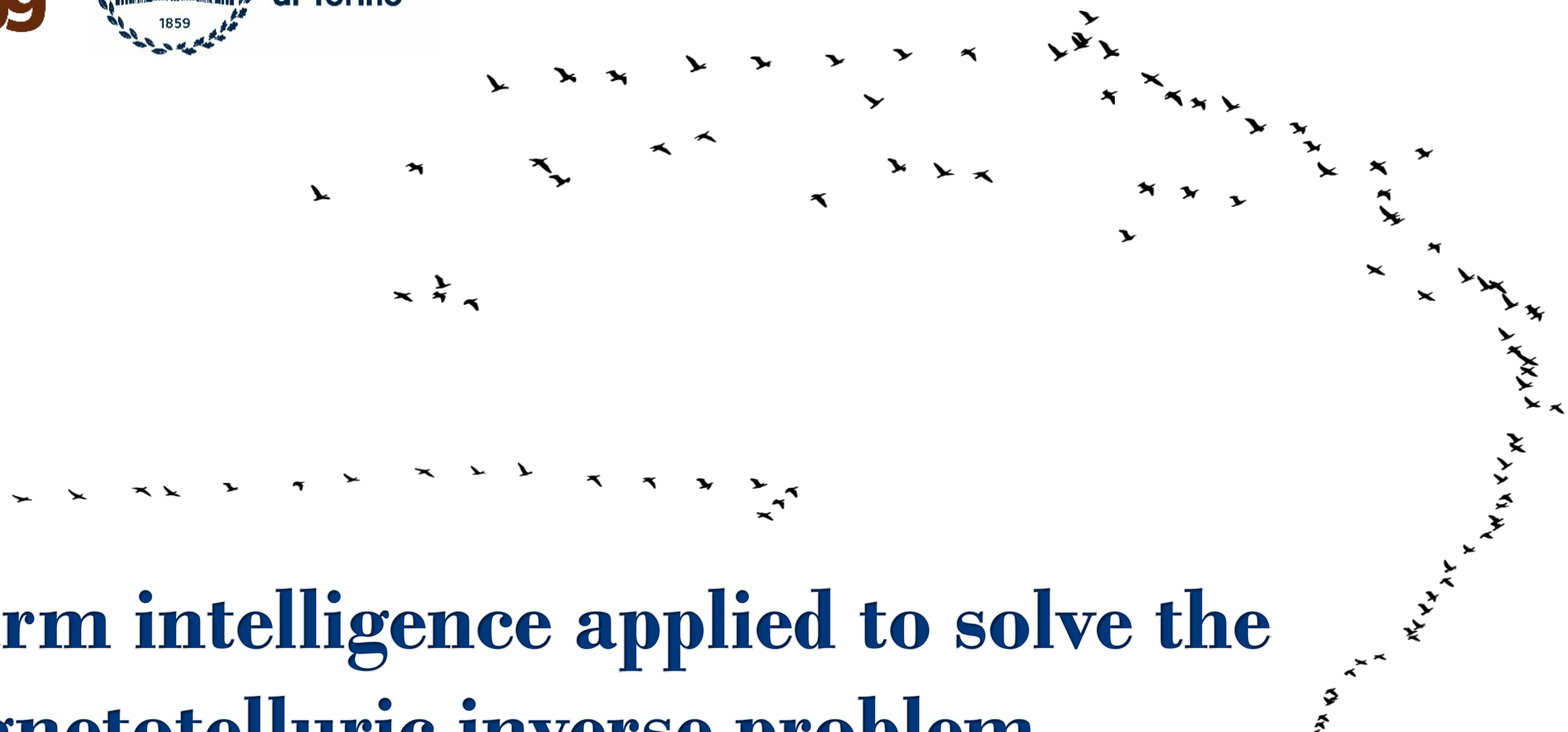


Swarm intelligence applied to solve the magnetotelluric inverse problem

Presenter: Alessandro Santilano
(CNR-IGG, Institute of Geosciences and Earth Resources)

Research group
Alberto Godio (Professor at Politecnico di Torino)
Francesca Pace (Research fellow at Politecnico di Torino)
Adele Manzella (Senior researcher at CNR-IGG)

Eminar series
16/06/2021

Acknowledgment

A special thank goes to the organization for this EMinar opportunity and to Alan Jones for the networking effort

Today I am the presenter on behalf of a great research group

Alberto Godio
(Professor at Politecnico di Torino)

Francesca Pace
(Research fellow at Politecnico di Torino)

Adele Manzella
(Senior researcher at CNR-IGG)

Me
(Researcher at CNR-IGG)

Outline

This EMinar is structured into two main parts

A **first educational part** is focused on the optimization and the swarm intelligence. The goal is to frame the pillar (and simple) concepts behind computational intelligence

Follows a **second research part** focused on the application of swarm intelligence to solve geophysical inverse problems. The goal is to share with the community the results of the research activities highlighting advantages, drawbacks and future perspective

Roadmap

Part 1: Optimization - an introduction

What **optimization** means?
Is the **optimization** a common process in real life?

Optimization is the act of achieving the **best** possible result under given circumstances *(Astolfi, 2018. The art of optimization)*

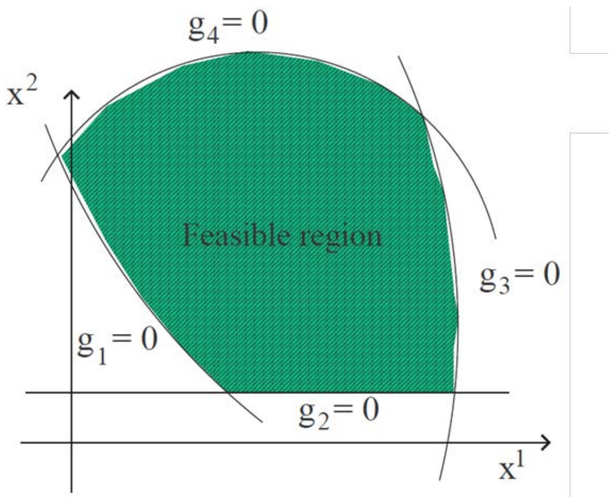
People optimize
Investors seek to create portfolios avoiding risks while achieving high returns.
Travellers minimize (at least attempt to) the travel time (or costs?)

Nature optimizes
Fermat’s Principle: rays of light follow paths that minimize their travel time
Evolution: highly specialized, complex structures often emerge when their most inefficient elements are selectively driven to extinction

Part 1: Optimization – math description

Optimization is the minimization or maximization of a function subject to constraints on its variables

Find
$\mathbf{x} = (x_1, x_2, ..., x_n)$, n=number of variables
which minimizes an objective function
$f(\mathbf{x})$
subject to the constraints
$g_j(\mathbf{x}) \leq 0$
for $j = 1, . . . , m$, and
$l_i(\mathbf{x}) = 0$
for $i = 1, . . . , p$



Example of feasible region in a 2D design space. Only inequality constraints are present (Astolfi, 2018)

Several problems cannot be analytically solved and a numerical approach is a way
One of the simplest case of optimization is: the knapsack problem…

Part 1: Optimization – an example

The knapsack problem: informal description

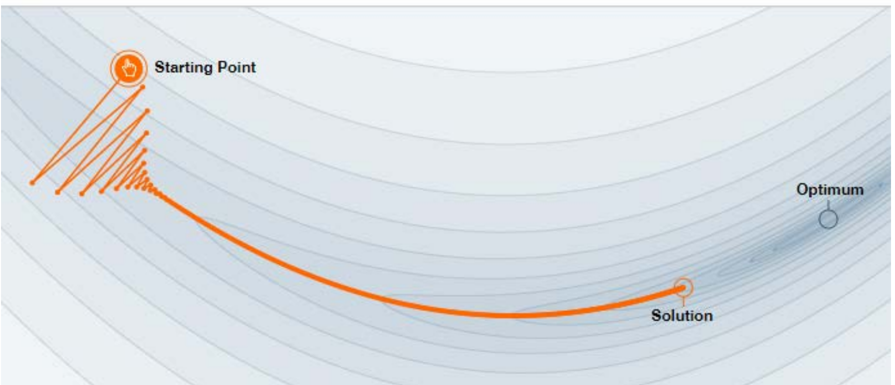
The burglar, who breaks into a house, faces the **(knapsack) problem:**
Determine the items to put in the knapsack, with a given constraint (knapsack strength), ensuring a total value as large as possible

Given 100 available items the possible solutions are 2^{100} (1.26765060022823e+30)
The problem is exponential
and the complexity must be reduced

The burglar can exploit some algorithm to select the **best** items to put in the knapsack and satisfying the constraint
What does "the best" mean? (This is a key question in optimization modelling)
Most valuable item? Lightest item? Ratio value/weight?

Part 1: Optimization - search algorithms

The solution of an optimization problem is a set of allowed values of the variables for which the objective function assumes an “optimal” value.
The approach implies to iteratively search for the solution **refining the initial guess(es)**.
We can take advantage of the following information: **the objective function value**

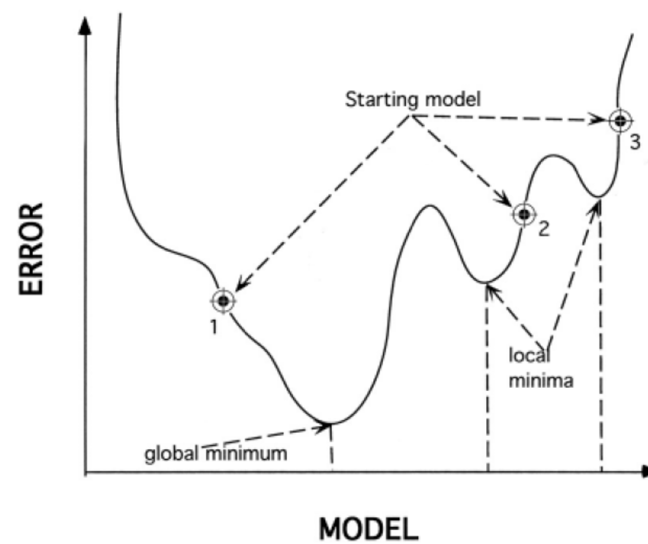


From http://fa.bianp.net/

The conventional approach is deterministic and derivative-based. The methods are the state of the art, efficient and stable. The solution is searched locally
An alternative approach exploits stochastic processes. **Global search metaheuristics** are aimed to reach the global minimum

The global minimum is reached when
$\mathbf{x}^* \in F$ and $f(\mathbf{x}^*) \leq f(\mathbf{x})$ for all $\mathbf{x} \in F$

F is the feasible set of variables



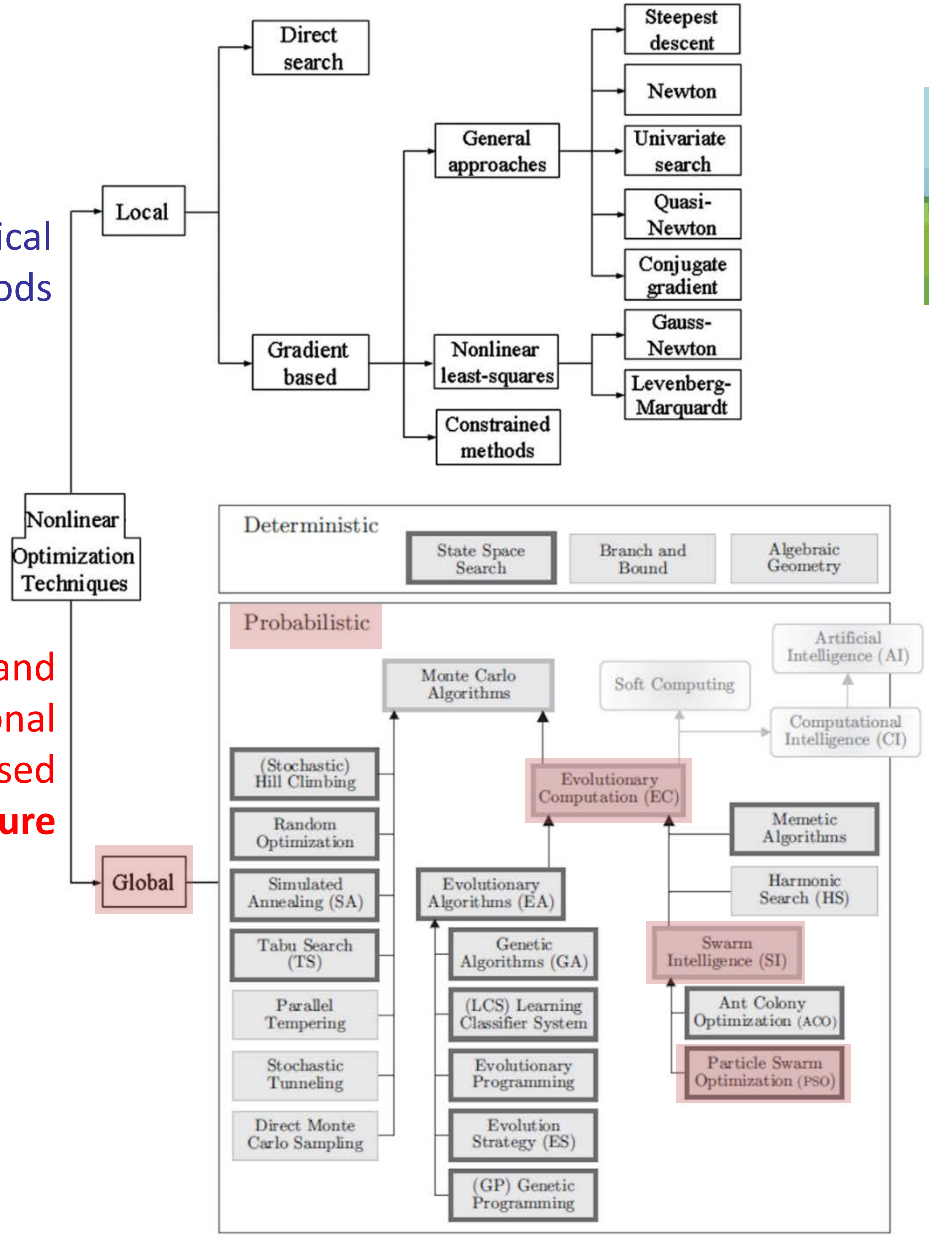
From Sen and Stoffa, 2013

Part 1: Search algorithms - Taxonomy

Classical methods

Stochastic and Computational Intelligence-based

Focus of the lecture



modified from Wang et al., 2008; Weise, 2009

Part 1: Search algorithms - classical

The conventional algorithms are **iterative** and exploit **deterministic** transition rules. A deterministic method produces always the same output if it is run on the same input.
An initial guess of the solution is iteratively refined. For non-linear problem, some methods implies a **linearization** (by differentiation)

As example, a common method is the Gauss-Newton
An early description appeared in Gauss' 1809 work to solve an astronomic problem: *Theoria motus corporum coelestium in sectionibus conicis solem ambientum*

The algorithm finds **x** that minimizes the vector of residuals (between measured and modelled data).
Starts with an initial guess for **x**, the function is linearized exploiting the Jacobian matrix and iteratively the least square solution is obtained

1st Hero of the day
Carl Friedrich Gauss
1777-1855
A genius who has contributed in several fields of science

Part 1: Computational intelligence

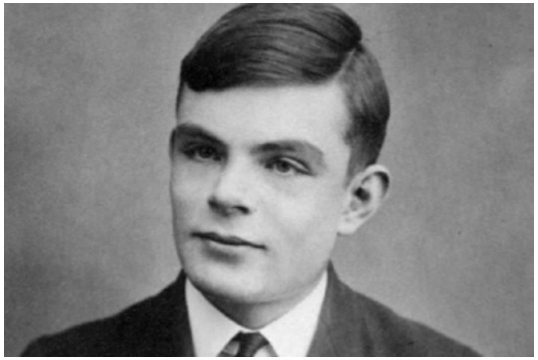
Metaheuristics are strategies that guide the search process

Computational intelligence-based **global search** paradigms differ from traditional search:

- **Use a population of points (agents) in their search.** Each point represents a solution to the problem, interact each other intelligently and explore larger model space domain
- **Learning strategies are used and stochastic transition rules are adopted**
- Use directly "fitness" information, instead of function derivatives

Can computers be intelligent?

