

→ Modeled annual average PM_{2.5} concentration (2018) µg/m³

For urban Panjim, average PM_{2.5} concentration was 28.3 ± 10.7 µg/m³. This is within the national standard (40) but nearly three times the WHO guideline (10).

→ Air monitoring infrastructure



MANUAL STATIONS



CONTINUOUS STATIONS

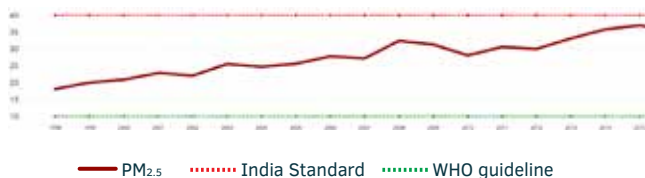


REQUIRED STATIONS

→ Annual averages from the national ambient monitoring program (2011–2015) µg/m³

PM ₁₀	SO ₂	NO ₂
62.4 ± 38.2	5.2 ± 4.0	13.4 ± 7.6

→ Trend in PM_{2.5} concentrations, based on satellite observations and global model simulations (1998–2016) µg/m³

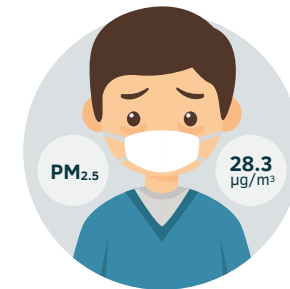


Designing an effective Air Quality Management (AQM) plan for a city requires robust data on levels of pollution, affected areas, source contributors, peaking trends and possible control mechanisms.

The Air Pollution Knowledge Assessment (APnA) City Program seeks to make this database available and also serve as a starting point for understanding air pollution.

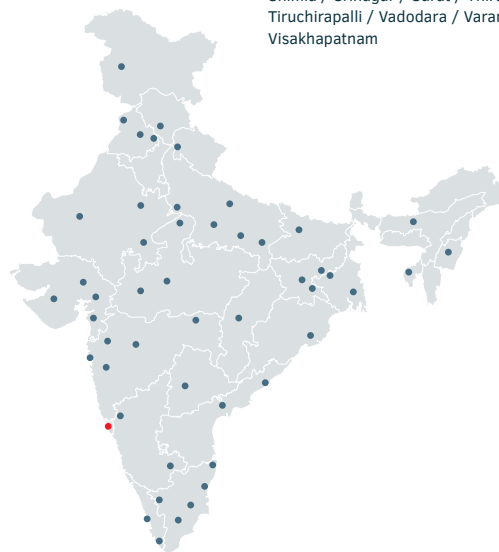
The program, implemented by Urban Emissions and facilitated by Shakti Sustainable Energy Foundation, seeks to create a comprehensive, city-specific information pool by pulling together data from disparate sources, surveys, mapping and atmospheric modeling.

Policy options based on this information, and their implementation, would be the effective next steps in improving the air quality of our cities.



THE AIR POLLUTION KNOWLEDGE ASSESSMENT (APnA) CITY PROGRAM

- Agartala / Agra / Ahmedabad / Allahabad / Amritsar / Asansol / Aurangabad / Bengaluru / Bhopal / Bhubaneswar / Chandigarh / Chennai / Coimbatore / Dehradun / Dhanbad / Dharwad-Hubli / Gaya / Guwahati-Dispur / Gwalior / Hyderabad / Imphal / Indore / Jaipur / Jamshedpur / Jodhpur / Kanpur / Kochi / Kolkata / Kota / Lucknow / Ludhiana / Madurai / Mumbai / Muzaffarpur / Nagpur / Nashik / Panjim / Patna / Puducherry / Pune / Raipur / Rajkot / Ranchi / Shimla / Srinagar / Surat / Thiruvananthapuram / Tiruchirapalli / Vadodara / Varanasi / Vijayawada / Visakhapatnam

