lot of documentation on the environmental footprint of PoW blockchains. Stoll et al. (2019) calculated the annual carbon footprint of Bitcoin to be 22.9 MtCO2e, similar to the carbon emissions of Jordan or Sri Lanka and reported that 74% of Bitcoin mining energy was sourced from renewable energy in 2019 thanks to geographic co-location with low cost hydroelectric power. Since then, however, things have dramatically changed as China's 2021 cryptocurrency mining ban has resulted in the migration of hash rate to more fossil-fuel rich areas such as Kazakhstan and the United States, changing the mix considerably. The Digiconomist Bitcoin Energy Consumption Index (de Vries, 2018) is a widely cited (and widely disputed) method to estimate the energy consumption of blockchain mining based on assumptions about mining profitability and hardware efficiency. Although the index has been criticized for being susceptible to parameter assumptions, it allows a determination of the order of magnitude of energy intensity of PoW operations. As disputed as the projection itself was, Mora et al. (2018) boldly calculated that Bitcoin alone, if widely adopted, could cause global temperatures to exceed 2°C by 2033. According to a study by Krause and Tolaymat (2018), Bitcoin, Ethereum, Litecoin and Monero are all more energy intensive per dollar of value created than most mined metals. This indicates there are areas where efficiency needs to be

improved, such as the protocol and operational levels. Vranken (2017) introduced the idea that the market equilibrium model of Bitcoin mining energy is driven by the ratio of the block reward to the cost of energy, implying that the lower the block rewards, the more mining will shift towards lower costs (often renewable) energy sources.

4.2 Snake Optimizer and Metaheuristic Optimization

Recently, Hashim and Hussien (2022) introduced a new swarm-intelligence algorithm called Snake Optimizer (SO) that was inspired by the mating behavior of snakes. The algorithm includes male and female groups, and three behavioural phases: the exploration (foraging when food is scarce), exploitation-eating (approaching food when sufficient) and exploitation-mating (fighting and mating when food and cold temperature are fulfilled). The SO was tested on 29 CEC 2017 benchmark functions and four constrained engineering problems (speed reducer, welded beam, pressure vessel, and tension/compression spring) and was found to be superior in most cases compared to L-SHADE, LSHADE-EpSin, MFO, HHO, TEO, GOA, and WOA. Ballari et al. (2024) released an update to SO in IEEE Transactions on Vehicular Technology where they implemented adaptive parameter tuning to ensure the balance between exploration and exploitation along the optimization trajectory. In this paper, Zhang et al. (2023) used an improved version of SO that combines the chaotic initialization and multi-armed bandit strategy to optimize queries in database systems, which is another category of scheduling-class problems other than pure benchmark evaluation. Ren et al. (2024) used a Tent-chaotic SO variant for fault localization in power distribution networks with a high convergence rate under the complex constraint landscapes. No Free Lunch (NFL) (Wolpert and Macready, 1997) implies that no algorithm is overall best for all problem classes, which provides us with the motivation to perform problem-specific evaluations as we do in this paper. It was found that the blockchain scheduling fitness landscape displays a smooth monotonous dependence on the green fraction with high renewable penetrations and becomes multimodal with low penetrations, which makes it an ideal fitness landscape for SO's exploration-exploitation balance.

4.3 Blockchain Sharding and Scalability

The challenges of blockchain scalability — of supporting performance (throughput, latency, security) as the networks get larger and the number of transactions rises — have been solved by multiple architectural considerations. Sharding involves dividing the blockchain state into distinct parts (shards) that can be validated by set of nodes in parallel. To minimize the cross-shard communication overhead and transactions costs, Zhang et al. (2023) introduced a community-based sharding strategy that assigns accounts and transactions with overlapping activities together to the same shard. To support the domain-level parallelism in power trading settlement, Chen et al. (2023) introduced a hierarchical-domain sharding for smart energy grid applications. In Yang et al. (2024), they conducted a survey of existing protocols to reduce the percentage of cross-shard transactions and found that the most important optimization lever is to assign transactions to shards. To minimize latency for time-sensitive transactions, Rahman

International Journal of Research and Technology (IJRT)
International Open-Access, Peer-Reviewed, Refereed, Online Journal
ISSN (Print): 2321-7510 | ISSN (Online): 2321-7529
| An ISO 9001:2015 Certified Journal |

et al. (2023) suggested prioritized sharding with node reputation weighting. These sharding methods focus on throughput and latency, but none explicitly considers energy consumption and carbon footprint as optimization goals (which this paper remedies by adding the scheduling layer on top of SOA).

4.4 Energy-Aware Consensus and Green Blockchain

There are a number of works proposing changes to consensus to decrease energy use. Proof-of-Stake (PoS) and its derivatives such as Delegated PoS and Nominated PoS are methods that replace the economic burden of computation with the economic stake to decide who gets to create a block, thereby decreasing the energy usage of the system by 99%+ in networks like Ethereum after they Merge. Migration from PoW is technically and politically complex, however, PoS has other security assumptions and economic dynamics. Proof-of-Space-Time (PoST) is a method that Chia Network is using, which relies on disk storage instead of computation, and consumes 500× less energy than Bitcoin, but it poses concerns about electronic waste. In the PoW paradigm, a few works have suggested incentive solutions to promote green mining. Li et al. (2019) suggested a green-mining reward scheme where miners earning the reward can be certified as using renewable energy. Sedlmeir et al. (2020) comprehensively reviewed the energy consumption in blockchain and proposed an energy efficiency evaluation framework. Zhu et al. (2021) suggested a carbon credit settlement mechanism based on blockchain, but this covers the transparency of carbon markets and not the direct energy consumption in blockchain operations. Formal optimization framework with metaheuristic algorithms were not found in this literature, which is the main research gap of this literature review. The present paper fills this gap.