This question redirect us to the concept of Artificial Intelligence.
In the mid-1900s, Alan Turing paved the way on this topic. Turing strongly believed that a well-designed computer could do anything that the brain does. His statements are still breakthrough.

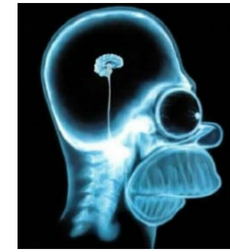


2° Hero of the day

Alan M. Turing (1912-1954)

A genius father of informatics and AI contributed to many fields of science and to the end of the WWII

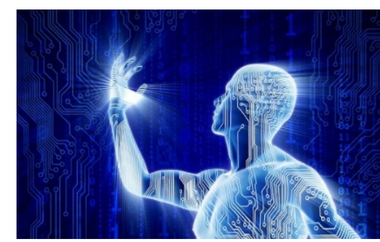
Part 1: Computational intelligence



Intelligence: Ability to perceive information, and retain it as knowledge to be applied towards adaptive behaviours (Wikipedia)

- **Computational intelligence**
  - **Nature-inspired metaheuristics**
    - **Population-based algorithms**
      - **Evolutionary algorithms (e.g. GAs)**
      - **Swarm intelligence (e.g. PSO)**

Focus of the EMinar



Computational Intelligence (CI) comprises algorithms that enable an intelligent behaviour in complex and changing environments. Ability to learn and/or to deal with new situations

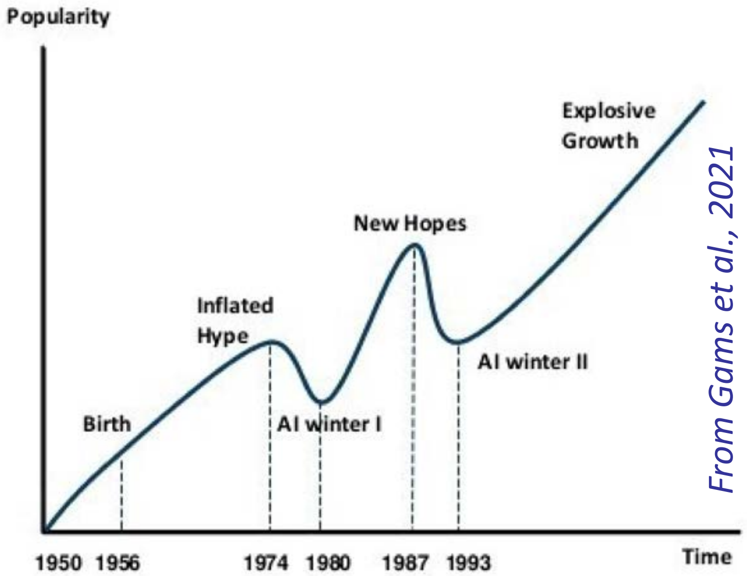
Swarm Intelligence (SI) emerges from the collective behavior of a large number of agents. SI-Algorithms are inspired by the behavior of groups of animals in nature

Part 1: Computational intelligence

Are computational intelligence methods recent? CI is known since decades even if most scientists were (sometimes still are) skeptical.
It is hard to frame its history (we risk to get stuck in a philosophical debate)

1950s: Alex Fraser used computers to **simulate natural genetic systems**
1970s: John Holland developed the famous “**genetic algorithm**”
1980s: **swarm intelligence** was simulated and metaheuristics arrived since the 1990s.
1984: the “Santa Fe” was founded as the first institute to study complex adaptive systems (Waldrop’s book *Complexity: The Emerging Science at the Edge of Order and Chaos)*

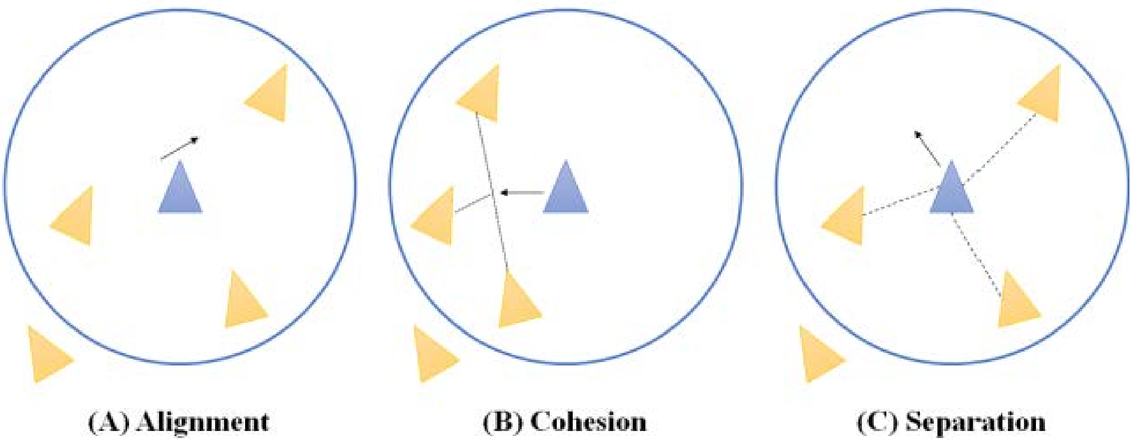
A main limit was the computational requirements
Recently, industry, science and governments focused the attention as computational power increased drastically in the last years



From Gams et al., 2021

Part 1: Swarm intelligence

One of the first example of swarm intelligence was provided by Craig Reynolds (*1987*) with **"Boids", an artificial life program**, which simulates the flocking behavior of birds.
The boids' flight (bird-oid objects) obeys to the three rules: **Alignment-Cohesion-Separation**



The program is able to artificially simulate real life animal social behavior. **In this case without any scope to solve optimization problems**

Video 1

The video is generated by a Matlab code available at: *https://github.com/b3rnoulli/boids-model*
The picture on rules is from Di Caro's lecture (CMU), available at https://web2.qatar.cmu.edu/~gdicaro/15382/

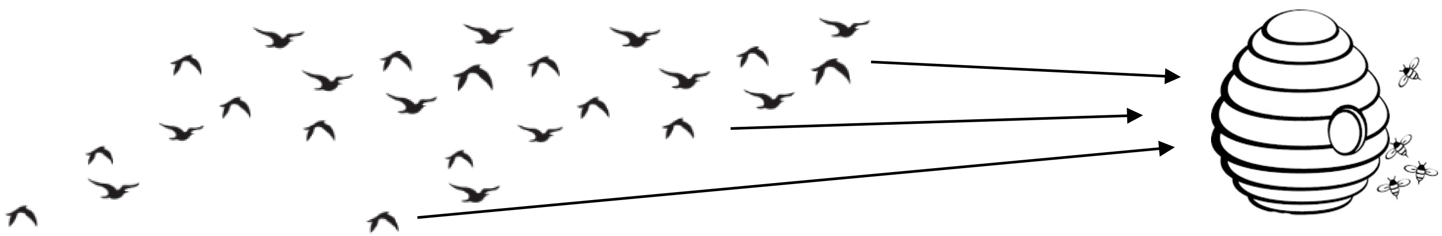
Part 1: Swarm intelligence

How Swarm intelligence can be applied to solve problems?

Let's think about a boids-like simulation with an additional attraction point (food or roost)

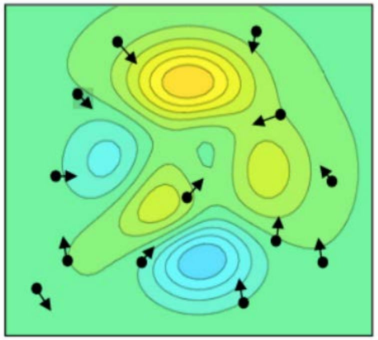
Each agent:

- is attracted to the location of the food
- remembers its closest position to the food
- shares information about its closest location to the food



What about if?

- **Flight space=**optimization landscape (search space, values of x)
- **Food (or roost)**= extremum of a function (the **best** solution)
- **Distance to the roost** = quality of the current solution of each agent
- **How to assess the quality? By using the objective function**

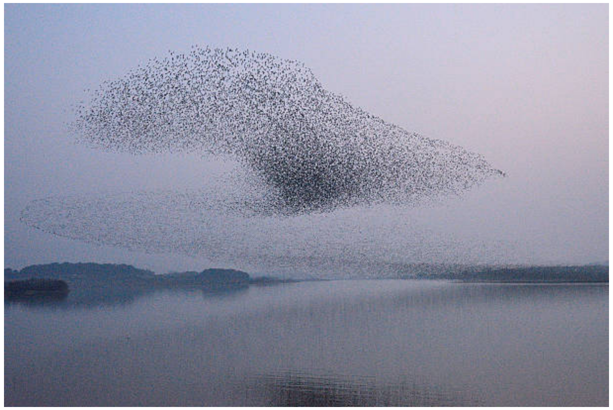


this slide is adapted from Di Caro's lecture (CMU) available at https://web2.qatar.cmu.edu/~gdicaro/15382/

The Particle Swarm Optimization PSO is a nature-inspired heuristic optimization method proposed by Kennedy and Eberhart (1995) and is based on two main concepts:

1) **simulation of the swarm intelligence and the social behavior observed in animals that group together**
2) **evolutionary computation**

Features:
Multi-agent: population based
Interaction: information is shared between agents
Emergence: interaction lead to the emergence of a "super-organism". The whole is more than the sum of its parts



PSO looks for the global minimum solution of a problem by mimicking the social behavior of flock of birds or school of fishes

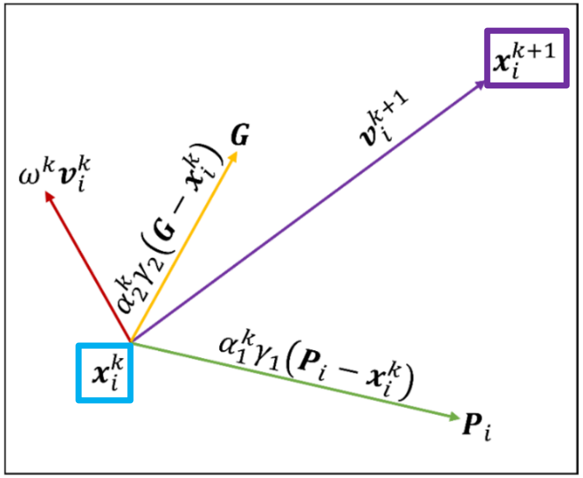
How it works to solve numerical optimization problems

- The swarm consists of N particles
- Each particle represents a feasible solution $\boldsymbol{x} \in X^n \subseteq R^n$ for the problem, sampling in a multidimensional search space (e.g. in MT is the set of electrical resistivity values)
- At the k^{th} iteration, the position of each particle in the search space is evaluated by the objective function $f(\boldsymbol{x})$ and represents the **fitness**
- Particles iteratively move and fly over the search space updating their position by using a displacement vector called **velocity**
- **The velocity vector of each particle is influenced by randomness, by its own experience and that of its neighbors (intelligent behavior)**
- Theoretically (and hopefully) the swarm will converge to optimal positions

