Simulating FaaS Applications at the Edge for IoT Workloads:
Energy-Aware Scaling Strategies in Heterogeneous Computing Environments

Mahfoud Abd el Ali SAYAH

Academic Year 2024-2025

Contents

Chapter 1

General Introduction

(8-10 pages)

1.1 Introduction

- Context of serverless computing growth and FaaS popularity
- Edge computing paradigm and resource-constrained environments
- Convergence opportunity: FaaS at the edge for IoT applications
- Main challenge: shortage of simulation tools for edge FaaS deployments

1.2 Objective

Primary Goal: Develop energy-aware scaling framework for FaaS applications in edge computing environments through simulation

Specific Objectives:

1. Conduct state-of-the-art analysis on edge computing simulators and FaaS tools
2. Identify suitable simulation framework and extend with energy modeling
3. Implement energy-aware scaling strategies for heterogeneous edge devices
4. Validate framework through IoT smart city use case and comparative evaluation
5. Provide deployment recommendations for edge-native serverless computing

1.3 Dissertation Structure

- Chapter 1: Research context, objectives, and methodology overview
- Chapter 2: Foundational concepts in edge computing, FaaS, and energy optimization
- Chapter 3: State-of-the-art analysis and simulation framework selection
- Chapter 4: Implementation, experiments, and results analysis

Chapter 2

Basic Concepts

(18-22 pages)

2.1 Introduction

2.2 Edge Computing Fundamentals

- Definition and key characteristics
- Edge vs. cloud computing paradigms
- Resource constraints and deployment challenges
- Heterogeneous device ecosystems

2.3 Serverless Computing and FaaS

- Serverless computing evolution
- Function-as-a-Service model characteristics
- Event-driven execution and automatic scaling
- Benefits and limitations for edge deployment

2.4 FaaS at the Edge

- Opportunities and challenges
- Resource allocation considerations
- Cold start and warm start implications
- Network latency and bandwidth constraints

2.5 Heterogeneous Edge Devices

- Device taxonomy (RPi, Coral TPU, Jetson devices, Intel NUC)
- Performance and energy characteristics
- Specialized accelerators and general-purpose compute
- Real-world deployment distributions

2.6 Energy Models and Optimization

- Power consumption models (idle vs. dynamic)
- Energy efficiency metrics
- Performance vs. energy trade-offs
- Battery-powered device considerations

2.7 Autoscaling and Resource Management

- Traditional cloud scaling approaches
- Edge-specific scaling challenges
- Load balancing in heterogeneous environments
- Quality of Service requirements

2.8 IoT Applications and Smart Cities

- IoT application characteristics
- Smart city infrastructure requirements
- Traffic monitoring use case
- Real-time processing constraints

2.9 Conclusion

Chapter 3

State of the Art

(15-18 pages)

3.1 Introduction

3.2 Edge Computing Simulation Tools

- CloudSim and EdgeCloudSim
- iFogSim and YAFS (Yet Another Fog Simulator)
- Limitations for FaaS workloads
- Gap analysis for serverless simulation

3.3 FaaS Simulation Frameworks

- SimFaaS: Performance simulator for serverless platforms
- CloudSim extensions for FaaS
- FAAS-Profiler and performance modeling
- Limitations for edge environments

3.4 Edge-FaaS Simulation Tools

- CloudSim extension for distributed functions
- HeroSim and existing solutions
- **faas-sim framework identification and selection**
- Requirements for energy modeling extension

3.5 Energy-Aware Computing Approaches

- Single-objective energy optimization
- Multi-objective optimization approaches
- Scaling strategies in literature

3.6 Research Gap Analysis

- Shortage of edge-FaaS simulation tools
- Lack of energy modeling in existing frameworks
- Missing scaling strategies for heterogeneous devices
- Limited IoT use case validation

3.7 Framework Selection Justification

- Multi-criteria evaluation of existing tools
- faas-sim selection rationale
- Extension requirements identification
- Energy model integration strategy

3.8 Conclusion

Chapter 4

Our Contribution

(25-30 pages)

4.1 Introduction

Overview of research contributions and implementation approach

4.2 Framework Architecture and Extensions

4.2.1 faas-sim Framework Analysis

- Original framework capabilities and limitations
- Architecture analysis and extension points
- Energy modeling gap identification

4.2.2 Energy Model Integration

- Linear power consumption model implementation
- Device-specific energy profiles
- Dynamic vs. static power modeling
- Validation against manufacturer specifications

4.3 Heterogeneous Edge Device Modeling

4.3.1 Device Distribution Strategy

4.3.2 Power Consumption Implementation

- Device-specific power profiles
- Idle and maximum power consumption values
- Utilization-based dynamic power calculation

Table 4.1: Implemented Edge Device Distribution

Device Type	**Percentage**	**Primary Use Case**
Raspberry Pi	35%	IoT sensors, basic monitoring
RockPi	25%	General compute nodes
Coral TPU	15%	AI inference acceleration
Jetson Nano	12%	GPU-based inference
Jetson NX	8%	High-performance AI
Intel NUC	2.5%	Edge coordinators
Jetson TX2	2.5%	Specialized workloads

- Energy efficiency metrics calculation

4.4 Energy-Aware Scaling Strategies

4.4.1 Base Autoscaler Framework

- Template method pattern implementation
- Common scaling decision logic
- Metrics collection system
- Strategy interface design