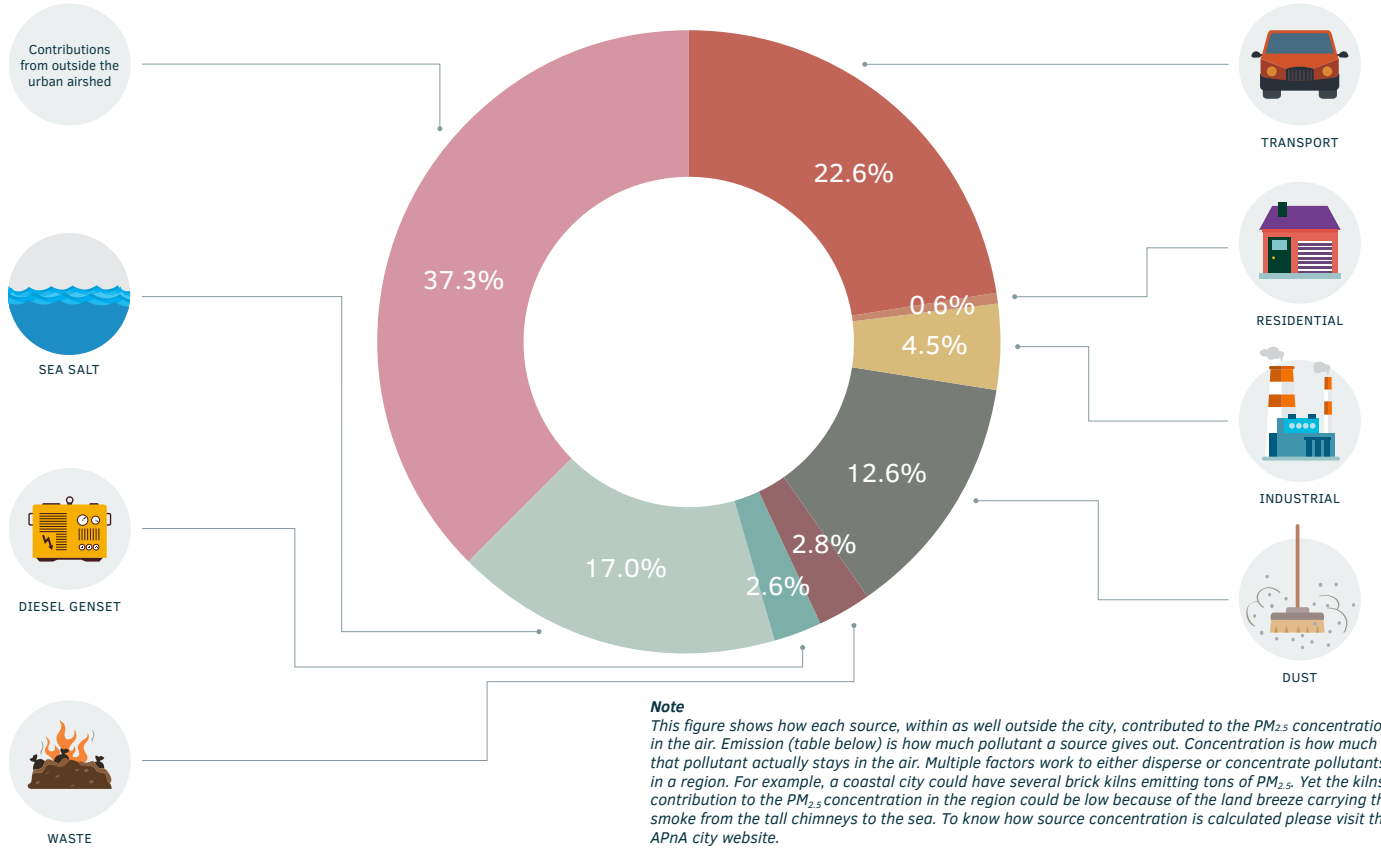


PANJIM

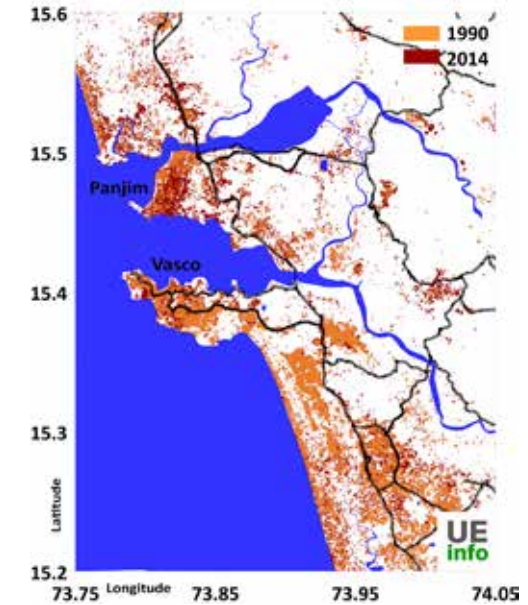
The city's PM_{2.5} concentration is nearly three times the WHO standards. Transport and seasalt are the primary contributors.

For detailed information on Panjim Air Quality, visit www.urbanemissions.info/india-apna

PM_{2.5} concentration : source-wise percentage share in 2018



Note
 This figure shows how each source, within as well outside the city, contributed to the PM_{2.5} concentration in the air. Emission (table below) is how much pollutant a source gives out. Concentration is how much of that pollutant actually stays in the air. Multiple factors work to either disperse or concentrate pollutants in a region. For example, a coastal city could have several brick kilns emitting tons of PM_{2.5}. Yet the kilns' contribution to the PM_{2.5} concentration in the region could be low because of the land breeze carrying the smoke from the tall chimneys to the sea. To know how source concentration is calculated please visit the APnA city website.



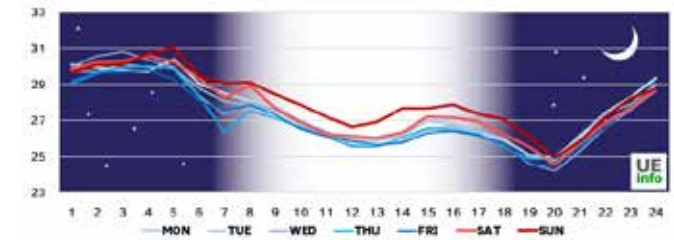