For each particle $\mathbf{x_i}$ the position $\boldsymbol{x}_i^k$ is updated at $\boldsymbol{x}_i^{k+1}$ by computing a velocity vector $\boldsymbol{v}_i^{k+1}$

$$\boldsymbol{v}_i^{k+1} = \omega^k \boldsymbol{v}_i^k + \alpha_1^k \gamma_1 (\boldsymbol{P}_i - \boldsymbol{x}_i^k) + \alpha_2^k \gamma_2 (\boldsymbol{G} - \boldsymbol{x}_i^k)$$

$$\boldsymbol{x}_i^{k+1} = \boldsymbol{x}_i^k + \boldsymbol{v}_i^{k+1}$$



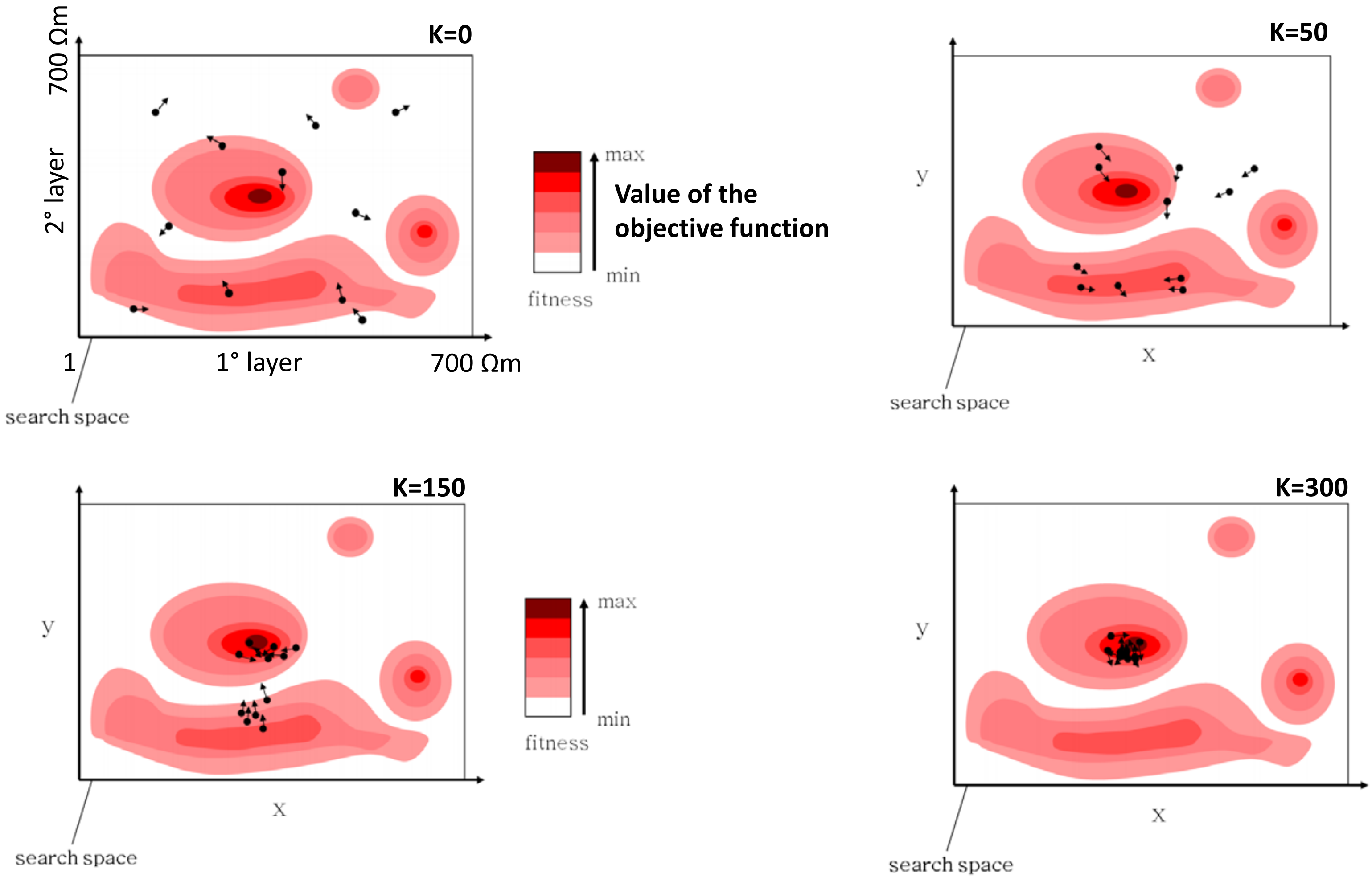
The adaptive behavior is a balance among:

- ✓ **Cognitive acceleration** α_1^k: towards personal best Pi
- ✓ **Social acceleration** α_2^k: towards global best G
- ✓ **Inertia weight** ω^k: momentum of the particle

Exploration → Exploitation → Convergence

Video 2

Ideal convergence of the swarm



this slide is adapted from Di Caro's lecture (CMU) available at https://web2.qatar.cmu.edu/~gdicaro/15382/

Part 2: The geophysical inverse problem

The solution of the geophysical inverse problem provides a physical **earth model** *m*, from the observations measured on surface *d*

$$G(\boldsymbol{m}) = \boldsymbol{d}$$

The physic behind the method is understood and specified in the “**forward operator**”, **G**
Conversely, the forward problem is to find ***d*** given ***m***

The magnetotelluric (MT) inverse problem is non-linear and ill-posed.
The computational complexity of the problem drastically increases with its dimensionality. For this reason, the application of stochastic population-based algorithms is not conventional and represents a challenge.

Deterministic algorithms are by far conventional for the inversion of MT data. The community relies on the state-of-the-art algorithms for 1D, 2D and 3D inversion. We stress that the dimensionality of the MT data drove the dimension of the problem to solved

Part 2: PSO - Geophysical application

Why CI-based approach to solve the geophysical inverse problem?

The CI-based approaches attempt to:

- **face the dependence of the final solution from the starting guess**
- **look for the global minimum of the function**
- **exploit randomness to assess the uncertainty**

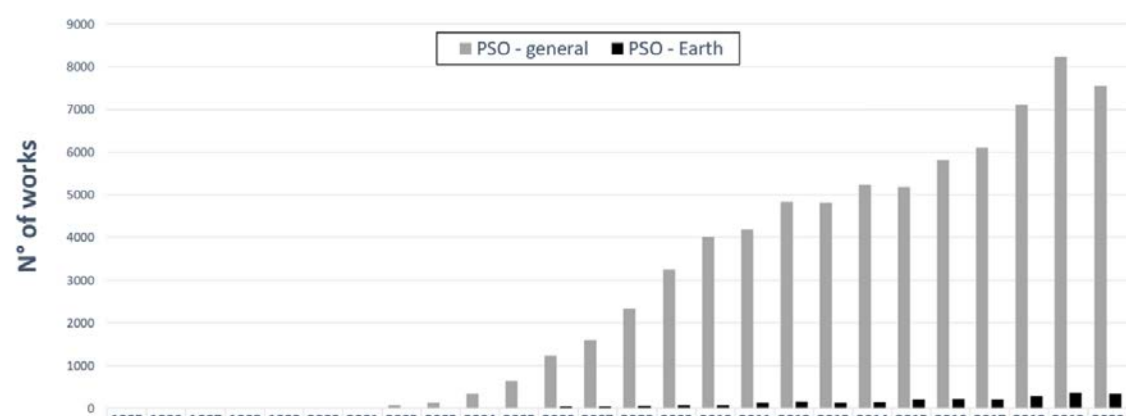
The aim of our research was to adopt and validate PSO to solve the MT inverse problem
Various research groups are working with PSO in geophysics but few studies were completely addressed to MT
We started with the 1D problem and continued with the solution of more complex 2D and joint problems, now implemented in the **"GlobalEM" matlab package**

Comprehensive works are required to lay the foundations for more complex problems such as the 3D (our next target)

Part 2: PSO - Geophysical application

The adoption of PSO is increasing in scientific works, recently, also in Earth Sciences

In Pace et al., 2021 a review of PSO application for geophysical modelling is available



	1995	1996	1997	1998	1999	2000	2001	2002	2003	2004	2005	2006	2007	2008	2009	2010	2011	2012	2013	2014	2015	2016	2017	2018	2019	2020
PSO - general	2	0	3	7	6	11	13	73	121	332	630	1223	1607	2348	3249	3995	4184	4831	4819	5226	5181	5800	6094	7111	8233	7547
PSO - Earth	0	0	0	0	0	0	0	1	0	2	12	34	37	47	70	75	121	153	127	142	207	222	201	280	363	331

Year

Surveys in Geophysics
https://doi.org/10.1007/s10712-021-09638-4

Check for updates

A Review of Geophysical Modeling Based on Particle Swarm Optimization

Francesca Pace[1] · Alessandro Santilano[2] · Alberto Godio[1]

Received: 14 December 2020 / Accepted: 27 February 2021

Abstract

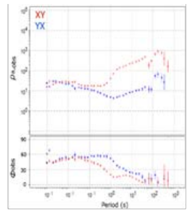
This paper reviews the application of the algorithm particle swarm optimization (PSO) to perform stochastic inverse modeling of geophysical data. The main features of PSO are summarized, and the most important contributions in several geophysical fields are analyzed. The aim is to indicate the fundamental steps of the evolution of PSO methodologies that have been adopted to model the Earth's subsurface and then to undertake a critical evaluation of their benefits and limitations. Original works have been selected from the existing geophysical literature to illustrate successful PSO applied to the interpretation of electromagnetic (magnetotelluric and time-domain) data, gravimetric and magnetic data, self-potential, direct current and seismic data. These case studies are critically described and compared. In addition, joint optimization of multiple geophysical data sets by means of multi-objective PSO is presented to highlight the advantage of using a single solver that deploys Pareto optimality to handle different data sets without conflicting solutions. Finally, we propose best practices for the implementation of a customized algorithm from scratch to perform stochastic inverse modeling of any kind of geophysical data sets for the benefit of PSO practitioners or inexperienced researchers.

Keywords Particle swarm optimization · Stochastic inverse modeling · Inversion · Swarm intelligence · Optimization · Joint optimization

Part 2: PSO – 1D MT

PSO for solving the magnetotelluric inverse problem: 1D

Resistivity Search domain



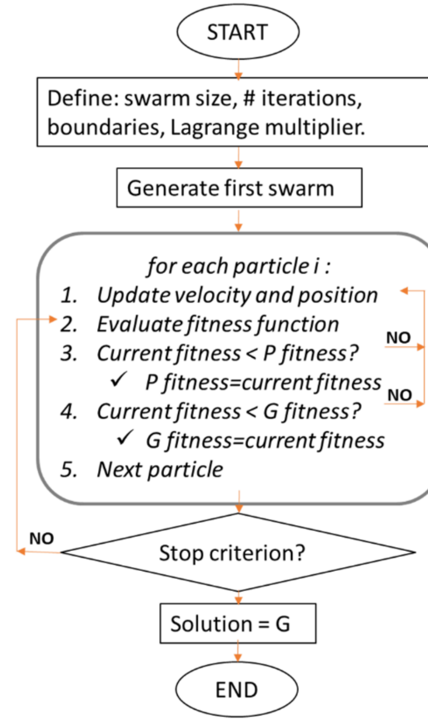
Resistivity search domain	Discretized earth model
*1-10000 Ω*m*	m_1
*1-10000 Ω*m*	
*1-10000 Ω*m*	
*1-10000 Ω*m*	*Discretized earth model*
*1-10000 Ω*m*	m_n

Fixed thickness of layers

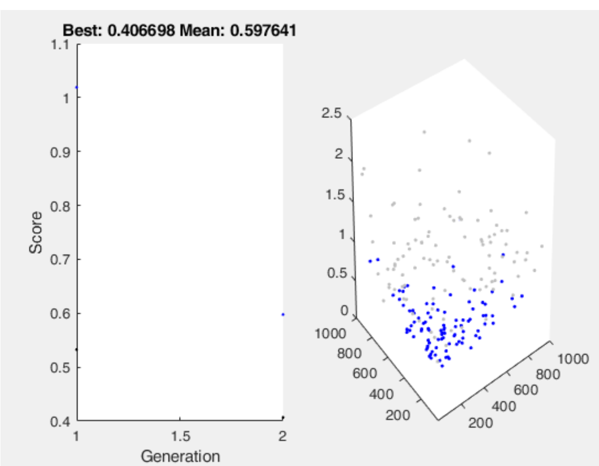
- The parameters to be optimized are the resistivities of layers
- Select the main settings of PSO (population, iterations, accelerations)
- Population size and iteration number are problem dependent
- The search domain (lower and upper boundaries) is defined for each parameter
- Each particle is a vector design ***m*** of model parameters and represents an earth model that is tested according to a minimization function

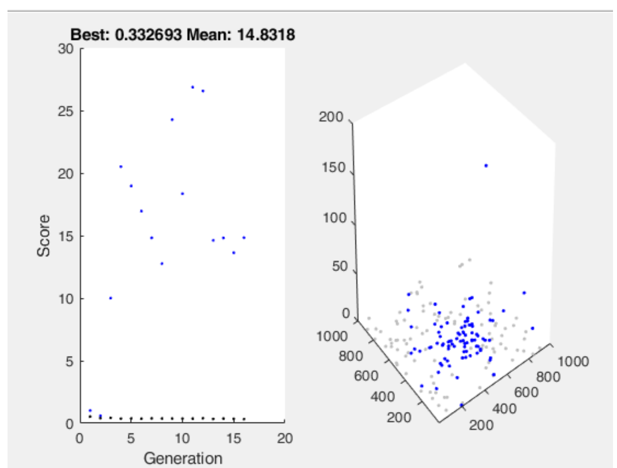
Part 2: PSO – 1D MT

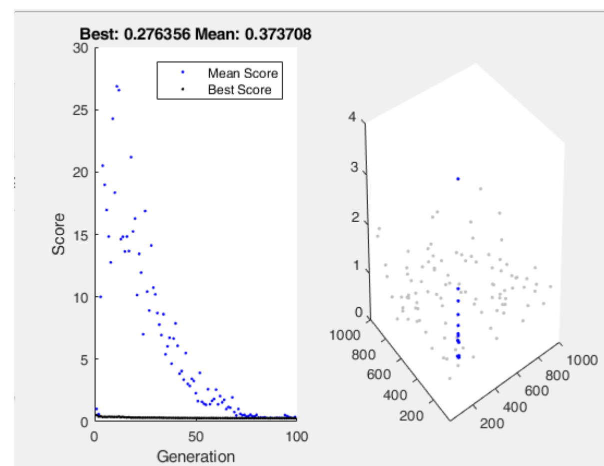
- The algorithm, each generation, updates the model parameters
- **The emerged model toward the swarm converged is the final resistivity model (or a set of model within a tolerance)**
- The convergence is affected by velocity's terms, the population size and iterations
- Several models are tested, i.e. several forward computations
- **More trials can be run, i.e. optimizations with same settings**



From Pace et al., 2019a





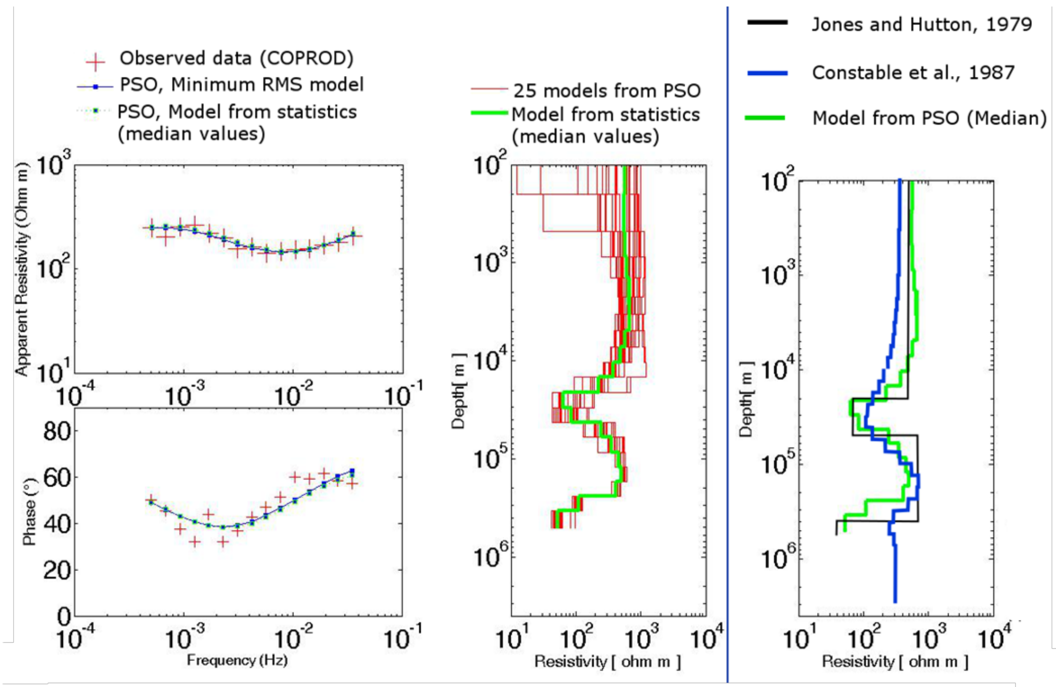


Part 2: PSO – 1D MT

Godio and Santilano (2018) proposed an early work on the adoption of PSO in MT.
An Occam-like CI-based optimization was implemented

