Network Slicing Simulation Technical Documentation

AI-Generated Content Disclaimer

This document has been generated using GPT-based AI. While efforts have been made to ensure accuracy, the content may contain irrelevant, outdated, or imprecise information. Users are advised to critically evaluate the information provided and verify important details independently. Any inconsistencies or errors should be disregarded.

---

1. Introduction

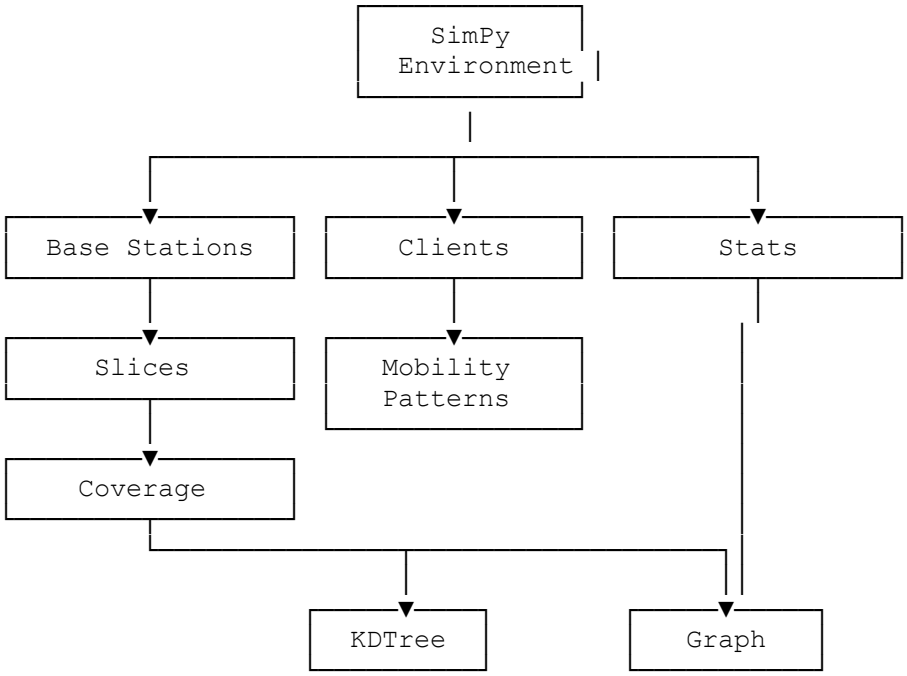
This document provides technical documentation for a Network Slicing Simulation framework. The system simulates 5G network slicing environments, allowing for analysis of resource allocation, client mobility patterns, and quality of service metrics across different network slices.

2. System Architecture

The simulation is built on a discrete event simulation model using the SimPy framework. It consists of several key components:

- **Base Stations**: Physical network infrastructure with coverage areas and bandwidth capacity
- **Slices**: Logical partitions of network resources with specific QoS parameters
- **Clients**: End users with mobility patterns and usage requirements
- **Stats Collector**: System for gathering performance metrics during simulation
- **Visualization**: Real-time and post-simulation graphical representation of results

2.1 Component Relationships



3. Core Components

3.1 Base Station

Base stations represent physical network infrastructure with defined coverage areas and total bandwidth capacity.

Key attributes:

- Unique identifier (`pk`)
- Coverage area (defined by center coordinates and radius)
- Total capacity bandwidth
- Collection of slices

3.2 Network Slice

Slices are logical partitions of network resources with specific QoS parameters.

Key attributes:

- Name

- Resource ratio
- Connected users count
- User share
- Delay tolerance
- QoS class
- Guaranteed bandwidth
- Maximum bandwidth
- Initial capacity
- Usage pattern

Key functionalities:

- Resource allocation/deallocation
- User connection management
- Slice availability checking

3.3 Client

Clients represent end users with mobility patterns and slice subscriptions.

Key attributes:

- Unique identifier (`pk`)
- Position coordinates (x, y)
- Mobility pattern
- Usage frequency
- Subscribed slice index
- Connected state
- Usage metrics

Key functionalities:

- Connection management (connect, disconnect)
- Resource consumption (request, consume, release)
- Movement according to mobility patterns
- Base station assignment and handover

3.4 Coverage

Defines the geographical area covered by a base station.

Key attributes:

- Center coordinates
- Radius

Key functionalities:

- Point-in-coverage checking

3.5 Distributor

Generates random values for various simulation parameters using different probability distributions.

Key functionalities:

- Value generation for various distributions
- Scaled value generation
- Movement vector generation

3.6 Statistics Collection

Gathers and computes performance metrics during the simulation.

Key metrics:

- Connected users ratio
- Total bandwidth usage
- Average slice load ratio
- Average slice client count
- Coverage ratio
- Block ratio
- Handover ratio

3.7 Visualization (Graph)

Provides graphical representation of the simulation state and metrics.