→ Global Human Settlements (GHS) built-up area

Urban areas in India are growing at a rapid rate. Using satellite observations derived Global Human Settlements (GHS) database, we can map the spatial footprint of a city over time. The map above shows the increase in built-up area between 1990 and 2014. An increase in built-up area usually means greater construction activity, intra-city transport, waste generation, and overall energy demand.

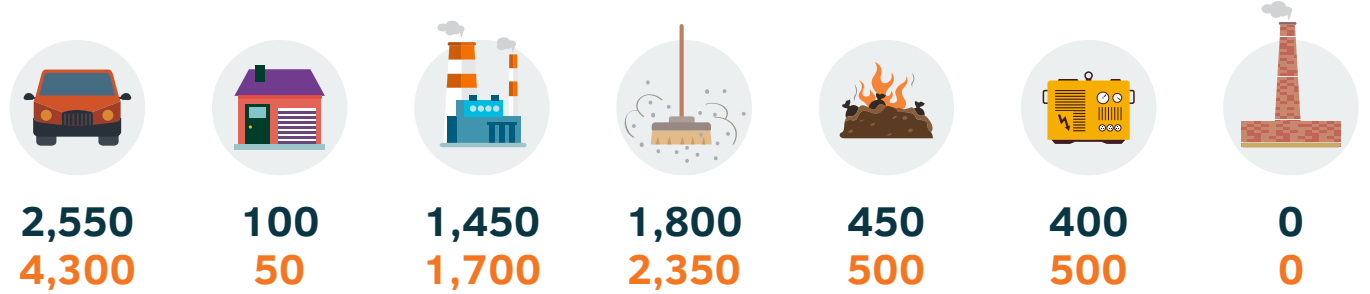
Urban land-use planning and provision of public transportation services is essential to address air pollution for cities in the future.

The graph below charts average speed of traffic by hour for everyday of the week within the city. As expected, speeds are greater at night and on Sundays and slows at peak times during the week. This is a summary of data extracted using Google Maps API services.

→ Hourly urban traffic speeds



PM_{2.5} emissions : source-wise share in tons in 2018 and 2030 (projected)



Total emissions in 2018 = 6,750 tons Total emissions in 2030 = 9,400 tons