Structure minimization sensu Constable et al., 1987

$$\Psi\,(\boldsymbol{m}) = \left(a\left\|\boldsymbol{\rho}_{a,o} - \boldsymbol{\rho}_{a,p}\right\|_2 + b\left\|\boldsymbol{\phi}_{a,o} - \boldsymbol{\phi}_{a,p}\right\|_2\right) + \lambda^2\,\|\partial m\|_2$$



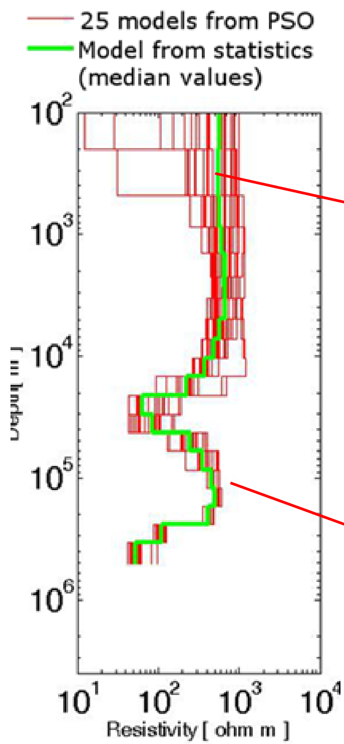
- **Module**: PSO 1D MT
- **Dataset**: Coprod (Jones and Hutton, 1979)
- **Unknowns**= 20 (layers)
- **Swarm size**= 300
- **Max iterations**= 200
- **Boundaries** = 1 – 5000 Ωm
- **Model initialization**: Random
- **Trials**: 25
- **Computation-time** = 6 min
- **Parallelization**= no

*Godio and Santilano (2018): **On the optimization of electromagnetic geophysical data: Application of the PSO algorithm**. Journal of Applied Geophysics 148 (2018) 163–174*

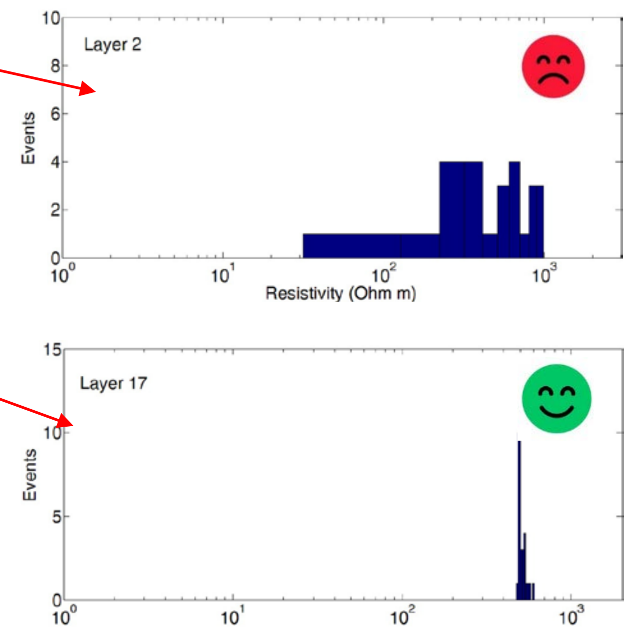
Part 2: PSO – 1D MT

The stochastic nature of the optimization can be exploited to retrieve information on the uncertainties of the results, by analyzing the a posteriori distribution of solutions
A best practice, if possible, is to run the optimization several times with the same settings and assess statistically the quality of the resulting models

For each parameter a unimodal distribution of the estimated resistivity can point out the validity of the solution.
Multimodal distributions indicate poorly resolved model. Simply statistic can evaluate uncertainty



A-posteriori distribution from the Coprod dataset



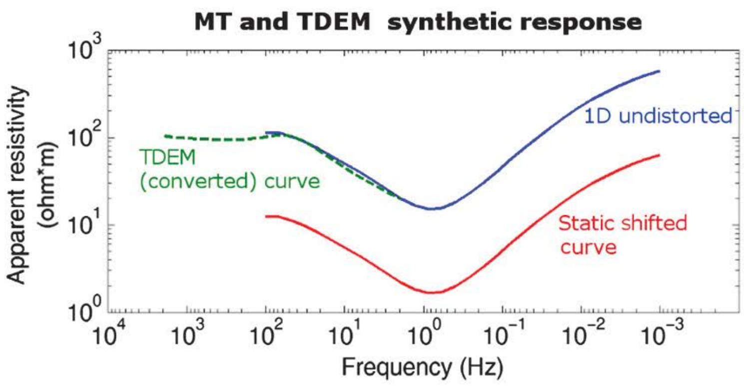
The a-posteriori distribution analysis includes several effects

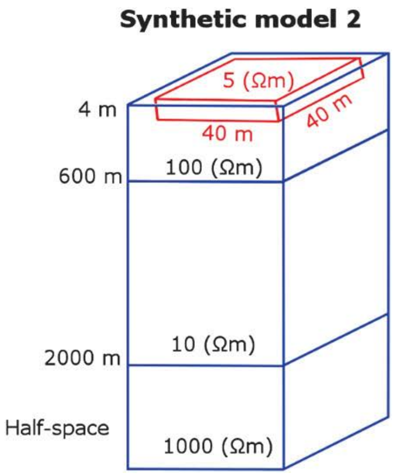
Part 2: PSO – Simultaneous 1D MT-TEM

A further scheme was developed for the simultaneous analysis of MT and transient EM (TEM) data based on swarm intelligence (Santilano et al., 2018)
The work addressed the galvanic distortion of MT data. The effect is a frequency-independent shift of the MT apparent resistivity curve for a constant multiplier

- This scheme is not purely joint and a single objective PSO is adopted
- TEM data are converted in MT data (Sternberg et al., 1988) based on the correlation between the time-domain diffusion depth and the frequency-domain skin depth:

$$t=194/f$$

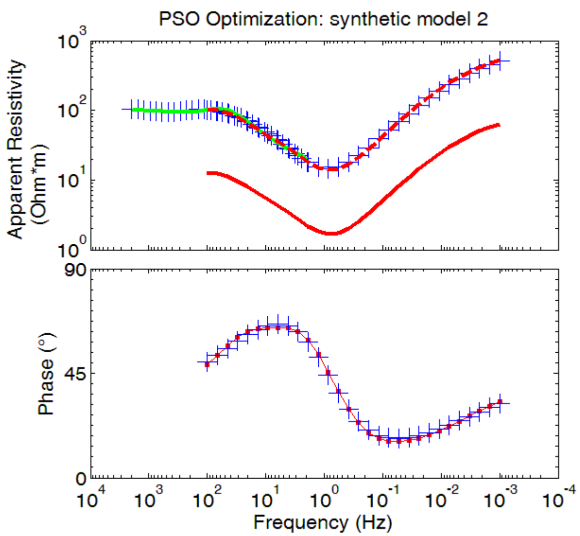


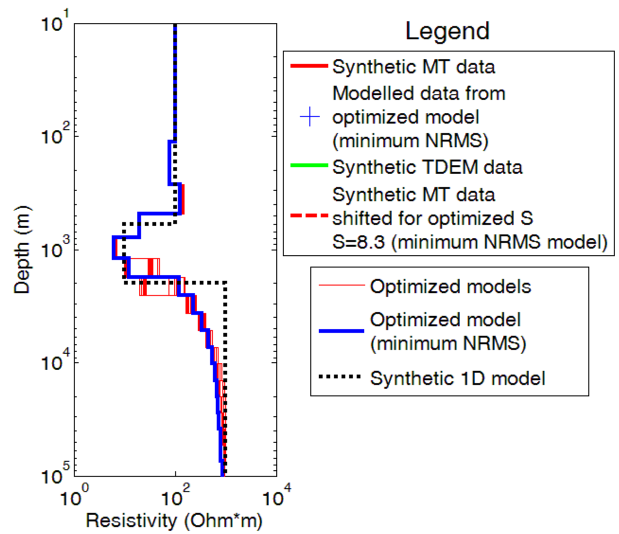


Part 2: PSO – Simultaneous 1D MT-TEM

In this work, the static shift parameter is included in the objective function as well as in the design vector (to be optimized) in addition to the resistivities.

$$\Psi(\mathbf{m}) = (a\|(\mathbf{S}\boldsymbol{\rho}_{\mathbf{a,o}}) - \boldsymbol{\rho}_{\mathbf{a,p}}\|_2 + b\|\boldsymbol{\phi}_{\mathbf{a,o}} - \boldsymbol{\phi}_{\mathbf{a,p}}\|_2 + c\|\boldsymbol{\rho}_{\mathbf{a(TDEM),o}} - \boldsymbol{\rho}_{\mathbf{a(TDEM),p}}\|_2) + \lambda\|\partial \mathrm{m}\|_2.$$





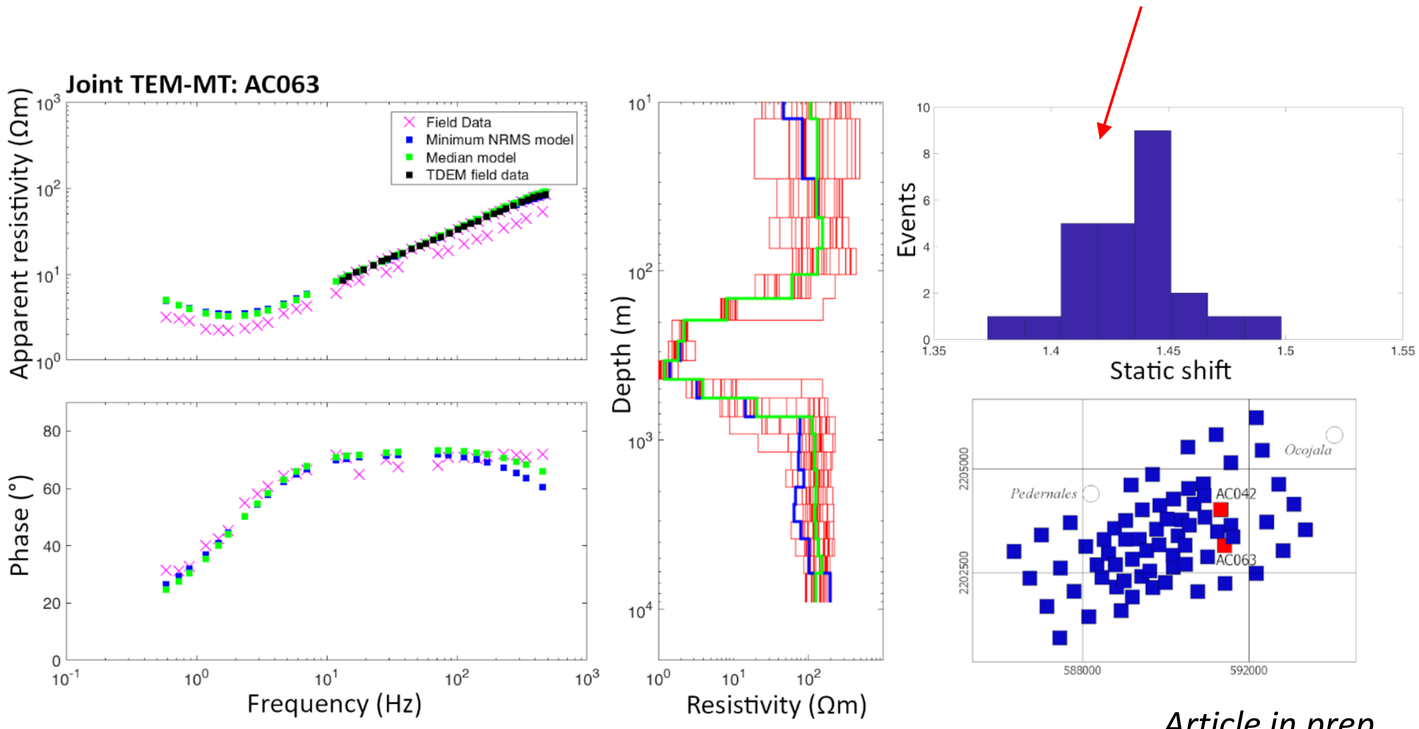
- **Module**: PSO MT-TEM
- **Dataset**: synthetic
- **Unknowns**= 19 +1(S)
- **Swarm size**= 300
- **Max iterations**= 200
- **Boundaries** = 1 – 2000 Ωm; 0.001-10 (S)
- **Model initialization**=Random
- **Trials**: 25

Santilano et al., 2018: ***Particle swarm optimization for simultaneous analysis of magnetotelluric and time-domain electromagnetic data****. Geophysics, 83 (3), E151–E159*

Part 2: PSO – Simultaneous 1D MT-TEM

In the frame of the GEMex project, we analysed the MT and TEM data acquired by our Mexican and Icelandic colleagues for the study of the Acoculco Caldera.
Results from an MT sounding with 1D dimension of Z are shown:

The stochastic optimization of the static shift provides insight on the **uncertainty**



Article in prep.

Part 2: PSO – 2D MT

The solution of the 2D MT problem by swarm intelligence faced main barriers: computational complexity and resource demanding

Let’s think about the number of forward solution to be computed by PSO:

*a*b*

a population size (problem dependent)

b number of iterations (problem dependent)

Time to solve 1D MT forward problem t_{1D}*=0.02 (s),* about *a=170* and *b=100*

The total forward simulations last about 5-6 mins

Time to solve 2D MT forward problem t_{2D}*=0.12 (s)*, about *a=5000* and b=3000

The total forward simulations last at least 1800000 mins

We spent much effort to optimize the code, to eliminate non essential computations and for parallelization

The code is usually run on HPCs and an optimization lasts few hours.

