

Resistivity analysis: Ecogas vs EPDT

M. Verzeroli



EP-DT
Detector Technologies



Analysis

Analysis

New functions that allows to perform the data analysis in a fast, easy way with the possibility to export them in a dedicated library

data_loading

It loads all the information and setup parameter, such as temperature, pressure...

```
def data_loading(path, rpc, mix_dict, client, run_number = 0):
    """
    function to load all the data:
    |
    INPUT:
    - path: path of the test beam run
    - rpc: rpc number for the selected run
    - mix_dict: dictionary to convert the gas composition into the plot label
    - client: client needed to download infos from InfluxDB
    - run_number: number of the run for fixed gas mixture

    OUTPUT:
    a dataframe with the following columns:

    - run_name: name of the hv scan
    - rpc: number of the rpc
    - voltage: voltage with no correction
    - voltage_app: voltage corrected
    - efficiency: detector's efficiency indata
    - currents_standard_mean_indata: currents indata
    - currents_beam_mean_indata: beam currents indata
    - currents_standard_std_indata: currents standard deviation indata
    - effmax: maximum efficiency
    - gamma: sigmoid fit parameter
    - hv50: sigmoid fit parameter
    - working_point: working point of the detector
    - up_abs: filter values as str, if source is off, up_abs = 'off'
    - mix: gas mixture composition
    - mix_label: gas mixture plot label
    - init_time: time of the beginning of the scan -> needed to download the environmental params
    - dew_point: humidity of the gas system
    - o2: oxygen of the gas system
    - p_bunker: pressure inside the bunker
    - p_outside: pressure outside the bunker
    - t_detector: temperature on the detector frame
    - t_bunker: temperature inside the bunker
    - t_outside: temperature outside the bunker
    - t_gas_room: temperature inside the gas room

    for the next step, this df need to be merged with all the scan for fixed source, gas mixture and rpc and different abs
    """
```

Analysis

currents_fit

```
def physics_fit(Vgap, a, V_T, sigma_T):  
    """  
    Function to fit all the hvscan plot.  
    """  
    term1 = (sigma_T / np.sqrt(2 * np.pi)) * np.exp(-((Vgap - V_T) ** 2) / (2 * sigma_T ** 2))  
    term2 = 0.5 * (Vgap - V_T)  
    erf_term = 0.5 * (1 + erf((Vgap - V_T) / (np.sqrt(2) * sigma_T)))  
    return a * (term1 + term2 * erf_term)
```

Fit function

It performs the current fit over the selected run, adding the fit parameter to the dataframe

resistivity_minimization_hv50

It consider the sigmoid fit for each dataset, and compute the minimization, as a function of R, of the following function over all the filter in the dataset wrt the source off scan considering HV50:

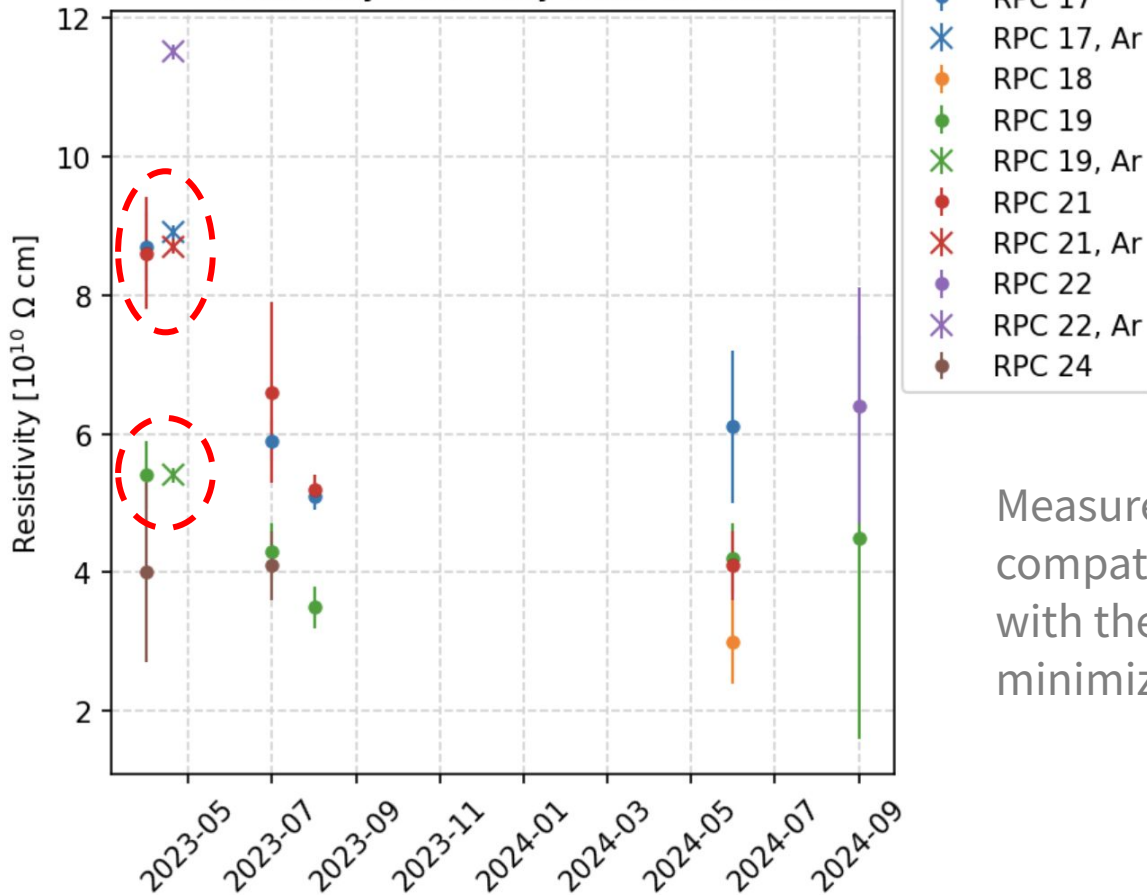
```
def objective(R):  
    return np.abs(np.sum(V_off - V_fil + R * I_fil))
```