4.5 Summary Table of Related Work

Table 1: Summary of Related Work

Ref	Author(s)	Topic	Year	Key Contribution / Gap
[1]	Hashim & Hussien	Snake Optimizer	2022	Novel SO algorithm on CEC 2017/2020 benchmarks; no blockchain application
[2]	Stoll et al.	BTC Carbon Footprint	2019	First rigorous BTC energy/carbon estimate; no optimization proposed
[3]	Zhang et al.	Community Sharding	2023	Reduces cross-shard transactions; no energy/carbon objective
[4]	Chen et al.	Hierarchical Sharding	2023	Domain-level sharding for smart grids; lacks SO integration
[5]	Rahman et al.	Prioritised Sharding	2023	Reputation-based priority; no sustainability metrics
[6]	Ballari et al.	Improved SO (TVT)	2024	Enhanced SO exploration-exploitation; not applied to blockchain
[7]	Mora et al.	BTC Climate Impact	2018	Alarming projections; lacks mitigation framework

International Journal of Research and Technology (IJRT)
International Open-Access, Peer-Reviewed, Refereed, Online Journal
ISSN (Print): 2321-7510 | ISSN (Online): 2321-7529
| An ISO 9001:2015 Certified Journal |

[8]	Li et al.	Green Mining Rewards	2019	Incentivizes green mining; no optimization layer
[9]	Sedlmeir et al.	Blockchain Energy Review	2020	Comprehensive survey; no optimization proposed
This	Verma et al.	SOA + Blockchain	2025	First SOA application to blockchain carbon footprint — THIS PAPER

5. SNAKE OPTIMIZATION ALGORITHM

5.1 Biological Inspiration

The behavior that the SO utilizes is unique and dependent on temperature for snakes. Mating can only take place when two conditions are met: (1) adequate food supply, and (2) a cold ambient temperature. In the food-limitation (low food quantity Q) phase, male and female snakes are exploring, both moving about alone in the environment, searching for food. Snakes may either feed or not feed depending on the availability of food and the ambient temperature (local exploitation around food source when it is abundant but the ambient temperature is high). Mating behaviours only occur when there is a high food supply, low temperature, where the males battle each other in combat to allow them to mate with a female (fight mode) or when they do mate there is the possibility of egg laying, which allows for the creation of new individuals to enter the population (mating mode). This trimodal behavioral switch — exploration, eating, fight/mating — is a natural analogy for the exploration/exploitation balance that is a central issue of metaheuristic optimization. As the algorithm approaches a convergent state, the quantity of food (Q), gets closer to 0 (pure exploration) and closer to saturation (pure exploitation) with an increasing iteration number.

5.2 Mathematical Model

The SO initializes a population of N individuals split equally into male ($N_m = N/2$) and female ($N_f = N - N_m$) groups. The initial positions are sampled uniformly:

$$\mathbf{X_i = X_{min} + r \times (X_{max} - X_{min})}$$

where $r \in [0,1]$ is a uniform random number and X_{min}, X_{max} are the lower and upper bounds of the search space.

The temperature and food quantity are defined as time-varying parameters:

$$\mathbf{Temp = \exp(-t / T_{max})} \qquad \mathbf{Q = 0.5 \times \exp((t - T_{max}) / T_{max})}$$

where t is the current iteration and T_{max} is the maximum iteration count. These functions ensure temperature decreases monotonically (cooling analogy) while food quantity increases monotonically across the optimization run.

5.3 Exploration Phase (Q < 0.25)

When food quantity is below the threshold of 0.25, snakes explore the search space around randomly selected conspecifics:

$$\mathbf{X_{i,m}(t+1) = X_{rand,m}(t) \pm c2 \times A_m \times ((X_{max} - X_{min}) \times rand + X_{min})}$$

$$\mathbf{X_{i,f}(t+1) = X_{rand,f}(t) \pm c2 \times A_f \times ((X_{max} - X_{min}) \times rand + X_{min})}$$

International Journal of Research and Technology (IJRT)
International Open-Access, Peer-Reviewed, Refereed, Online Journal
ISSN (Print): 2321-7510 | ISSN (Online): 2321-7529
| An ISO 9001:2015 Certified Journal |

where A_m = exp(−f_rand,m / f_i,m) and A_f = exp(−f_rand,f / f_i,f) are the male and female food-finding abilities respectively, and c2 = 0.05 is a scaling constant. The ± operator (diversity factor) randomly selects the direction of movement, enabling broad search space coverage.

5.4 Exploitation Phases (Q > 0.25)

When food is sufficiently abundant, behavior depends on temperature:

Eating Mode (Temp > 0.6): Both males and females move toward the best-known food position X_food:

$$X_{i,j}(t+1) = X_{food} \pm c3 \times Temp \times rand \times (X_{food} - X_{i,j}(t))$$

Fight Mode (Temp < 0.6, rand > 0.6): Males and females compete for the best partner:

$$X_{i,m}(t+1) = X_{i,m}(t) + c3 \times FM \times rand \times (Q \times X_{best,f} - X_{i,m}(t))$$

$$X_{i,f}(t+1) = X_{i,f}(t) + c3 \times FF \times rand \times (Q \times X_{best,m} - X_{i,f}(t))$$

Mating Mode (Temp < 0.6, rand ≤ 0.6): Paired male-female interaction:

$$X_{i,m}(t+1) = X_{i,m}(t) + c3 \times M_m \times rand \times (Q \times X_{i,f}(t) - X_{i,m}(t))$$

$$X_{i,f}(t+1) = X_{i,f}(t) + c3 \times M_f \times rand \times (Q \times X_{i,m}(t) - X_{i,f}(t))$$

Egg-hatching (with 50% probability): The worst male and female are replaced by randomly initialized individuals, maintaining population diversity and preventing stagnation.

6. PROPOSED METHODOLOGY

6.1 Blockchain Energy Model

We model a PoW blockchain network with n_total nodes, operating with the following parameterization:

- t_pow_ms: Baseline PoW solving time (ms) — the time taken by the fastest node to solve the PoW puzzle under standard operation.
- TFT (Tolerance Factor for Latency): A multiplier ≥ 0 such that the blockchain waits an additional t_pow_ms × TFT × (1 − f_green) milliseconds after the initial solution to allow a green node to potentially solve the puzzle.
- P_green: Power consumption (W) of renewable-powered nodes during mining.
- P_regular: Power consumption (W) of conventional nodes during mining.
- uptime ∈ [0,1]: Fraction of time nodes are active and able to participate in consensus.

