AI Load Balancer

Live Mozilla

Real-Time AI Network Load Balancing Dashboard

Enter city or place name (e.g., Lucknow | Latitude (optional) | Longitude (optional) | Fetch Towers

Tip: enter a city name OR latitude+longitude. If both, coordinates are used.

Baseline | Live Simulation | Run | Demo Mode | Compare | Analytics | Download PDF

Performance Metrics

Algorithm: Baseline **Overloaded BS:** 0 **Jain Index:** 0 **Throughput:** 0 **Utilization:** 0

Technical Guide – How It Works

This project demonstrates how **AI** and **Machine Learning** can make **mobile networks smarter** by automatically redistributing users between cell towers for optimal performance.

- **Baseline (Nearest BS):** Users connect to the closest Base Station (BS) only.
- **Heuristic Rebalance:** Uses simple load-sharing logic — moves some users from overloaded towers to nearby free ones.
- **K-Means ML Reassignment:** Uses clustering (K-Means) to find the most balanced user distribution across towers.

The dashboard shows live metrics like **Jain Index** (fairness), **Throughput** (total users served), and **Utilization** (efficiency of towers).

> 📘 **Tip:** You can run each algorithm from the dropdown above and compare how AI-driven balancing improves overall network performance in the **Compare** and **Analytics** sections.

🎥 Quick Demo Mode – Live Simulation Explained

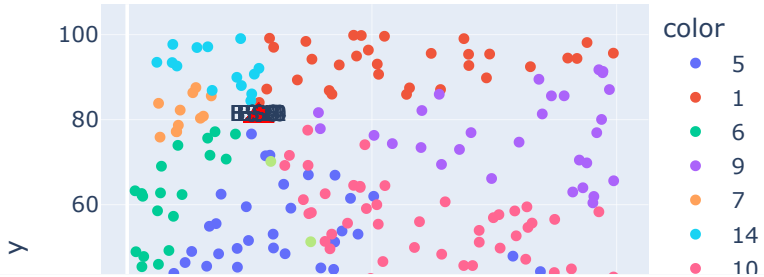
The **Demo Mode** is designed to simulate **real-world mobile traffic** in a short time. When activated, it automatically cycles through all algorithms — showing how network load changes as AI redistributes users between towers.

- ▶️ **Start Demo:** Click the yellow 🎬 **Demo Mode** button.
- 🔁 The dashboard will auto-run **Baseline** → **Heuristic** → **K-Means** sequentially.
- 📡 Charts and metrics update live — reflecting each algorithm's decisions.
- 📈 The **Live Prediction Feed** (bottom of the page) keeps showing real-time forecasts of throughput.

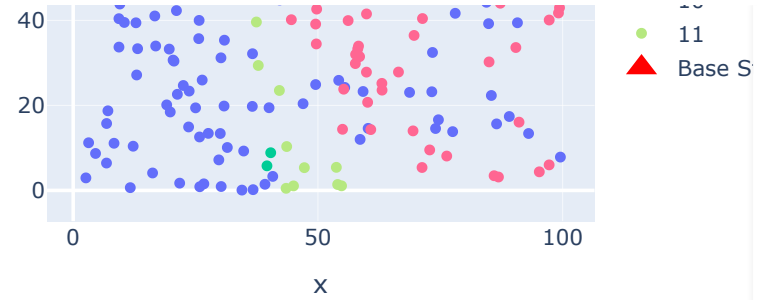
This is perfect for **presentations** — you can demonstrate how AI agents learn and balance tower loads without manual intervention.

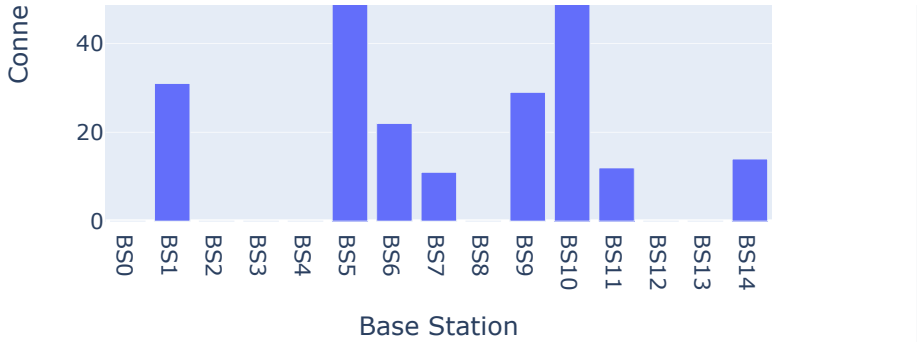
> 💡 **Pro Tip:** During Demo Mode, watch how **Heuristic** and **K-Means** gradually improve the **Jain Index** and **Throughput** compared to the baseline. It's AI model working in action!