Let’s think accomplishing sensitivity analysis on settings parameters…again!

Part 2: PSO – 2D MT

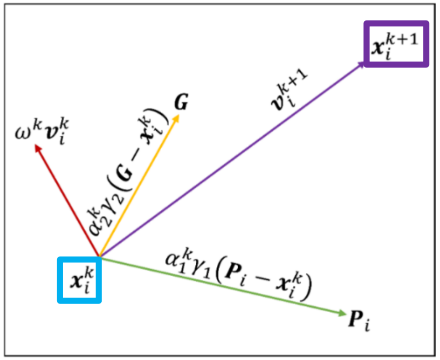
Objective function *f(m)* to be minimized for the 2D problem:

$$F(\boldsymbol{m}) = \left(\frac{1}{M}\left\|\frac{\boldsymbol{\rho}_{a,o}-\boldsymbol{\rho}_{a,p}}{\boldsymbol{\Delta\rho}_{a,o}}\right\|_2^2 + \frac{1}{M}\left\|\frac{\boldsymbol{\phi}_o-\boldsymbol{\phi}_p}{\boldsymbol{\Delta\phi}_o}\right\|_2^2\right)^{1/2} + \lambda_x \|\partial_x \boldsymbol{m}\|_2 + \lambda_z \|\partial_z \boldsymbol{m}\|_2$$

Data misfit — *Model smoothing*

In Pace et al., 2019 (Geophysics), we adopted a Time-Variant PSO (PSO)
The cognitive, social and inertia coefficients change with iterations

α_1^k	2 … 0.5
α_2^k	0.5 … 2
Φ^k	0.9 … 0.4



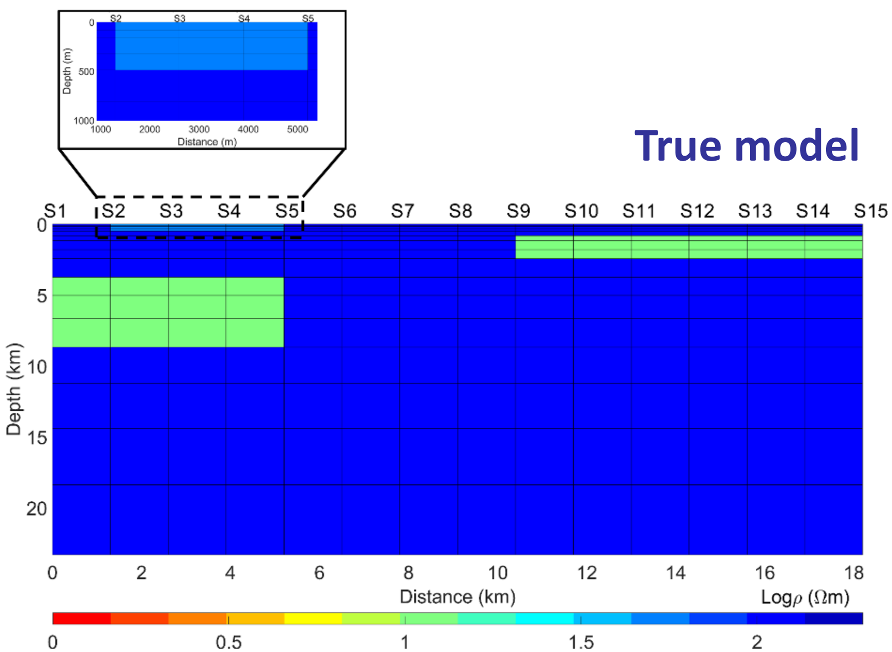
Input arguments:

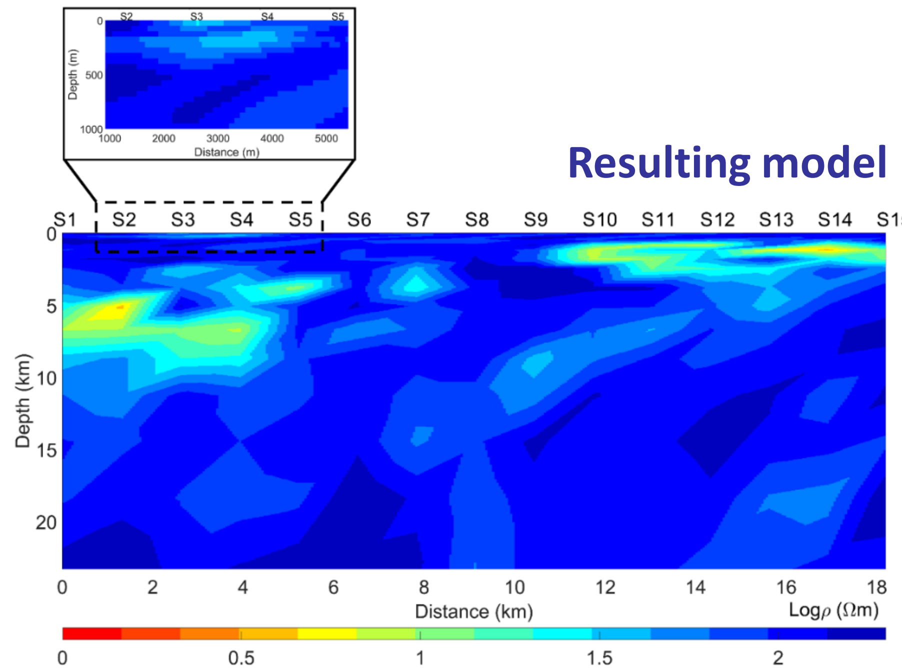
- **N° of iterations.** Problem dependent; ensure convergence
- **Stopping criteria.** Max iteration, same *f(m)* for n iteration, RMS
- **Acceleration coefficients α_1 and α_2.** Sensitivity analysis
- **Swarm size N.** (problem dependent) Sensitivity analysis
- **Initialization.** Sensitivity analysis
- **Boundaries of the search space.** Avoid unnecessary search (extremely important)

Part 2: PSO – 2D MT

Early attempts on synthetic data: early tests and sensitivity analyses

- 2D model: 15 MT sites, ~ 900 grid cells, 33 layers, 10% Gaussian noise
- Swarm size = 8600; Iteration = 1674; Lagrange multiplier: λx = λz = 0.1
- Runtime = 28 h; RMS= ~ 1; Trials=3





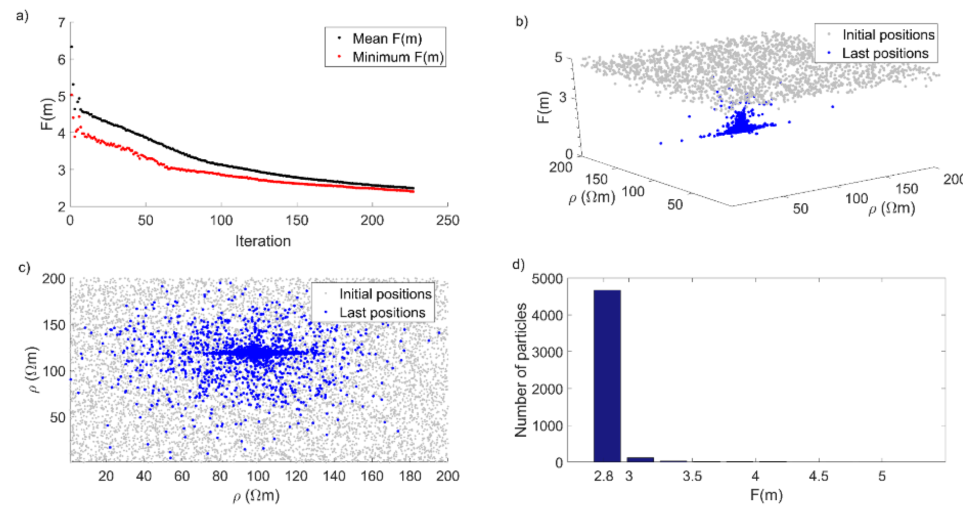
Pace et al., 2019a: ***Particle swarm optimization of 2D magnetotelluric data****. Geophysics, 84 (3) E125–E141*

Part 2: PSO – 2D MT

Optimization performance: Long runtime, few trials. Assess the rate of convergence

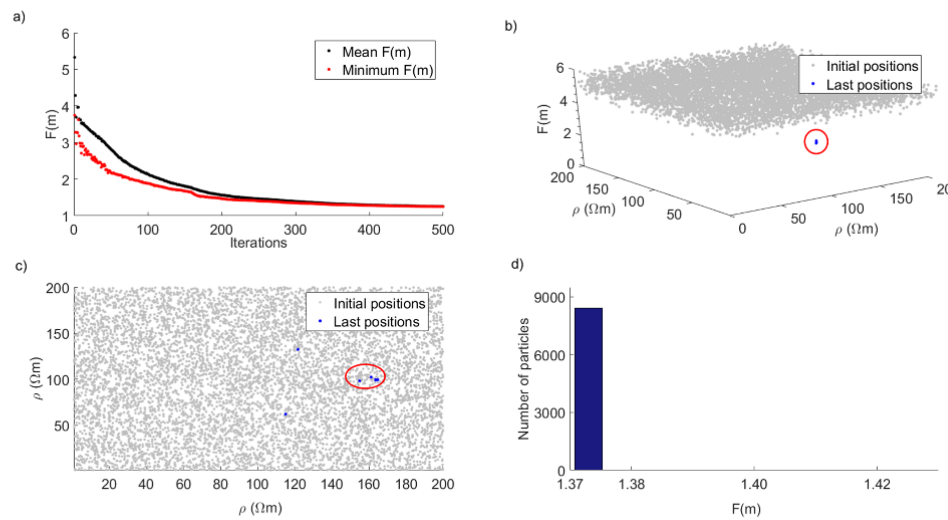
Worst case

- Incomplete minimization
- Bad convergence
- Scattered distribution of particles



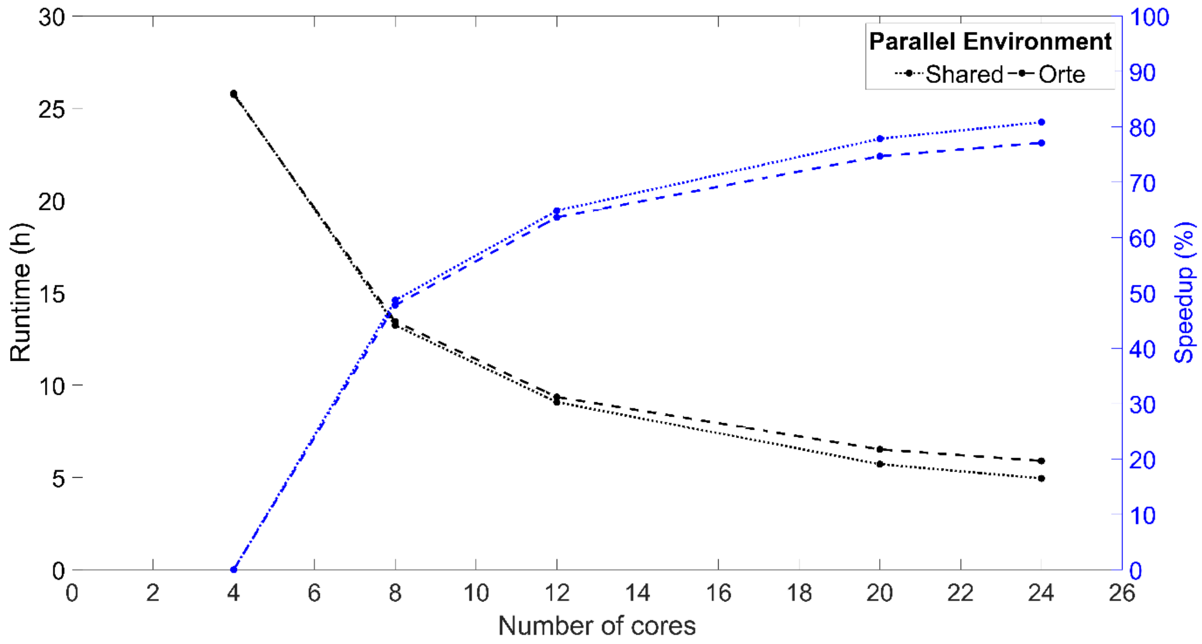
Best case

- Effective minimization
- Particles converge toward a unique position
- Unique peak in the histogram



Part 2: PSO – 2D MT

Runtime speedup
Test on HPC cluster for a reference PSO simulation:

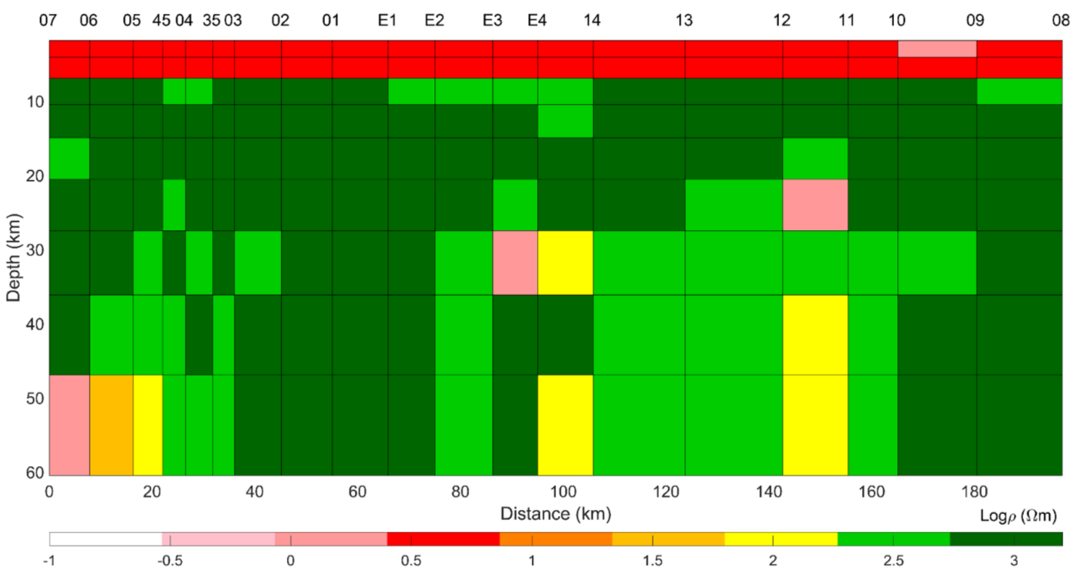


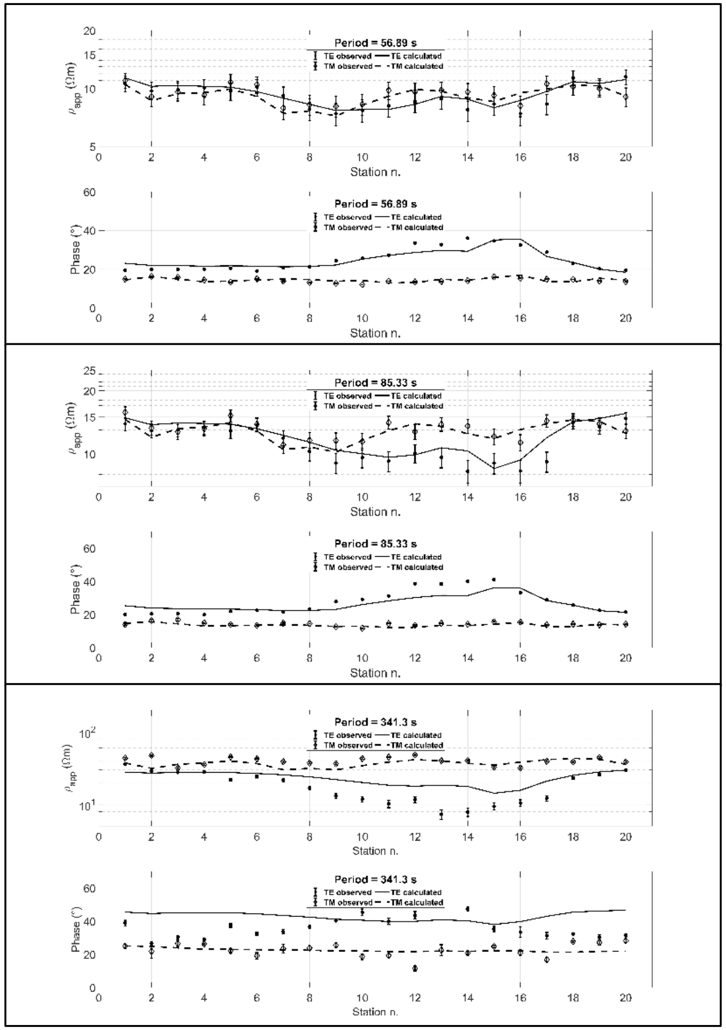
150 iterations
10000-particles
24 cores

5 hours instead of 26
runtime saving up to 80%
parallel environment “shared” faster than “orte”

Part 2: PSO – 2D MT

The PSO2DMT module was firstly tested on the COPROD2 dataset (Pace et al., 2019a)





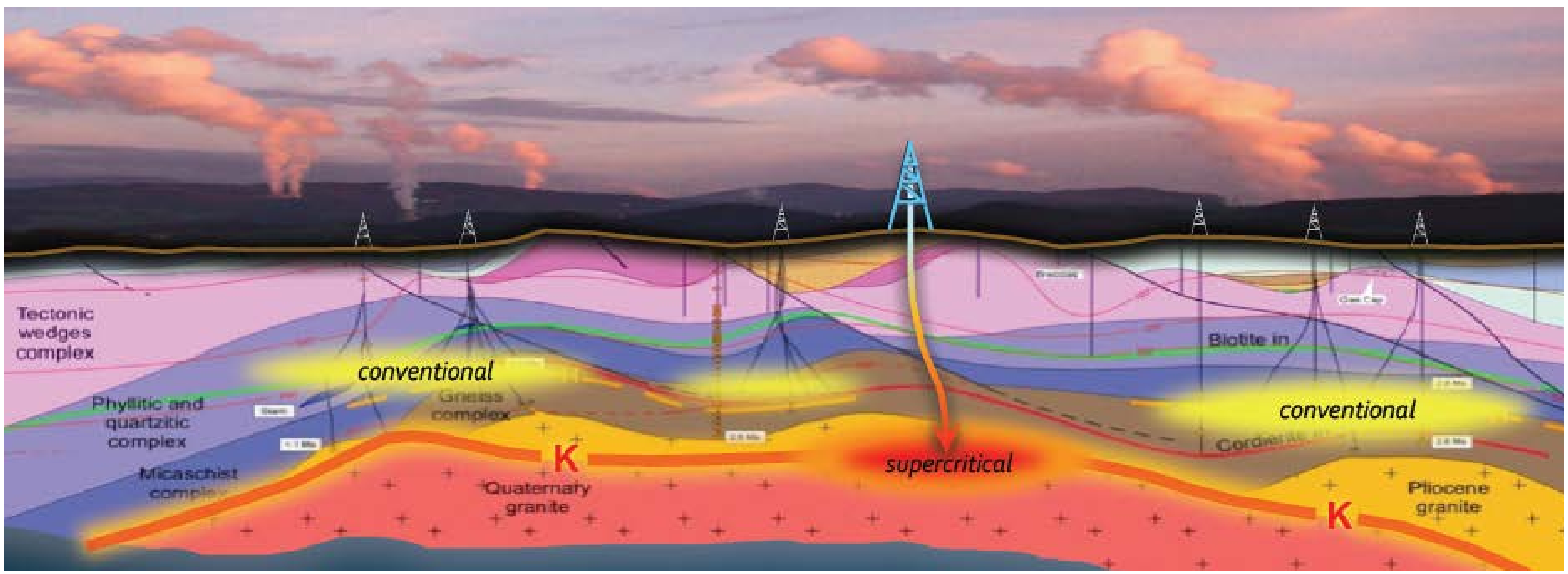
Module: PSO 2D MT
Dataset: COPROD2 (Jones, 1993)
Unknowns= 340 (cells)
Swarm size= 2500

Max iterations= 6000
RMSE= 2.42
Computation-time= 8h
Parallelization= yes

Part 2: PSO – 2D MT

PSO was also adopted for the study of the Larderello geothermal system. Larderello is the perfect natural gym, the oldest field in exploitation in the world and still a research frontier

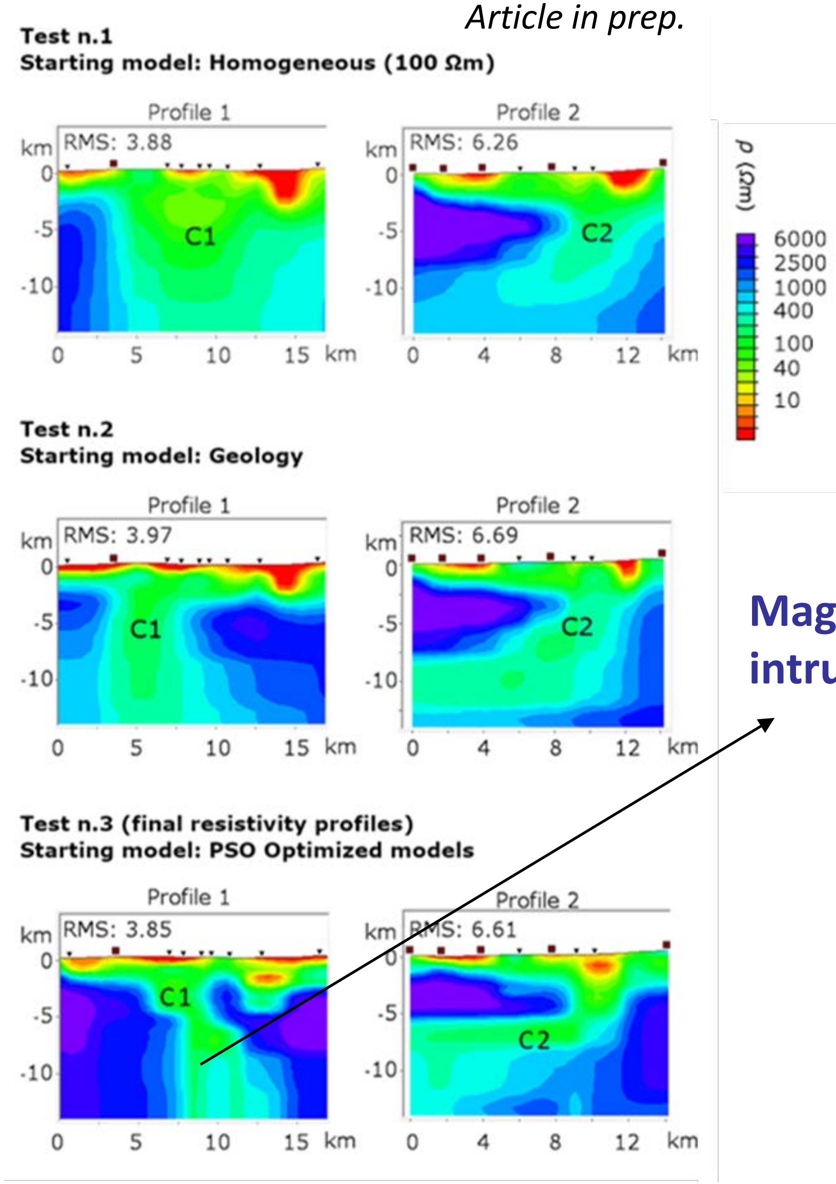
The PSO was tested in addition to conventional 2D and 3D inversions



from Bertani 2017

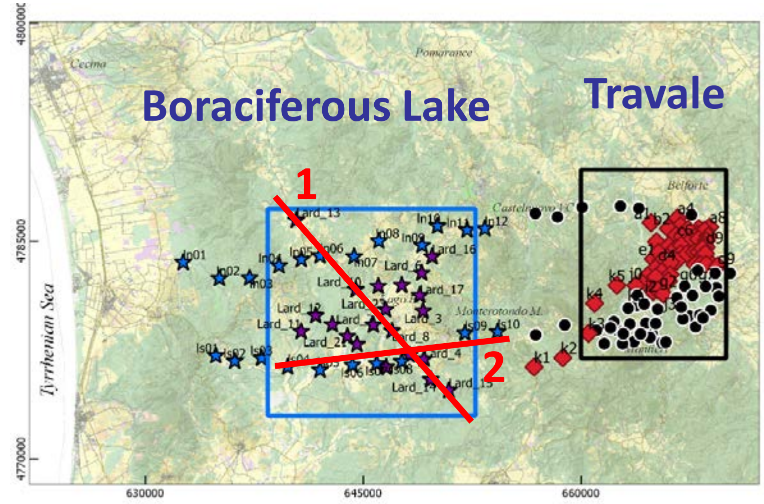
Part 2: PSO – 2D MT

Article in prep.

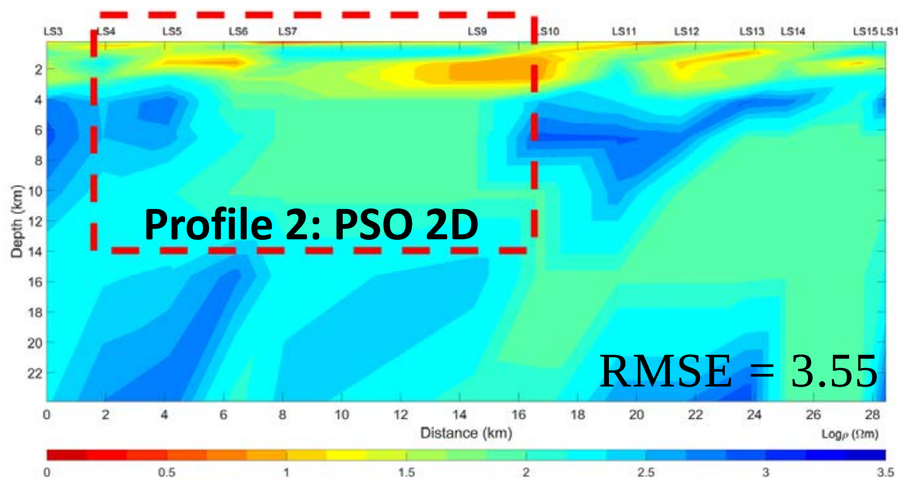


Boraciferous Lake sector.

2D NLCG (Rodi and Mackie, 2001) and 2D MT PSO



Magmatic intrusions



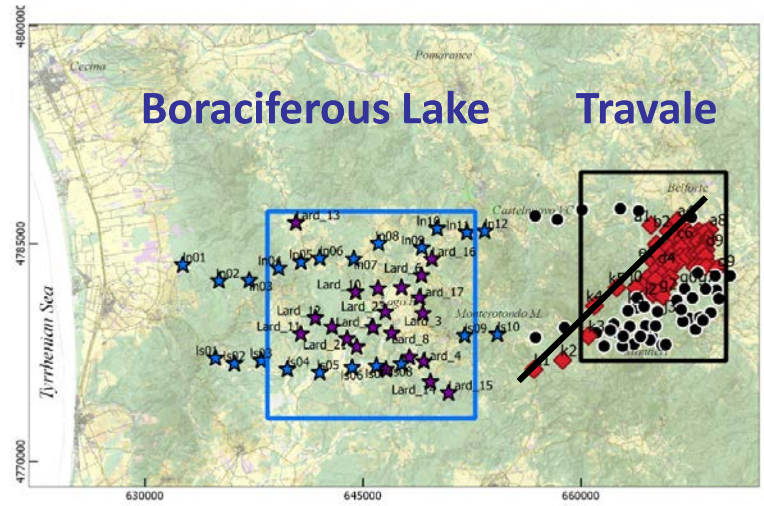
Part 2: PSO – 2D MT

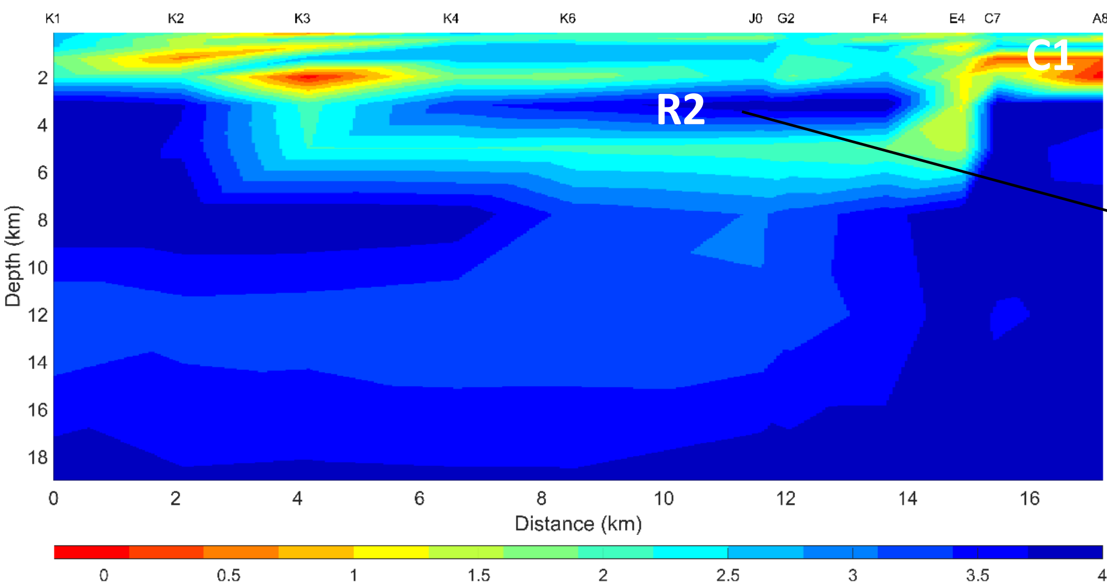
Boraciferous Lake. 2D deterministic (Rodi and Mackie, 2001) and 2D MT PSO

Best trial: 3975 iterations

final RMSE = 4.18

Computation-time on 24 cores (HPC) = 60 h





Vapour-dominated, crystalline reservoir

Part 2: PSO – 2D MT

The detailed study of the Larderello geothermal field deserves much more time to be presented and it is not showed today being out of scope….