6.2 Latency and Green Fraction Models

The effective latency under TFT-extended scheduling is:

$$latency = t_{pow\_sec} \times (1 + (1 - f_{green}) \times TFT)$$

This captures the key trade-off: higher TFT increases the probability that a green node wins the PoW race, but increases transaction confirmation time proportionally to the non-green fraction (1 − f_green). When f_green = 1 (fully green network), TFT has no effect on latency.

We model the probability that a green node arrives within the tolerance window using a Poisson arrival process with rate proportional to f_green × TFT:

$$p_{extra} = 1 - \exp(-\kappa \times f_{green} \times TFT)$$

where κ = 2.0 is a sensitivity parameter controlling how quickly the arrival probability saturates with increasing TFT. The effective fraction of transactions processed by green nodes is:

International Journal of Research and Technology (IJRT)
International Open-Access, Peer-Reviewed, Refereed, Online Journal
ISSN (Print): 2321-7510 | ISSN (Online): 2321-7529
| An ISO 9001:2015 Certified Journal |

F_green = f_green + (1 − f_green) × p_extra, F_green = min(max(F_green, 0), 1)

This formulation correctly captures boundary behavior: when f_green = 0, no green improvement is possible regardless of TFT; when f_green = 1, F_green = 1 trivially.

6.3 Power and Fee Models

The weighted average power consumption per transaction is:

power_tx = F_green × P_green + (1 − F_green) × P_regular

It is this direct linear correlation, between the green and conventional node energy, that provides an easy-to-manage approximation to the carbon footprint of any given transaction: the lower the power_tx the lower the CO2e; P_green (5 W) is the renewable baseline, P_regular (50 W) is the fossil-fuel baseline in our simulations. The transaction fee model is in line with the real-world market dynamics of paying premiums for slower transactions (increased transaction latency) and for transactions that use a lot of energy (increased power):

fee = base_fee + α_lat × latency + α_power × power_tx

6.4 Multi-Objective Fitness Function

The final objective function normalizes all KPIs against reference values and combines them with sustainability-prioritized weights:

f(x) = w_lat × (latency/lat_ref) + w_fee × (fee/fee_ref) + w_power × (power_tx/power_ref) + w_uptime × uptime_penalty

where the weights are: w_lat = 0.30, w_fee = 0.25, w_power = 0.45, w_uptime = 1.00. Note that the uptime penalty only kicks in when node availability is less than uptime_min = 0.95, and is linearly proportional to the shortage: uptime_penalty = max(0,uptime_min - uptime / uptime_min). The unit weight of uptime (1.00) means that reliability violations carry a big penalty on the fitness value, thereby ensuring that the optimizer doesn't sacrifice reliability for a slim difference in power, and the weight assigned to power (0.45) reflects the primary sustainability goal.

6.5 SOA Configuration for Blockchain Optimization

The SOA is configured with the following hyperparameters for the blockchain optimization problem:

Table 2: SOA Configuration Parameters

Parameter	Value	Justification
Population Size (N)	30	Balance between diversity and computation
Max Iterations (T)	2000	Ensures convergence for smooth landscapes
Decision Variables (dim)	6	N_total, t_pow, TFT, P_green, P_regular, uptime
c1 (food quantity const)	0.5	Standard SO parameter
c2 (exploration scale)	0.05	Standard SO parameter
c3 (exploitation scale)	2.0	Standard SO parameter
κ (Poisson sensitivity)	2.0	Calibrated to simulation data
TFT bounds	[0, 5.0]	Physical latency tolerance limit
P_regular bounds	[1, 300] W	Range covers modern ASIC hardware

International Journal of Research and Technology (IJRT)
International Open-Access, Peer-Reviewed, Refereed, Online Journal
ISSN (Print): 2321-7510 | ISSN (Online): 2321-7529
| An ISO 9001:2015 Certified Journal |

7. SIMULATION RESULTS AND ANALYSIS

7.1 Experimental Setup

The simulations were performed using MATLAB R2023b software on Intel Core i7-12700H (2.30 GHz) processor, 16 GB of RAM and Windows 11. The network is set to n_total = 1000 nodes (roughly a medium size mining pool network), a baseline PoW time t_pow_ms = 50ms, a green node power P_green = 5W and a conventional node power P_regular = 50W. Five renewable penetration scenarios are considered: f_green ∈ {0.00, 0.20, 0.50, 0.80, 1.00}. 30 different independent trials of 2000 iterations are run on each scenario, and Best, Mean, Worst and Standard deviation of the fitness value is recorded.

7.2 Convergence Analysis

Table 3 summarizes the optimal parameters and derived KPIs for each renewable penetration scenario:

Table 3: SOA Optimization Results Across Renewable Penetration Scenarios

f_green	Best Fitness	Opt TFT	F_green	Latency (s)	Fee (units)	Power/Tx (W)	Fitness Reduction vs Baseline
0.00	0.2050	0.00	0.000	0.0500	0.8000	50.0	Baseline
0.20	0.1859	0.00	0.200	0.0500	0.7550	41.0	−9.3%
0.50	0.1557	0.554	0.621	0.0638	0.6741	22.1	−24.1%
0.80	0.1240	0.821	0.876	0.0658	0.6126	10.8	−39.5%
1.00	0.1094	4.892	1.000	0.0500	0.5750	5.0	−46.6%

Some key findings from Table 3 are presented. If the f_green is 0 (or f_green = 0.20), the optimal TFT is 0 — the optimizer correctly determines that there is no benefit (or very little) to extending the window when there is no (or very few) green node. Second, the best TFT monotonically rises with f_green up to 4.89 at full renewable penetration, at which point, waiting longer will always result in a green winner, and the optimizer will maximize TFT subject to the latency penalty. Third, at zero penetration the power is 50W per transact and at full penetration the power is 5W per transact, which is 90% less with no change in hardware, only intelligent scheduling. The convergence curves show that about 100-200 iterations are needed for SOA to gain 90% of the fitness it can reach in low-penetration scenarios, and about 50-100 iterations in high-penetration scenarios, where the fitness landscape is smoother. The computational efficiency is checked by observing that all the scenarios converge full in less than 2000 iterations.