4.4.2 High Performance Short Time (HPST)

- Philosophy: "high power Œ short time"
- Device prioritization hierarchy
- Implementation details and algorithm
- Use case: performance-critical applications

4.4.3 Low Power Long Time (LPLT)

- Philosophy: "low power Œ long time"
- Energy-first device selection
- Battery-aware considerations
- Use case: energy-constrained deployments

4.4.4 Kubernetes-Style First Fit

- Weighted scoring system implementation
- Device class matching logic
- Resource balancing approach
- Production baseline compatibility

4.4.5 Standard First Fit (Baseline)

- Resource-only allocation
- Comparison baseline implementation
- Traditional container orchestration approach

4.5 Smart City Use Case Implementation

4.5.1 IoT Traffic Monitoring Scenario

- Function types: resnet50-inference, speech-inference, etc.
- Initial deployment strategy (light deployment)
- Geographic distribution modeling
- Realistic workload patterns

4.5.2 Function Deployment Configuration

- 16 initial function instances across 6 types
- Cold-start problem mitigation
- Performance requirements per function
- Resource allocation specifications

4.6 Implementation and Development

4.6.1 Development Environment

- Programming languages (Python, etc.)
- Framework modifications and extensions
- Code organization and architecture

4.6.2 Simulation Configuration

- Experimental parameters
- Device generation and deployment
- Workload simulation setup
- Metrics collection implementation

4.7 Evaluation Methodology

4.7.1 Experimental Design

- Comparative evaluation approach
- Multiple simulation runs
- Statistical analysis methods
- Performance vs. energy metrics

4.7.2 Evaluation Metrics

- Energy Efficiency Score: requests/joule
- Response time percentiles
- Resource utilization patterns
- Scaling decision frequency

4.8 Results and Analysis

4.8.1 Energy Consumption Results

Table 4.2: Energy Consumption Comparison

Strategy	Total Energy (kJ)	Energy/Request (J)	Improvement
LPLT	1,156	2.68	16.8%
HPST	1,245	2.89	10.2%
Kubernetes	1,298	3.01	6.4%
Standard FF	1,387	3.22	Baseline

4.8.2 Performance Analysis

- Response time comparison across strategies
- SLA compliance rates
- Throughput analysis
- Trade-off evaluation

4.8.3 Device Utilization Patterns

- Strategy-specific device preferences
- Resource utilization efficiency
- Energy consumption by device type
- Scaling behavior analysis

4.9 Discussion

4.9.1 Key Findings

- LPLT achieves best energy efficiency (16.8% savings)
- HPST provides optimal performance with energy benefits
- Heterogeneous device modeling enables optimization
- Smart city deployment feasibility demonstrated

4.9.2 Practical Implications

- Strategy selection guidelines
- Deployment recommendations
- Energy-performance trade-off insights
- Framework extensibility for future research

4.9.3 Limitations

- Simulation-based evaluation constraints
- Limited to traffic monitoring use case
- Power model based on specifications
- Network modeling simplifications

4.10 Conclusion

Chapter 5

Conclusion

(4-6 pages)

5.1 Summary of Work

- Research problem and objectives revisited
- State-of-the-art analysis and framework selection
- faas-sim extension with energy modeling
- Four scaling strategies implementation and evaluation

5.2 Key Contributions

- Extended faas-sim with comprehensive energy modeling
- Implemented four distinct energy-aware scaling strategies
- Validated through realistic smart city IoT use case
- Demonstrated significant energy savings (up to 16.8%)
- Provided practical deployment guidelines

5.3 Research Impact

- Enhanced simulation capabilities for edge-FaaS research
- Energy-performance trade-off quantification
- Framework available for future research extensions
- Practical insights for edge computing deployments

5.4 Future Work

- Real-world testbed validation (Grid'5000 deployment)
- Extended use case scenarios beyond traffic monitoring
- Machine learning-based predictive scaling
- Integration with production FaaS platforms
- Advanced energy modeling with thermal considerations

5.5 Final Remarks

Appendix A

Implementation Details

- faas-sim framework modification details
- Key algorithm implementations
- Configuration files and parameters

Appendix B

Experimental Data

- Complete experimental results
- Statistical analysis details
- Additional performance charts

Estimated Page Distribution

Chapter	Pages
Chapter 1: General Introduction	8-10
Chapter 2: Basic Concepts	18-22
Chapter 3: State of the Art	15-18
Chapter 4: Our Contribution	25-30
Chapter 5: Conclusion	4-6
Appendices	8-12
Total	**78-98 pages**

Key Research Flow

Research Journey

1. **Problem Identification:** Shortage of FaaS simulation tools for edge environments
2. **State-of-the-Art Survey:** Analysis of existing simulators and frameworks
3. **Framework Selection:** Chose faas-sim as most suitable base framework
4. **Gap Analysis:** Identified missing energy modeling capability
5. **Energy Model Integration:** Extended framework with power consumption models
6. **Scaling Strategy Implementation:** Developed four energy-aware approaches
7. **Use Case Validation:** Smart city IoT traffic monitoring scenario
8. **Comparative Evaluation:** Energy-performance trade-off analysis
9. **Results and Insights:** Demonstrated significant energy savings potential

Technical Achievements

- Successfully extended faas-sim with realistic energy modeling
- Implemented comprehensive heterogeneous device modeling
- Developed four distinct scaling strategies with clear optimization objectives
- Validated framework through realistic smart city deployment scenario
- Achieved measurable energy improvements while maintaining performance