The resistivity parameter is then added to the dataframe, as 'resistance_minimization_hv50'

Results

EPDT detectors

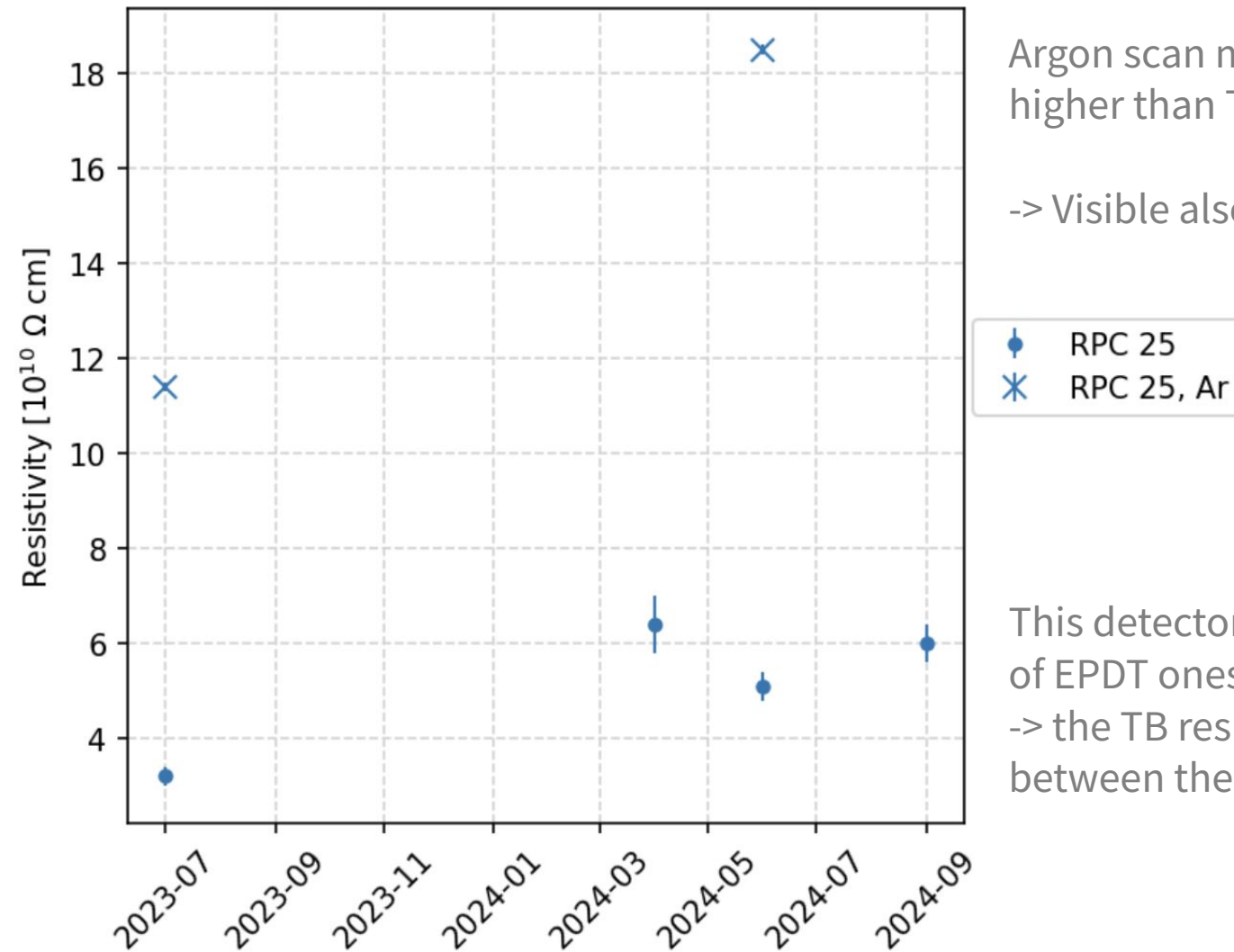
Summary resistivity from TB data



Measurements performed with Ar scan compatible with the results from the TB with the new analysis, using the minimization over all the abs method

ECOGAS detector

Summary resistivity from TB data



Argon scan measurements ~ 3 times higher than TB extrapolation

-> Visible also in Giuliana's analysis

This detector is from the same batch of EPDT ones

-> the TB resistivity is indeed similar between the two detectors

Proposal

- Perform GC analysis when the mixer is set to full Ar
 - > to exclude contamination from other gas that could increase the resistivity
- Still to be tested the old HFO bottle to search for impurities