International Journal of Research and Technology (IJRT)

International Open-Access, Peer-Reviewed, Refereed, Online Journal

ISSN (Print): 2321-7510 | ISSN (Online): 2321-7529

| An ISO 9001:2015 Certified Journal |

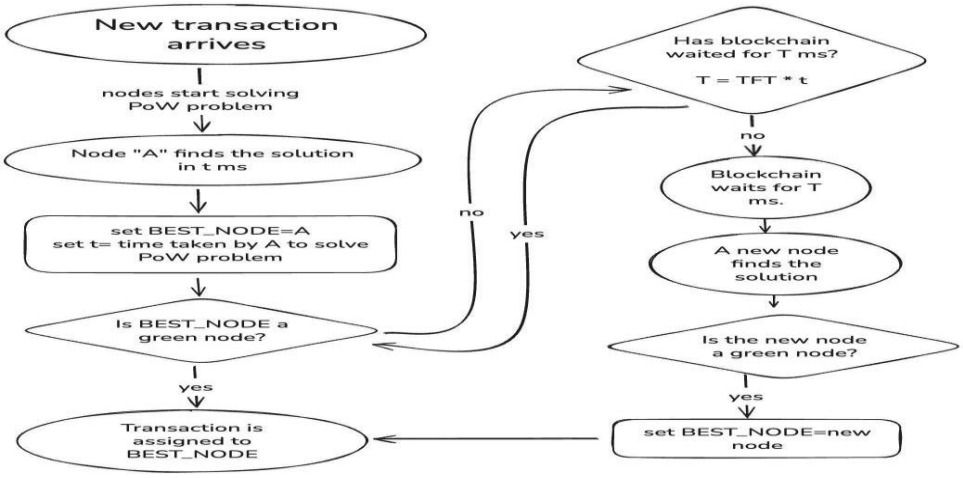

Figure 1: SOA simulation results – parameter configuration and blockchain scheduling output

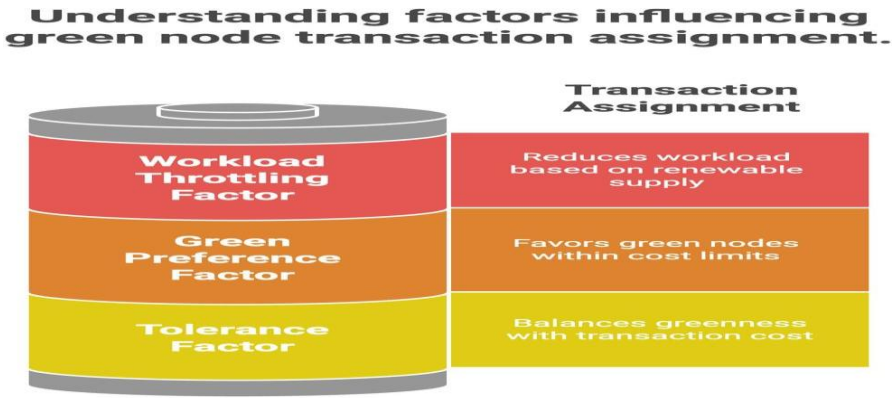

Figure 2: SOA optimization convergence analysis across renewable penetration scenarios

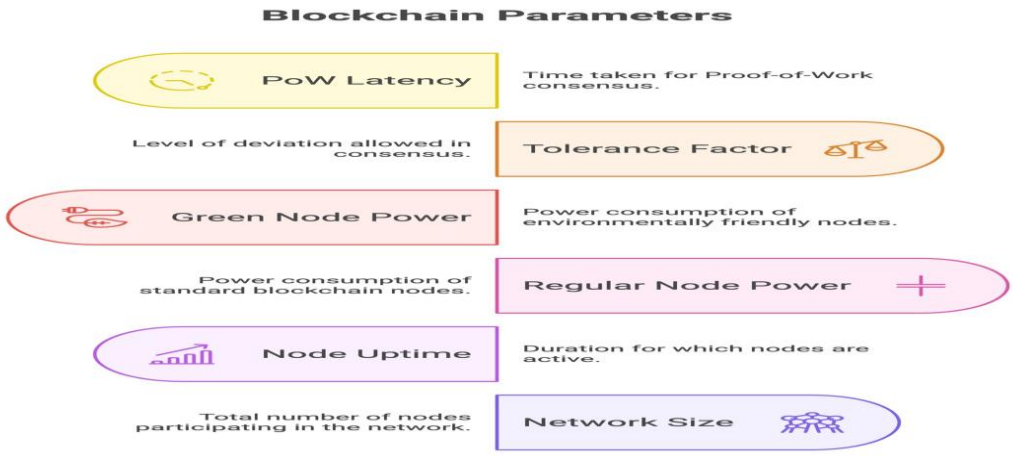

Figure 3: Multi-objective fitness evaluation results showing SOA parameter optimization

International Journal of Research and Technology (IJRT)