Just for curiosity

- The most important conductive and resistive structures retrieved by 2D inversion have been confirmed by large 3D inversion (NLCG, MODEM Kelbert et al., 2014)
- Data integration with hundreds of deep geothermal wells, seismic, seismology, gravity, MT was the basis to solve specific scientific issues (articles under revision and in prep.)
- In the Lago Boracifero sector magmatic intrusions have been imaged whereas in the Travale sector shallow melted intrusions were not recognized (totally crystallized). The hydrothermal circulation was imaged
- We highlighted the role of large tectonic structures in the evolution of the system
- Recently a scientific deep well was drilled in the Lago B. reaching >500°C at 3 km

Part 2: PSO – Joint inverse problem

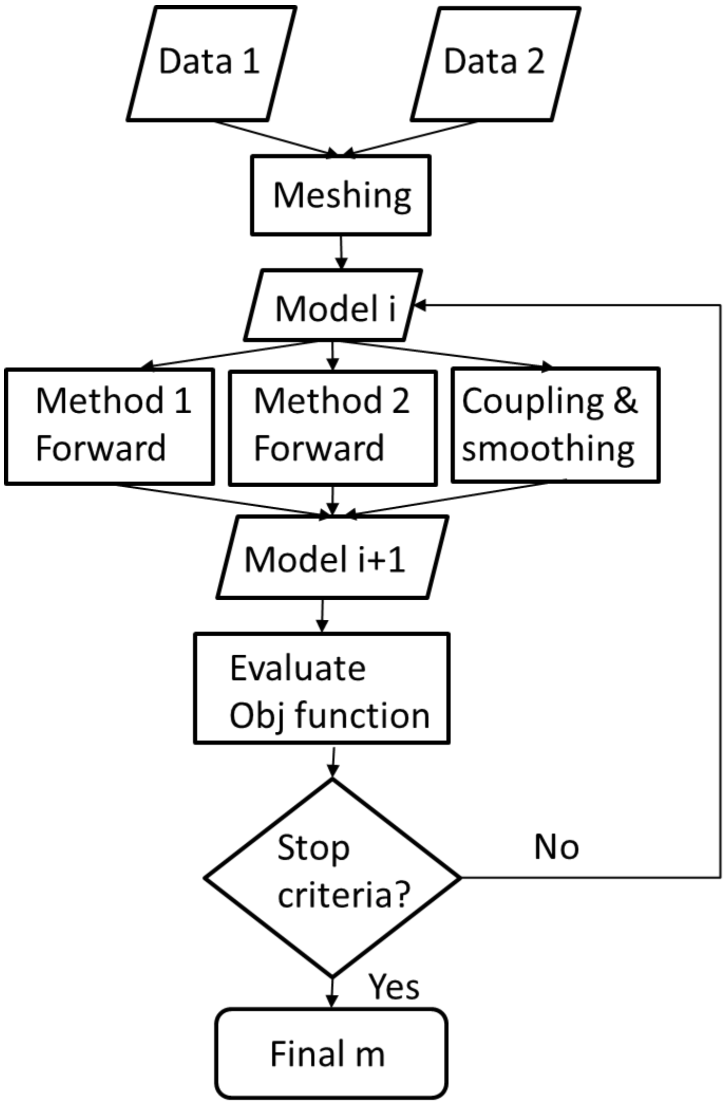
Quantitative data integration: Joint inversion

The joint inversion of different data sets can significantly improve the geophysical modelling:

- overcoming the limitations of each method
- reducing the number of equivalent solutions

Joint inversion common criticalities:

- The weighting factor between different scalar objective functions
- Data compatibility and possible conflicts between the objective functions
- Specifically for CI-based approach, increase of computational complexity



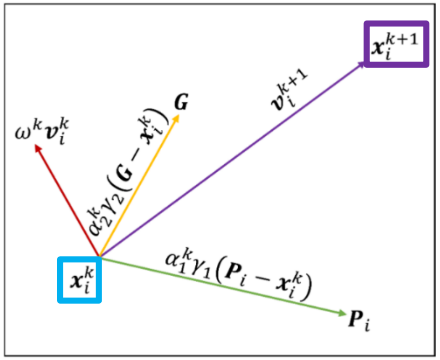
Part 2: PSO – Joint inverse problem

Multi-Objective Evolutionary Algorithms have been adopted in literature:

- deploy a multi-objective optimizer (MOO) to solve a multi-objective problem: no simplification into a single-objective
- deploy a vector objective function: no need of the weighting factor
- simultaneously minimize two (or more) scalar objective functions

We adopted a Time-Variant Multi-Objective PSO (MOPSO) and a mutation operator (similar to genetic algorithm) is applied to enhance global search

α_1^k	2 … 0.5
α_2^k	0.5 … 2
$\boldsymbol{\Phi}^{k}$	0.9 … 0.4
Mutation operator	0.5



In this EMinar, we show the results of our early attempt solving the joint inverse problem of Vertical Electrical Soundings (VES) and TEM.
The problem is simpler due to one-dimension and one physical parameter
Nowadays, we have solved and are validating the joint inverse problem of MT and gravity
The following concepts are valid disregarding dimensionality and number of parameters

Part 2: PSO – Joint inverse problem

Bi-objective optimization of TEM and VES (from Pace et al., 2019b, GJI)

The objective function to be minimized is a vector function:

$$f(\boldsymbol{m}) = [f_1(m),\, f_2(m)]$$

$$f_j(\boldsymbol{m}) = \left\| \frac{(\varphi_o) - (\boldsymbol{\varphi}_c)}{(\sigma_\varphi)} \right\|_2 + \lambda \parallel (\partial \boldsymbol{m}) \parallel_2$$

j=1 for TEM objective,

j=2 for VES objective

m *= [m_1, …, m_p]* vector of electrical resistivity

- The physical parameter to be optimized is the same.
- **The components of the vector function refer to the objective function of TEM and VES computed on the same resistivity model *m***
- In case of different physical parameters (e.g. resistivity and density) the function refers to different models and a coupling factor can be used (e.g. structural or petrophysical)

*Pace et al., 2019b: **Joint optimization of geophysical data using multi-objective swarm intelligence**. Geophys. J. Int. 218, 1502–1521*

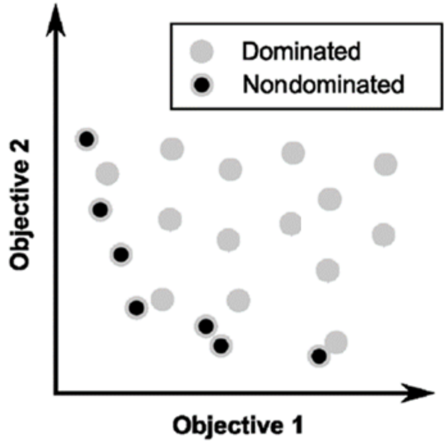
Part 2: PSO – Joint inverse problem

To avoid weighting the components of the objective function, a common approach is to exploit the **Pareto Optimality,** an economic concept developed by Vilfredo Pareto (1896)

Generally, given two solutions $\mathbf{m}_a$ and $\mathbf{m}_b$, a vector $f(\mathbf{m}_a)$ dominates $f(\mathbf{m}_b)$
$(f(\mathbf{m}_a) \preccurlyeq f(\mathbf{m}_b)) \leftrightarrow \forall j \in \{1, 2\},\ f_j(\mathbf{m}_a) \leq f_j(\mathbf{m}_b) \wedge \exists j \in \{1, 2\}: f_j(\mathbf{m}_a) < f_j(\mathbf{m}_b)$

A solution is considered Pareto optimal if there is not another feasible solution that improves one objective without deteriorating the other objective

All the non-dominated solutions form the **Pareto optimal set** (***P****) and are stored in the **repository**. The best model (***m***) is selected from the repository



(modified from Paasche et al., 2014)

The **Pareto Front (PF)** is composed of the corresponding objective functions (TEM and VES) for the solutions forming P*:

$$PF = \{ f(\mathbf{m}) = (f_1(\mathbf{m}), f_2(\mathbf{m}))\ |\ \mathbf{m} \in P^* \}.$$

Part 2: PSO – Joint inverse problem

The adoption of the **Pareto Optimality** provides useful **performance metrics** (Coello Coello et al., 2004):

Repository Index: $\boldsymbol{RI}\ (\%) = \frac{N_{rep}}{N_{tot}}$

Spacing: $\mathbf{SP} = \sqrt{\frac{1}{N_{rep}-1}\sum_{i=1}^{N_{rep}}\left(\bar{d}-d_i\right)^2}$

$$d_i = min_j\left(\left|f_1^i(\boldsymbol{m}) - f_1^j(\boldsymbol{m})\right| + \left|f_2^i(\boldsymbol{m}) - f_2^j(\boldsymbol{m})\right|\right)\ , \qquad i,j = 1,\dots,N_{rep}$$

to measure the spread (distribution) of solutions throughout the Pareto Front

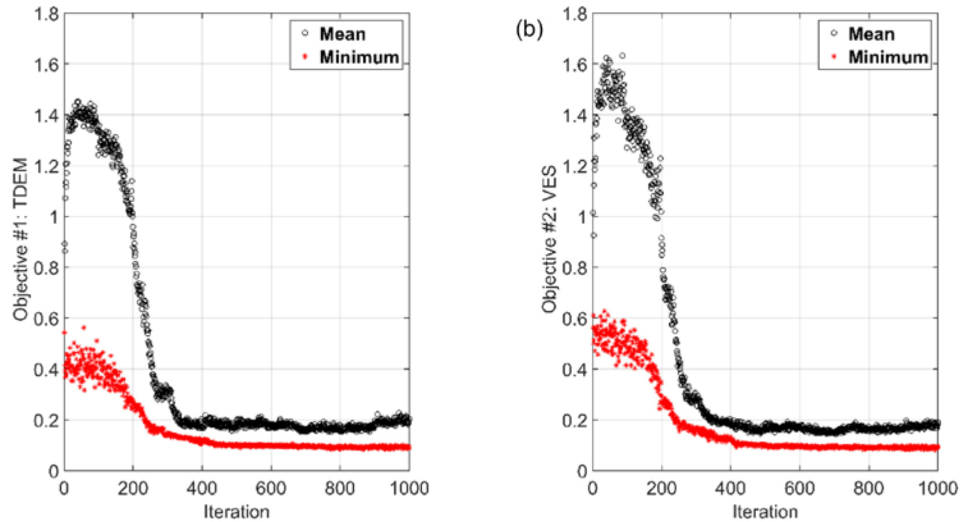
The **deviation angle (α)** $\tan(\alpha) = \left|\frac{\tilde{m}-1}{1+\tilde{m}}\right|$

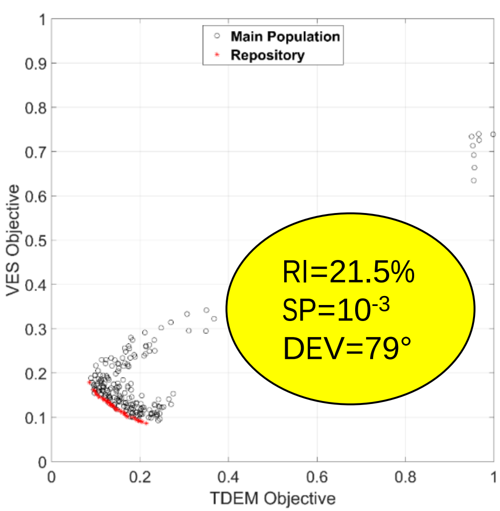
between two lines: the bisector of the objective space and the linear fit of the PF. The metric assess the data compatibility

Part 2: PSO – Joint inverse problem

A first real case study was on a dataset in Piemonte region (Italy) at Stupinigi close to a well (Pace et al., 2019b, GJI).