Key features:

- Network map visualization
- Time series charts for key metrics
- Summary statistics table

4. Simulation Lifecycle

4.1 Initialization Phase

1. Load configuration from YAML file
2. Create SimPy environment

3. Initialize base stations with coverage areas
4. Create slices with specified parameters
5. Generate clients with random positions and mobility patterns
6. Assign clients to closest base stations using KDTree

4.2 Simulation Loop

Each client runs through a 4-step cycle repeatedly:

1. **Lock (t + 0.00)**: Attempt to connect or consume resources
2. **Stats (t + 0.25)**: Collect statistics
3. **Release (t + 0.50)**: Release consumed resources
4. **Move (t + 0.75)**: Update position based on mobility pattern

4.3 Statistics Collection

Statistics are collected at regular intervals throughout the simulation.

4.4 Visualization

The visualization component can display:

- Real-time simulation state
- Performance metrics over time
- Summary statistics

5. Key Algorithms

5.1 Client Base Station Assignment

```
1. Build KD-Tree from base station coordinates
2. For each client:
   a. Query KD-Tree for k nearest base stations
   b. Check if client is within coverage radius
   c. Assign client to closest base station in range
   d. Store list of closest base stations for handover
```

5.2 Client Connection Management

```
1. Client attempts to connect to assigned base station
2. If resources available in subscribed slice:
   a. Increment connected users count
   b. Set client state to connected
3. If resources unavailable:
   a. Check alternative base stations
   b. If alternative available, handover
   c. If no alternative, block connection
```

5.3 Resource Consumption

```
1. Client generates usage request based on slice usage pattern
2. Client requests resources from base station
3. Slice allocates consumable share
4. Client consumes resources
5. Client releases resources after consumption
```

5.4 Client Mobility

```
1. At each movement step:
   a. Generate movement vector from mobility pattern
   b. Update client coordinates
   c. Check if client is still in coverage area
   d. If not, disconnect and find new base station
```

6. Configuration Parameters

The simulation can be configured through a YAML file with the following sections:

6.1 General Settings

- `simulation_time`: Duration of simulation
- `num_clients`: Number of clients
- `limit_closest_base_stations`: Maximum number of base stations to track per client
- `statistics_params`: Parameters for statistics collection
- `logging`: Enable/disable logging
- `plotting_params`: Visualization configuration

6.2 Slice Configuration

For each slice:

- `delay_tolerance`: Maximum acceptable delay
- `qos_class`: Quality of service class
- `bandwidth_guaranteed`: Minimum guaranteed bandwidth
- `bandwidth_max`: Maximum bandwidth per client
- `client_weight`: Distribution weight for client assignment
- `threshold`: Resource allocation threshold
- `usage_pattern`: Configuration for usage distribution

6.3 Base Station Configuration

For each base station:

- `x`, `y`: Coordinates

- `coverage`: Coverage radius
- `capacity_bandwidth`: Total bandwidth capacity
- `ratios`: Resource allocation ratios for each slice

6.4 Client Configuration

- `location`: Distribution parameters for client positioning
- `usage`: Distribution parameters for usage amounts
- `usage_frequency`: Distribution parameters for usage frequency

6.5 Mobility Patterns

For each pattern:

- `distribution`: Type of statistical distribution
- `params`: Parameters for the distribution
- `client_weight`: Distribution weight for client assignment

7. Output Analysis

The simulation produces several outputs:

7.1 Statistical Metrics

- Connected clients ratio
- Total bandwidth usage
- Average load factor of slices
- Average client count per slice
- Coverage ratio
- Block ratio
- Handover ratio

7.2 Client-Specific Metrics

- Total connected time
- Total unconnected time
- Total request count
- Total consume time
- Total resource usage

7.3 Visualization

- Network topology map
- Time series charts for key metrics
- Summary statistics table

8. Usage Example

```
# Run simulation with configuration file
python -m slicesim config.yaml
```

9. Best Practices

1. **Configuration**: Start with a simple configuration and gradually increase complexity
2. **Warmup Period**: Use warmup ratio to exclude initialization effects
3. **Simulation Duration**: Ensure sufficient duration to observe steady-state behavior
4. **Client Density**: Balance client count with coverage areas
5. **Slice Parameters**: Configure slice parameters to reflect real-world requirements

10. Technical Limitations

1. No consideration of interference between base stations
2. Simplified mobility models
3. No detailed PHY layer simulation
4. Focus on resource allocation rather than packet-level simulation
5. KD-Tree limitations for client-base station assignment in edge cases

11. Future Enhancements

1. Add interference modeling between base stations
2. Implement more sophisticated handover algorithms
3. Add adaptive slice resource allocation
4. Support for more complex QoS requirements
5. Integration with real-world network data

Appendix A: Class Reference

Class	Primary Responsibility
BaseStation	Manages physical network infrastructure
Client	Represents end users with mobility and usage requirements
Coverage	Defines geographical coverage area
Distributor	Generates random values for various parameters
Graph	Visualizes simulation state and metrics
Slice	Manages logical partition of network resources
Stats	Collects and processes performance metrics
KDTree	Efficient spatial indexing for base station assignment