Heuristic Rebalanced

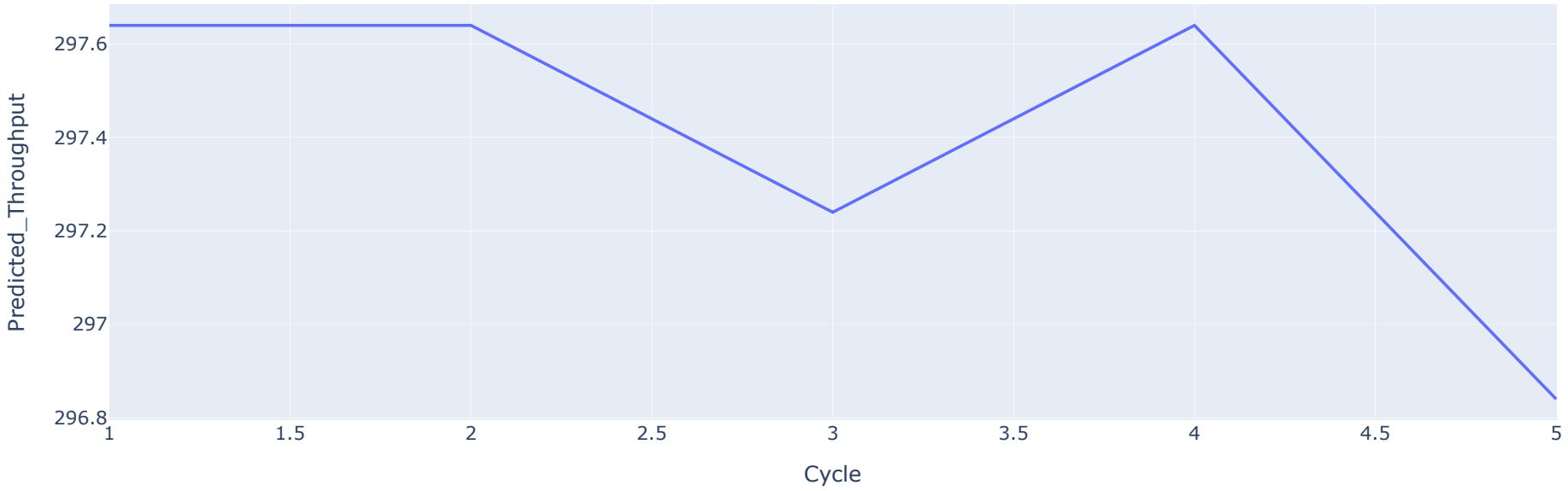


Base Station Loads





Forecasted Throughput (Next 5 Cycles)

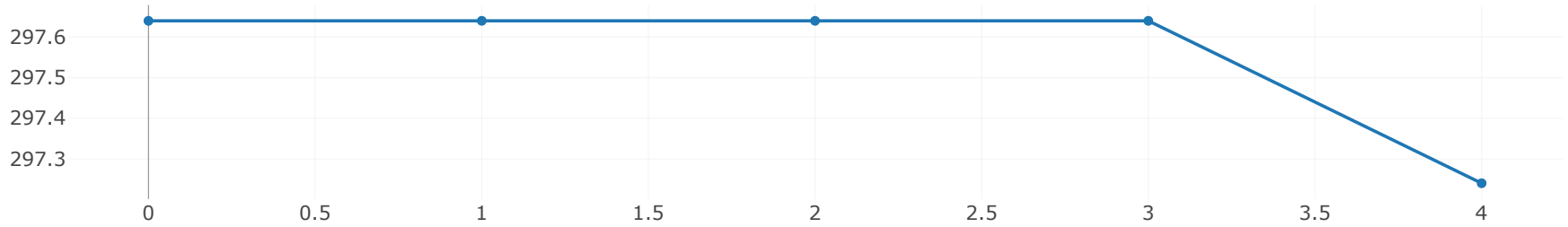


Live Prediction Feed

📶 AI Load Balancer Dashboard

🕒 05:39:56 → Forecasted Throughput: 297.24 users

🕒 05:39:46 → Forecasted Throughput: 297.64 users



📘 **Quick Guide:** Jain Index → Fairness (0–1). Throughput → Total users served. Utilization → Tower usage.