- **Module**: MOPSO Joint Tem-VES
- **Unknowns**= 19
- **Swarm size**= 200
- **Max iterations**= 1000
- **Boundaries** = 1 – 500 Ωm
- **Model initialization**: Random



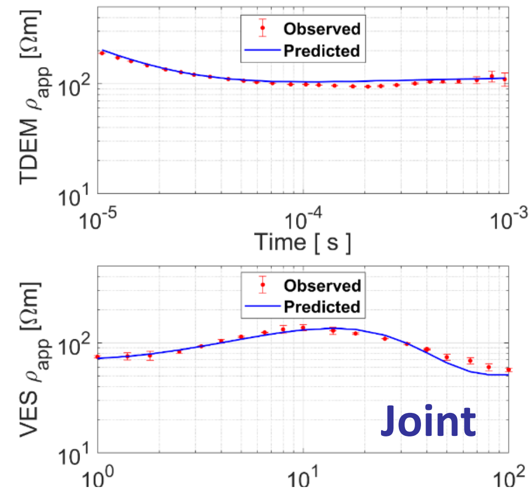


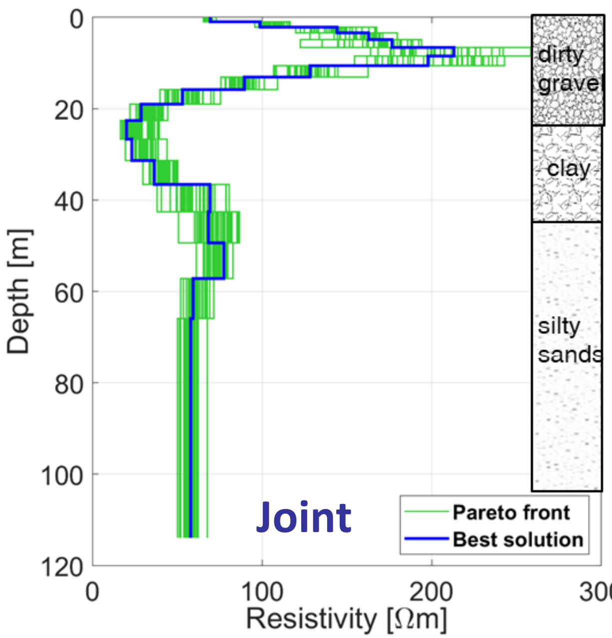
Obtained information on data compatibility and quality of the results

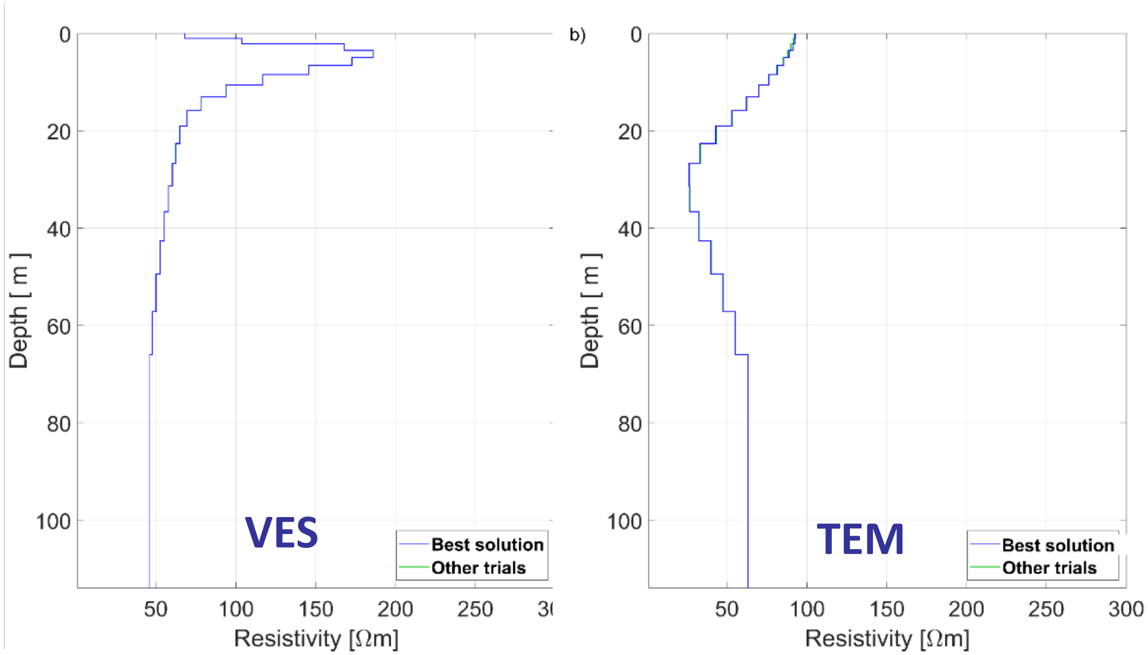
Part 2: PSO – Joint inverse problem

The match with the stratigraphic log is relevant. The limitation of different sensitivity of TEM and VES is overcame

The set of Pareto optimal solution is useful for uncertainty assessment







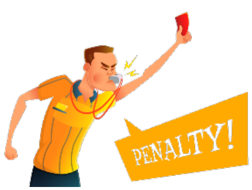
Part 2: PSO – External constraints

The geophysical inverse problem can be constrained by external information
Common practices on deterministic codes usually implies a-priori information on the starting model or more advanced constraining acting on the structure.

PSO can theoretically found an optimum without external information but for more complex problems and real data, external information can be of help to reach the optimum and to speed up the convergence

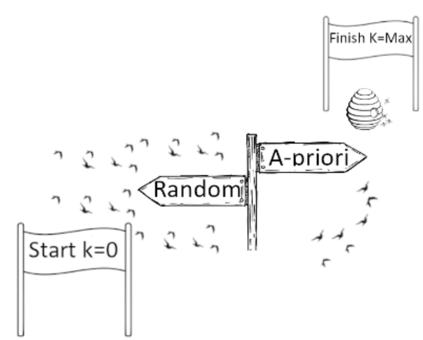
PSO supports various forms to exploit external information:

- **Selectively limit the search space (hard constraint)**



- **Insert a penalty (or a gain) in the objective function related to the model parameters (hard constraint)**

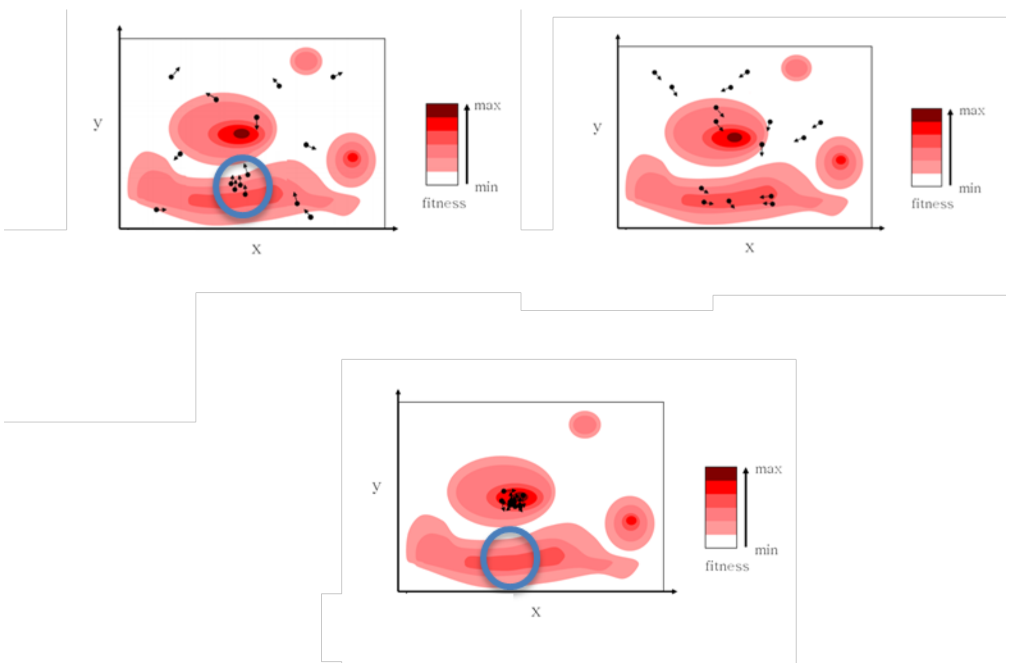
- **Partially guide the initialization of the swarm**



Part 2: PSO – External constraints

We adopted the partial initialization of the swarm with particles that include a-priori information in order to influence the swarm's flight

The idea is to adopt a technique able to consider the external constraints and to exploit or disregard them if necessary, in case the information is actually a bias



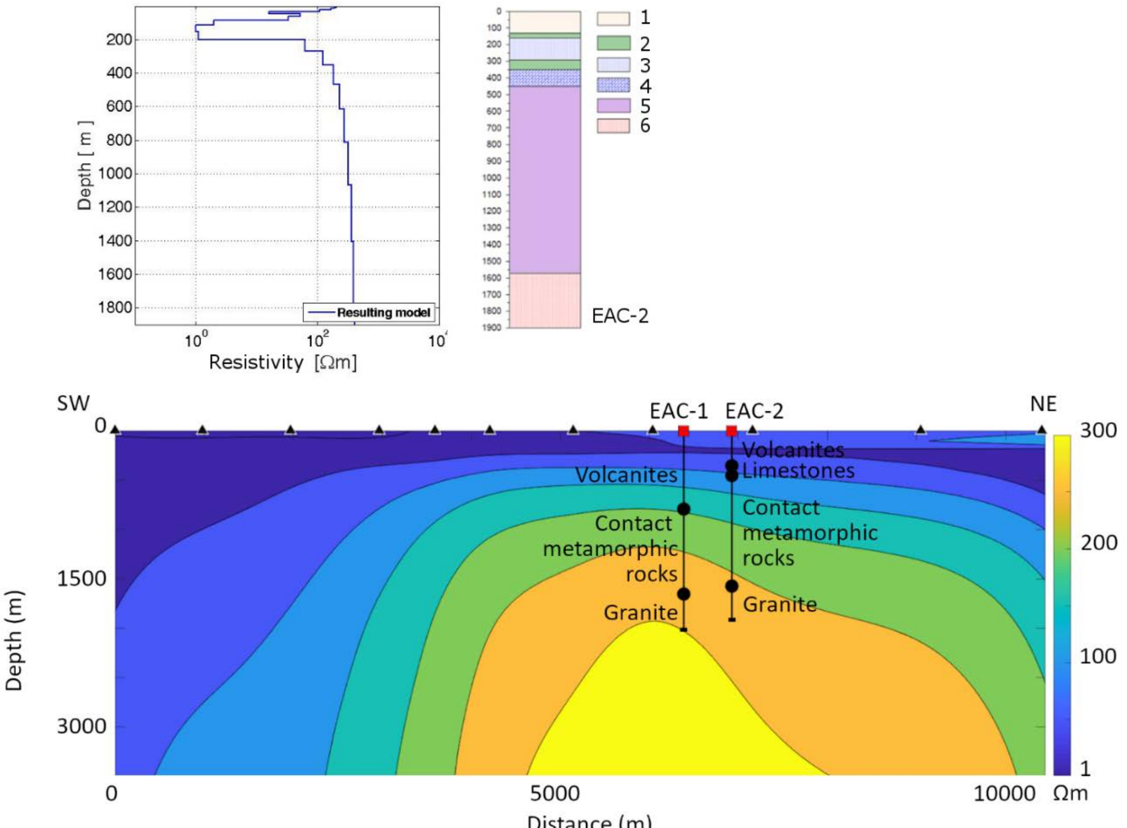
Ideal example of initialized swarm with not relevant information

The introduction of few known particles is a slight constraint and the swarm is able to move across the space if the a-priori earth models are not in agreement with data.

Part 2: PSO – External constraints

The following example of PSO on Vertical Electrical Sounding (VES) is from the GEMex Project aimed at studying the geothermal system of the Acoculco Caldera in Mexico. **The information from geothermal deep wells and the information from nearby VES soundings were used to partially initialize the swarm (a sort of laterally constrained)**

- **Module**: PSOVES
- **Dataset**: Acoculco
- **Swarm size**= 500
- **Boundaries** = 1 – 1500 Ωm
- **Model initialization**: Random+constrained
- **External information**= 5% for each information (well and other models)



Article in prep.

Part 2: Best practices

Practical and useful guidelines to solve the inverse problem by PSO:

1. Evaluate the mathematical complexity and computational load of the geophysical forward problem.
2. Choice of the PSO variant
3. Model discretization. Match the fundamental needs and avoid excessive unknowns
4. Set properly input arguments: they are problem-dependent
   - The accelerations coefficients
   - Swarm size
   - Set the search boundaries properly (avoid excessive search)
   - Formulate properly the objective function
5. Parallelization of the code:
6. When PSO is running, check for effective minimization of the objective function
7. Assess uncertainty with a-posteriori evaluation of the PSO outcome

Part 2: Conclusions

Solving the geophysical inverse problem, specifically the magnetotelluric, is feasible by using swarm intelligence

…

It is feasible but difficult not only in terms of computational demand

Deterministic algorithms are by far conventional for the inversion of MT data
This is not a reason to avoid increasing the knowledge on probabilistic approaches

Stochastics and global search can be of help to:

- **face the dependence of the final solution from the starting guess**
- **look for the global minimum of the function**
- **exploit randomness to assess the uncertainty**

Comprehensive works are required to lay the foundations for more complex problems
Ongoing and future research:

- PSO to solve 2D and 3D gravity inverse problems. Codes in final validation
- PSO to solve joint problem MT and Gravity. Codes in initial validation
- PSO to solve 3D MT inverse problem. Early-stage research

Acknowledgement

Part of the research on the development of algorithms received funds by the EU in the frame of the IMAGE Project (FP7 no. 608553) and GEMex Project (H2020 no. 727550)

Part of the geophysical models are from the analysis of data acquired in the frame of research projects (IMAGE, GEMex, INTAS and FP6 I-GET) or kindly available from ENEL and CFE (Comisión Federal de Electricidad)

The computational resources are from the HPC of the Politecnico di Torino and from CNR-IGG

References

GlobalEM is programmed in Matlab

It includes in-house programmed routines and routines available from literature and further modified:

- The **module 1D PSO MT** exploits a modified PSO code from Chen, S. 2009. Constrained particle swarm optimization. “File Exchange” Environment of MathWorks
- The **module 2D PSO MT** exploits a code for the forward problem by Candansayar, M. E., 2008, Geophysical Prospecting, 56, 141–157. The module exploits a modified PSO code from Ebbesen et al., 2012. A generic PSO Matlab function. IEEE, Extended Abstracts, 1519–1524,
- The **module joint TEM-VES** exploits a modified MOPSO code from Coello Coello et al., 2004: Handling Multiple Objectives with PSO. The forward code are from CR1Dmod by Ingeman-Nielsen and Baumgartner, 2006. Comput. Geosci., 32, 1411–1419 and Ekinci and Demirci, 2008, J. Appl. Sci., 8, 4070–4078

The results presented in this EMinars are from article in preparation or from the following published articles:

- Pace, F., Santilano, A. Godio, A., 2021. **A review of geophysical modeling based on Particle Swarm Optimization**. *Surveys in Geophysics*, 42, 505–549
- Pace, F., Godio, A., Santilano, A., Comina, C. 2019. **Joint optimization of geophysical data using multi-objective swarm intelligence**. *Geophysical Journal International*, 218 (3), 1502-1521
- Pace, F., Santilano, A., Godio, A. 2019. **Particle Swarm Optimization of 2D Magnetotelluric data**. *Geophysics*, Volume 84 (3), E125-E141
- Santilano, A., Godio, A., Manzella, A. 2018. **Particle swarm optimization for simultaneous analysis of magnetotelluric and time-domain electromagnetic data**. *Geophysics*, 83 (3), E151-E159.
- Godio, A. and Santilano, A. 2018. **On the optimization of electromagnetic geophysical data: application of the PSO algorithm**. *Journal of Applied Geophysics* 148, 163-174.

igg

Politecnico di Torino

Complexity

Contact: Alessandro Santilano
alessandro.santilano@igg.cnr.it