**RoverX: Autonomous Rescue Rover System**

**Technical Documentation**

**Executive Summary**

**RoverX is an advanced autonomous rescue rover system designed for disaster response operations. The system combines intelligent navigation, multi-sensor survivor detection, and sophisticated power management to operate effectively in disaster zones. This document provides a comprehensive overview of the system's features and capabilities.**

**1. Core System Architecture**

**1.1 Navigation System**

**The navigation system implements sophisticated path planning and movement control:**

**Zigzag Pattern Movement**

* **Implements efficient area coverage algorithms**
* **Uses natural, non-linear movement patterns**
* **Adjusts pattern based on battery level and terrain**

**Obstacle Avoidance**

* **Real-time ultrasonic sensor processing**
* **Dynamic path recalculation**
* **Safe distance maintenance**

**Path Planning**

* **Bezier curve implementation for smooth movement**
* **Battery-aware path optimization**
* **Natural direction changes**

**1.2 Survivor Detection System**

**The multi-sensor fusion system provides reliable survivor detection:**

**Sensor Integration**

* **Ultrasonic sensors for distance measurement**
* **Infrared sensors for heat detection**
* **RFID readers for tag detection**
* **Accelerometer for movement analysis**

**Data Processing**

* **Real-time sensor fusion**
* **Confidence-based detection**
* **Priority-based rescue planning**

**Tracking System**

* **Historical detection storage**
* **Movement pattern analysis**
* **Survivor location tracking**

**1.3 Power Management System**

**Intelligent power management ensures optimal operation:**

**Battery Management**

* **Dynamic power consumption optimization**
* **Battery life estimation**
* **Automatic charging station navigation**

**Power States**

* **Normal operation mode**
* **Low power mode**
* **Critical power mode**
* **Recharging mode**

**Consumption Optimization**

* **Movement-based power adjustment**
* **Sensor activity management**
* **Communication optimization**

**2. Safety and Reliability**

**2.1 Safety Features**

* **Obstacle detection and avoidance**
* **Battery level monitoring**
* **Temperature monitoring**
* **Communication loss handling**
* **Emergency stop functionality**
* **Safe distance maintenance**

**2.2 State Management**

**The system implements a comprehensive state machine:**

* **Idle state**
* **Searching state**
* **Moving to survivor**
* **Delivering aid**
* **Returning to charge**
* **Recharging state**

**3. Simulation Environment**

**3.1 Webots Integration**

* **Realistic 3D simulation environment**
* **Physics-based movement simulation**
* **Custom disaster zone scenarios**
* **Real-time visualization tools**

**3.2 Sensor Simulation**

* **Ultrasonic sensor modeling**
* **Infrared sensor simulation**
* **RFID reader emulation**
* **Accelerometer physics**

**3.3 World Design**

* **Custom disaster zone environment**
* **Multiple terrain types**
* **Dynamic obstacle placement**
* **Survivor placement tools**
* **Charging station locations**

**4. Technical Specifications**

**4.1 Navigation Parameters**

* **Obstacle threshold: 50cm**
* **Safe distance: 100cm**
* **Search pattern width: 200cm**
* **Search pattern length: 300cm**
* **Turn interval: 5 seconds**

**4.2 Power Management Parameters**

* **Recharge start: 5% battery**
* **Recharge stop: 80% battery**
* **Communication loss: 10% battery**
* **Low power mode: 20% battery**
* **Temperature warning: 45°C**
* **Temperature critical: 60°C**

**4.3 Sensor Thresholds**

* **Ultrasonic: 200cm**
* **IR reflection: 0.7**
* **RFID signal: 0.5**
* **Accelerometer: 2.0 m/s²**

**5. System Integration**

**5.1 Web Interface**

* **Real-time dashboard**
* **Sensor data visualization**
* **Control interface**
* **Status monitoring**

**5.2 API Integration**

* **Base URL: https://roverdata2-production.up.railway.app**
* **Session management**
* **Real-time data transmission**

**6. Project Structure**

**RoverX/**

**├── app.py # Main application entry point**

**├── navigation\_system.py # Navigation and path planning**

**├── sensor\_fusion.py # Multi-sensor data processing**

**├── power\_management.py # Power and battery management**

**├── rover\_controller.py # Main rover control system**

**├── static/ # Static web assets**

**├── templates/ # Web templates**

**├── worlds/ # Webots world files**

**├── controllers/ # Webots controllers**

**└── protos/ # Webots PROTO files**

**7. Dependencies**

* **Flask 2.0.1**
* **Flask-SocketIO 5.1.1**
* **Python-dotenv 0.19.0**
* **Requests 2.26.0**
* **Colorama 0.4.4**
* **Gunicorn 20.1.0**
* **Webots 2023b**

**8. Future Enhancements**

* **Advanced path planning algorithms**
* **Enhanced sensor fusion techniques**
* **Improved power optimization**
* **Additional safety features**
* **Extended simulation capabilities**

**9. Conclusion**

**RoverX represents a significant advancement in autonomous rescue rover technology, combining sophisticated navigation, reliable survivor detection, and intelligent power management. The system's modular architecture and comprehensive simulation environment make it an ideal platform for disaster response operations.**