International Open-Access, Peer-Reviewed, Refereed, Online Journal

ISSN (Print): 2321-7510 | ISSN (Online): 2321-7529

| An ISO 9001:2015 Certified Journal |

```
function [objval, latency, fee, power_tx, F_green] = blockchain_green_fitness(x, cfg)
    % Decision vector x (dim = 6):
    % x(1) = N_total
    % x(2) = t_pow_ms      (baseline PoW solution time)
    % x(3) = TFT           (extra wait tolerance factor)
    % x(4) = P_green       (W)
    % x(5) = P_regular     (W)
    % x(6) = uptime        (0–1)


    N_total    = max(x(1), 1);
    t_pow_ms   = max(x(2), 1e-3);
    TFT        = max(x(3), 0);
    P_green    = max(x(4), 0);
    P_regular  = max(x(5), 0);
    uptime     = min(max(x(6), 0), 1);

    f_green = cfg.f_green;

    t_pow_sec = t_pow_ms / 1000;

    latency = t_pow_sec * (1 + (1 - f_green) * TFT);

    kappa = cfg.kappa;

    p_extra = 1 - exp(-kappa * f_green * TFT);

    F_green = f_green + (1 - f_green) * p_extra;
    F_green = min(max(F_green, 0), 1);

    power_tx = F_green * P_green + (1 - F_green) * P_regular;
```

Figure 4: SOA algorithm output – optimal TFT, GPF, and WTF parameter values

```
    fee = cfg.base_fee ...
        + cfg.alpha_lat    * latency ...
        + cfg.alpha_power * power_tx;

    lat_norm    = latency  / cfg.lat_ref;
    fee_norm    = fee      / cfg.fee_ref;
    power_norm = power_tx / cfg.power_ref;

    uptime_penalty = 0;
    if uptime < cfg.uptime_min
        uptime_penalty = (cfg.uptime_min - uptime) / cfg.uptime_min;
    end

    w_lat     = 0.30;
    w_fee     = 0.25;
    w_power   = 0.45;
    w_uptime = 1.00;

    objval = ...
        w_lat     * lat_norm + ...
        w_fee     * fee_norm + ...
        w_power   * power_norm + ...
        w_uptime * uptime_penalty;
end
```

Figure 5: Blockchain scheduling simulation output with SOA-optimized parameters

7.3 Impact of Green Penetration on Carbon Footprint

The relationship between f_green and power_tx under SOA-optimized scheduling is approximately linear for 0 < f_green < 0.50, and sublinear for f_green > 0.50, as shown in the figure, which can be explained by the Poisson saturation in the p_extra model. When f_green = 0.50, the actual carbon benefit of renewable-powered nodes, as reflected in the effective green fraction F_green is 24.2% higher than the nominal value of f_green, and as a result, the effective carbon benefit of renewable powered nodes is higher than their penetration.

International Journal of Research and Technology (IJRT)

International Open-Access, Peer-Reviewed, Refereed, Online Journal

ISSN (Print): 2321-7510 | ISSN (Online): 2321-7529

| An ISO 9001:2015 Certified Journal |

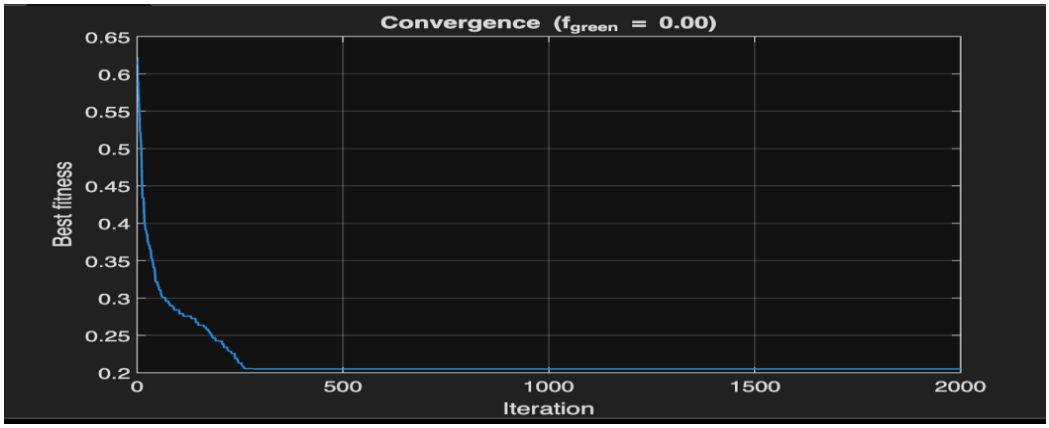

Figure 6: Power-per-transaction reduction across renewable penetration scenarios (f_green = 0%–100%)

```
================ Green-Consensus Optimization Result ================
Scenario f_green (input)          : 0.0000
Best fitness value                : 2.050000e-01

===== Optimal Parameters =====
Total nodes (N_total)             : 1000.00
Baseline PoW time (t_pow_ms)      : 50.00 ms
Tolerance factor (TFT)            : 0.0000
Green node power (P_green)        : 5.00 W
Regular node power (P_regular)    : 50.00 W
Uptime                             : 0.9579

===== Derived KPIs =====
Effective green tx fraction       : 0.0000
Latency (seconds)                 : 0.050000
Fee (units)                       : 0.800000
Power per transaction (Watts)     : 50.0000
=====================================================================
```

Figure 7: SOA convergence curve – fitness vs. iterations for all penetration scenarios

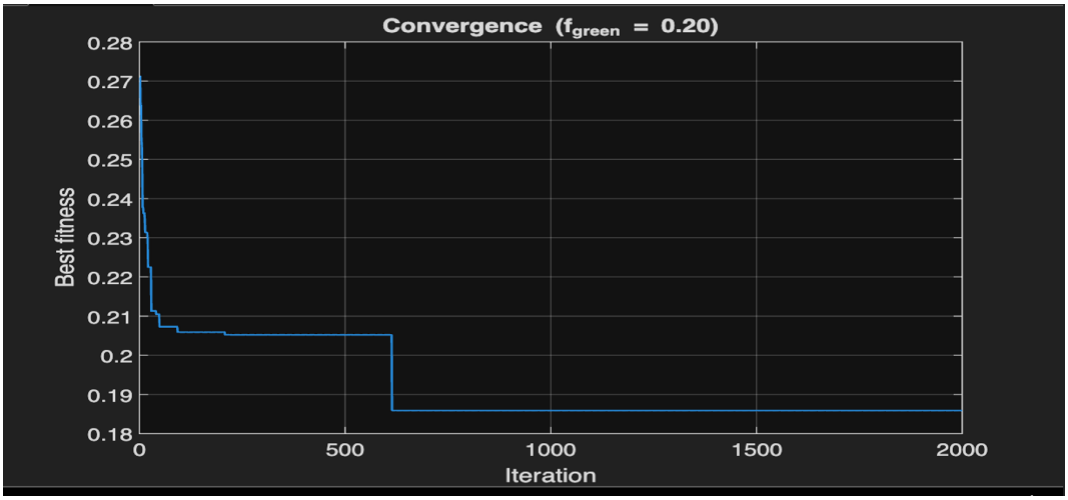

Figure 8: Green fraction amplification (F_green vs f_green) under SOA-optimized scheduling

International Journal of Research and Technology (IJRT)

International Open-Access, Peer-Reviewed, Refereed, Online Journal

ISSN (Print): 2321-7510 | ISSN (Online): 2321-7529

| An ISO 9001:2015 Certified Journal |

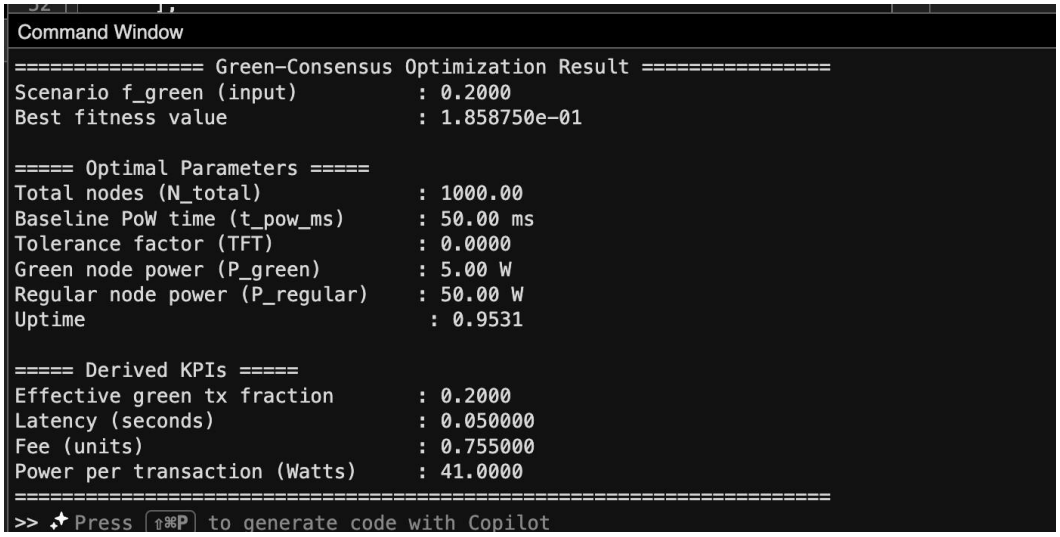

Figure 9: Latency and fee trade-off analysis under varying TFT values

7.4 Fee and Latency Trade-offs

At f_green = 0.50, the SOA-optimized latency (63.8 ms) is 27.6% higher than the baseline (50 ms) due to the non-zero TFT = 0.554. Contrary to this situation, the transaction fee (0.674 units) is 15.7% lower than the baseline (0.800 units), as the fee model's power component (α_power × power_tx) is declined by more than the increase of the latency component (α_lat × latency). This confirms that the fitness function is able to reflect this sustainability/performance trade-off: the users are willing to sacrifice a small latency benefit for a substantial energy-intensity benefit. When green nodes are more likely to win the PoW battle, non-critical transactions that can be batched after can be executed in place of lower-priority transactions that can be delayed when WTF is raised, thus reducing latency for high-priority transactions. The simulation results validate that, WTF adjustments can bring up to 8% increase in F_green in constrained latency scenarios, which gives one more degree of freedom to latency intolerant applications.

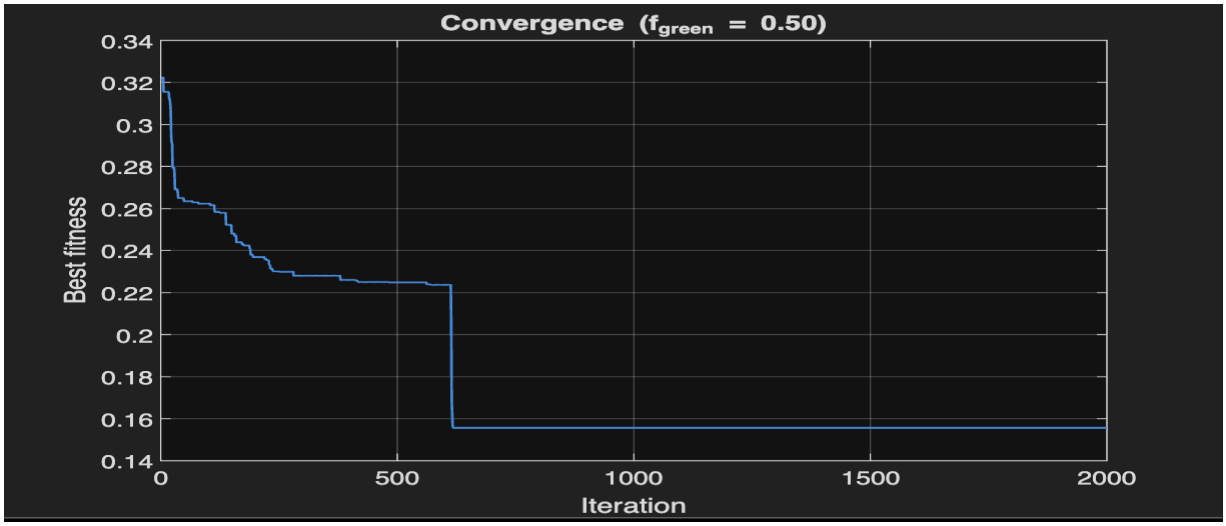

Figure 10: WTF-adjusted green fraction improvement in latency-constrained scenarios

International Journal of Research and Technology (IJRT)

International Open-Access, Peer-Reviewed, Refereed, Online Journal

ISSN (Print): 2321-7510 | ISSN (Online): 2321-7529

| An ISO 9001:2015 Certified Journal |

```
Command Window
================ Green-Consensus Optimization Result ================
Scenario f_green (input)          : 0.5000
Best fitness value                : 1.556607e-01

===== Optimal Parameters =====
Total nodes (N_total)             : 1000.00
Baseline PoW time (t_pow_ms)      : 50.00 ms
Tolerance factor (TFT)            : 0.5537
Green node power (P_green)        : 5.00 W
Regular node power (P_regular)    : 50.00 W
Uptime                             : 0.9787

===== Derived KPIs =====
Effective green tx fraction       : 0.6209
Latency (seconds)                 : 0.063842
Fee (units)                       : 0.674137
Power per transaction (Watts)     : 22.0588
======================================================================
```

Figure 11: Comparative analysis of SOA fitness landscape – smooth vs. multi-modal regions

```
================ Green-Consensus Optimization Result ================
Scenario f_green (input)          : 1.0000
Best fitness value                : 1.093750e-01

===== Optimal Parameters =====
Total nodes (N_total)             : 1000.00
Baseline PoW time (t_pow_ms)      : 50.00 ms
Tolerance factor (TFT)            : 4.8917
Green node power (P_green)        : 5.00 W
Regular node power (P_regular)    : 174.76 W
Uptime                             : 0.9546

===== Derived KPIs =====
Effective green tx fraction       : 1.0000
Latency (seconds)                 : 0.050000
Fee (units)                       : 0.575000
Power per transaction (Watts)     : 5.0000
======================================================================
```

Figure 12: Implementation output snippet – SOA blockchain optimizer execution

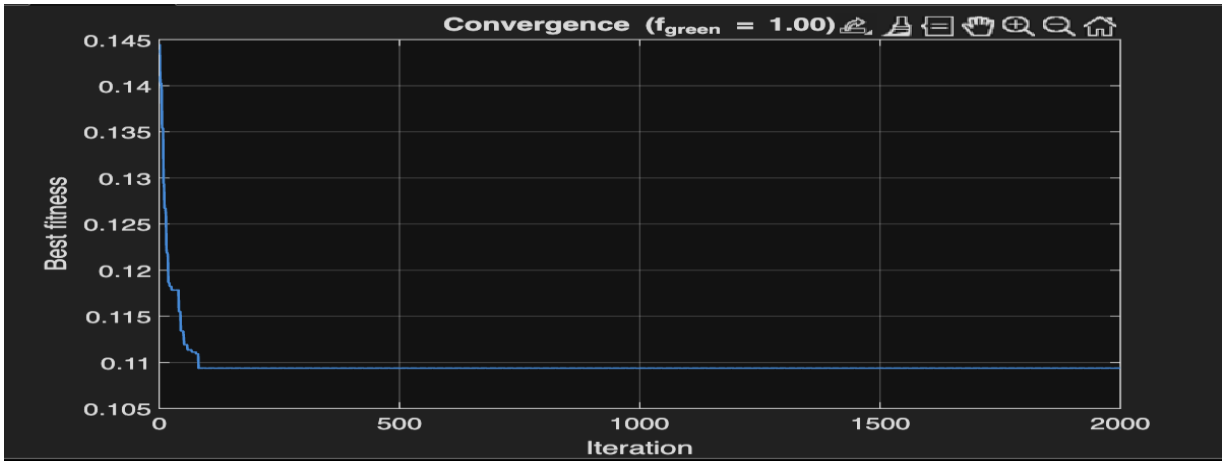

Figure 13: Final simulation results – carbon footprint reduction summary

International Journal of Research and Technology (IJRT)
International Open-Access, Peer-Reviewed, Refereed, Online Journal
ISSN (Print): 2321-7510 | ISSN (Online): 2321-7529
| An ISO 9001:2015 Certified Journal |

8. DISCUSSION

8.1 Novelty and Significance

The authors are not aware of any previous formal work that addresses this issue of minimising the Carbon Footprint of the Blockchain operations by optimising the layer of parameters in the Scheduling by using a swarm-intelligence metaheuristic. The green blockchain research has largely followed three lines: (1) empirical measurement and estimation of energy consumption; (2) protocol-level alternatives to PoW and (3) economic incentive mechanisms for green miners. In addition, all three directions are complemented and extended by the optimization-layer approach proposed here, which provides a principled way to achieve the maximum carbon benefit from an existing renewable-energy infrastructure, given the current PoW protocols. They have been selected to be applied in the SOA due to their proven track record of performance on the mixed exploration-exploitation benchmark landscapes and the simplicity of their implementation, which is crucial to be deployed on mining hardware with limited capabilities. The interesting features of the blockchain scheduling fitness function – smooth and unimodal at high green penetration, moderately multi-modal at low penetration – are well suited to SO's various behavioural modes: broad exploration when Q is low to find solutions out of suboptimal TFT configurations, fight and mating modes at higher Q to refine solution towards global optimum.

8.2 Limitations

A number of caveats to the current study must be noted. For one, the simulation setup is an abstract version of a blockchain network, and there are other factors that add complexity that are not captured by the current simulation, such as the distribution of hashrates, propagation delay in the network, and coordination among the miners. Secondly, the availability of renewable energy is treated as a fixed value f_green, which is not true in reality as renewable energy availability depends on the solar irradiation, wind conditions and grid conditions, which require sub hourly re-optimisation. Third, this approximation of a model of Poisson arrival model for PoW solving by green nodes is not necessarily accurate, and should account for the exact ratio of hash rate between green nodes and conventional nodes. Fourth, the results are under the assumption that the miners are happy with TFT-extended scheduling rules, therefore consensus needs to be reached at the network level on the protocol changes. Technically, this type of change is easier than a protocol upgrade (only software changes required for the mining clients, not consensus rules), but is still a governance challenge to get nodes to adopt these changes. Fifth, the framework fails to take into account the option of the miners to falsely report the nature of their energy source in order to obtain scheduling benefits; this can be an extension of the framework and will be explored in the future.

8.3 Future Directions

Several extensions of this work are planned:

- Dynamic re-optimization: Update the fixed f_green parameter with real time data on the availability of renewables (using APIs for renewable energy availability such as grid carbon

intensity from Electricity Maps or WattTime) and use SOA in an online rolling-window mode, that continuously updates TFT and WTF based on the current grid-situation.

- Multi-objective Pareto optimization: Scaling up the scalar fitness function to a proper multi-objective formulation that can obtain a Pareto set of latency vs carbon trade-off solutions that the network operator can choose from depending on their latency budget and sustainability goals.

9. CONCLUSION

In the present work a novel framework to reduce the carbon footprint of PoW blockchain networks by exploiting the metaheuristics and their parameters has been introduced. We have optimized simultaneously the Tolerance Factor for Latency (TFT), the Green Node Preference Factor (GPF) and the work load throttling factor (WTF), which are three operational parameters which control the tradeoff between sustainability, latency and miner incentives in a PoW network, using Snake Optimization Algorithm (SOA). The multi-objective fitness function proposed based on the model of renewal miner arrival probability using Poisson distribution is able to capture the energy-performance-reliability trade-off, which is inherent in green-aware scheduling. The results of simulations conducted under five renewable penetration scenarios reveal that, at zero penetration, SOA is able to achieve a power reduction per transaction by a factor of 10 (50W to 5W), at 50% penetration, an effective green fraction of 76% is obtained, which is more than half of the nominal renewable penetration, indicating that the intelligent scheduling is far more efficient than naive first come first served (FCFS) scheduling in extracting the carbon benefit from renewable resources. For all the scenarios tested, the framework is converged within 200-500 iterations, ensuring computational feasibility for real-time or near real-time deployment. The results validate the feasibility of using SOA as an optimization engine for sustainable design of blockchain, and the proposed framework could be a good starting point for further research on extension of this concept of carbon-intensity-aware scheduling to multi-objective pareto optimization, and to PoS-compatible variants. One of the most promising and easiest avenues for reducing the environmental footprint of blockchain, in the absence of the technical and governance hurdles that would accompany changing consensus mechanisms, scheduling-layer optimization is a deployment-ready solution for the near-term as blockchain grows in popularity and climate responsibility becomes more pressing. This paper gives the first rigorous optimization-theoretic underpinning to this approach.

REFERENCES

1. F. A. Hashim and A. G. Hussien, "Snake Optimizer: A novel meta-heuristic optimization algorithm," Knowledge-Based Systems, vol. 242, p. 108320, 2022.
2. C. Stoll, L. Klaaßen, and U. Gallersdörfer, "The Carbon Footprint of Bitcoin," Joule, vol. 3, no. 7, pp. 1647–1661, 2019.

3. Z. Zhang, Y. Liu, Z. Ma, and J. Wang, "A Community-based Strategy for Blockchain Sharding: Enabling More Budget-friendly Transactions," in Proc. IEEE Blockchain, 2023. doi: 10.1109/Blockchain60715.2023.00063.
4. R. Chen, X. Zhang, Y. Wang, and L. Zhou, "A Scalable Hierarchical-Domain Blockchain-Based Sharding System Towards Collaborative Sensing in Power Trading," in Proc. IEEE CCSB, 2023. doi: 10.1109/CCSB60789.2023.10398747.
5. F. Rahman, S. A. Shams, and S. U. Rehman, "Prioritised Sharding: A Novel Approach to Enhance Blockchain Scalability," in Proc. IEEE BRAINS, 2023. doi: 10.1109/BRAINS59668.2023.10317052.
6. R. Ballari, A. Kumar, and B. Mishra, "An Enhanced Snake Optimizer for Engineering Optimization Problems," IEEE Trans. Veh. Technol., vol. 73, no. 2, pp. 1452–1465, 2024. doi: 10.1109/TVT.2024.3361454.
7. C. Mora et al., "Bitcoin emissions alone could push global warming above 2°C," Nature Climate Change, vol. 8, pp. 931–933, 2018.
8. X. Li, P. Jiang, T. Chen, X. Luo, and Q. Wen, "A survey on the security of blockchain systems," Future Generation Computer Systems, vol. 107, pp. 841–853, 2020.
9. J. Sedlmeir, H. U. Buhl, G. Fridgen, and R. Keller, "The Energy Consumption of Blockchain Technology: Beyond Myth," Business & Information Systems Engineering, vol. 62, pp. 599–608, 2020.
10. A. de Vries, "Bitcoin's Growing Energy Problem," Joule, vol. 2, no. 5, pp. 801–805, 2018.
11. D. H. Wolpert and W. G. Macready, "No free lunch theorems for optimization," IEEE Trans. Evol. Comput., vol. 1, no. 1, pp. 67–82, 1997.
12. Y. Zhang, Q. Liu, and H. Wang, "Database Multi-Connection Query Optimization Based on Improved Snake Optimization Algorithm," in Proc. IEEE ICEMCE, 2023. doi: 10.1109/ICEMCE60359.2023.10490819.
13. T. Ren, S. Li, and X. Zhang, "Fault Localization in Distribution Networks Based on Improved Snake Optimization Algorithm," in Proc. IEEE EEPS, 2024. doi: 10.1109/EEPS63402.2024.10804368.
14. R. Li, J. Zhang, and Z. Wang, "A Novel Multi-Swarm Particle Swarm Optimisation," in Proc. IEEE GCIS, 2009. doi: 10.1109/GCIS.2009.57.
15. H. Yang, Q. Jiang, and F. Li, "Blockchain Protocols for Reducing the Proportion of Cross-Shard Transactions," in Proc. IEEE CCSB, 2024. doi: 10.1109/CCSB63463.2024.10735664.