Self Consistent ill-posed Inverse Pronlem With Evolutionary Algorithm

August 19, 2015

1 Introduction

In principle, ill-posed inverse problem with the regularization method can give out a very good reconstructed result, but in practice, this is not always the case. The main reason is that we apply our regularization method on the basis of the given, noisy measured value.

In order to solve the ill-posed inverse problem which the measure value or the model is not accurate enougth to give out a good reconstructed result, we introduce the self consistent regularization method With Cross Validation.

2 Methodology

2.1 Fredholm integral equation of the first kind

A Fredholm integral equation of the first kind is an integral equation of the form.

$$f(x) = \int_a^b K(x,t)\phi(t)dt \tag{1}$$

Where $K(x,t)$ if the kerel and $\phi(t)$ is an unknown function to be solved for [1].

The first step to obtain a numerical solution of Eq.1 is to change the integral equation to matrix form by applying discretization method. There are two basically different approaches (Quadrature Methods and Expansion Methods) to do this [2]. In this work, we use quadrature methos for simplicity and easy to implement.

After the discretization, Eq.1 becomes a linear system.

$$Ax = b \tag{2}$$

The elements of the matrix A, the right-hand sized b, and the solution vector x are given by

$$\left.\begin{aligned} a_{ij} &= \omega_j K(x_i, t_j) \\ x_j &= \tilde{\phi}(t_j) \\ b_i &= f(x_i) \end{aligned}\right\} \qquad i,j = 1,...,n. \tag{3}$$

It is known that linear system arising from the first kind Fredholm integral equation such as Eq.1 are likely to be ill-conditioned [5].

2.2 Singular Value Decomposition (SVD) analysis and Picard condition

The singluar value decomposition (SVD) is a very powerful tool for analyzing first kind Fredholm integral equation in the discrete setting as shown in Eq.2 [6].

In this work, we treat the matrix A as a square matrix $A \in R^{N\times N}$, and therefore the SVD of A takes the form

$$A = U \sum V^T = \sum_{i=1}^{N} u_i \sigma_i v_i^T \tag{4}$$

where $\sum \in R^{N\times N}$ is a diagonal matrix with the singular values, satisfying

$$\sum = diag(\sigma_1, ..., \sigma_n), \qquad \sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_n \geq 0$$

The matrix $U \in R^{n\times n}$ and $V \in R^{n\times n}$ consist of the left and right singular vectors

$$U = (u_1, ..., u_N), \qquad V = (v_1, ..., v_N)$$

After we have performed SVD on the matrix A, the solution of x in Eq.2 takes the form

$$x = A^{-1} b = \sum_{i=1}^{N} \frac{u_i^T b}{\sigma_i} v_i \tag{5}$$

For a stable solution x to exist, the right-hand side coefficients $u_i^T b$ must decay to zero faster than the singular values σ_i, a situation that is referred to as the Picard condition. The violation of the Picard condition is the simple explanation of the instablility of the inverse problem, but it also gives a way to deal with it, by employing the regularization method.

2.3 Regularization Method

2.3.1 Truncated Singular Value Decomposition

From the discussion in the previous section, we compute a regularized approximate solution by simply eliminating those SVD componets where the $u_i^T b$ decay to zero more slowly than the sinular values σ_i. Hence we define the truncated SVD (TSVD) solution obtained by:

$$x_k = \sum_{i=1}^{k} \frac{u_i^T b}{\sigma_i} v_i \tag{6}$$

The truncation parameter k should be choosen such taht all the noise-dominated SVD coefficients are discarded. The value of k can be found from an inspection of the Picard plot.

2.3.2 Tikhonov Regularization

TSVD method is very intutitive. The disadvantage is thart it explicityly requires the computation of the SVD. This computation task can be too overwhelming for large-scale problems. Therefore, we need another regularization method which is suited for large computational problem which is the Tikhonov Regularization mehtod.

The Tikhonov solution x_λ is defined as the solution to the problem

$$\min_x \left\{ \|Ax - b\|_2^2 + \lambda^2 \|x\|_2^2 \right\} \tag{7}$$

where, the first term $\|Ax - b\|_2^2$ measures the goodness-of-fit, and the second term $\|x\|_2^2$ measures the regularity of the solution. The balance between the two terms is controlled by the factor λ^2.

The solution of Eq.7 cam also be rewritten as follows in terms of the SVD [3].

$$x_\lambda = \sum_{i=1}^{n} \varphi_i^{[\lambda]} \frac{u_i^T b}{\sigma_i} v_i \tag{8}$$

where $\varphi_i^{[\lambda]}$ is the filter factors and takes the form

$$\varphi_i^{[\lambda]} = \frac{\sigma_i^2}{\sigma_i^2 + \lambda^2} \approx \begin{cases} 1 & \sigma_i \gg \lambda \\ \sigma_i^2/\lambda^2 & \sigma_i \ll \lambda \end{cases}$$

2.4 Noise Analysis

Assuming all the error comes from the Eq.2 is from the right hand side b

$$b = b^{exact} + e$$

assuming e is a white noise.

In order to solve the Eq.7 with the noise biased right hand side b, we need to choose the factor λ. The Discrepancy Principle [4] shows that we should choose the factor λ such that the residual norm $\|Ax_\lambda - b\|_2$ equals the "discrepancy" in the data, as measured by $v_{dp}\|e\|_2$, where v_{dp} is the "safety factor".

$$\|Ax_\lambda - b\|_2 = v_{dp}\|e\|_2 \tag{9}$$

From Eq.9, we know that the goodness-of-fit of the problem is dependent on the error from the right hand side b. The larger the error, the worse the fit. This is the main challenge of using regularized method to solve the inverse problem. If the error from the right hand side is too large, we can only choose a relative large λ to discard the noise-dominated part, which also discard large amound of useful information to reconstruct x.

In order to solve the problem when the noise from the right hand side b cannot be ignored, we introduced a self consistent denoise technique to pre filter the noise. We will search the space $b \pm \eta$, where η is the standard deviation of the white noise as shown in Fig.1, and find the combination can give us the least residual $\|Ax_\lambda - b\|_2$ to form the new b which is very close to b^{exact}. Then we can feed the new b into the regularization process, which can give us a better reconstruct result. Since the search space is a continuous space and can be very large, there is no way to use brute force, or normal search algorithm to search the noise space. We need to use a stochastic search algorithm. Evolutionary Algorithm is the most famous stochastic search algorithm. In this work we use it to search the noise space and find the least residual $\|Ax_\lambda - b\|_2$.

2.5 Evolutionary Algorithm

3 Numerical Results and Discussion

We use a very simple model to illustrate how we use regularization method to solve the ill-posed inverse problem. We use a simplified problem from gravity surveying. An unknown mass distribution with density $f(t)$ is located at depth de below the surface. From 0 to 1 on the t axis shown in Figure. 2. We assume there is no mass outside this source, which produce a gravity field everywhere. At the surface, along the s axis in Figure. 2 from 0 to 1, we mreasure the vertical component of the gravity field, which reger to as g(s).

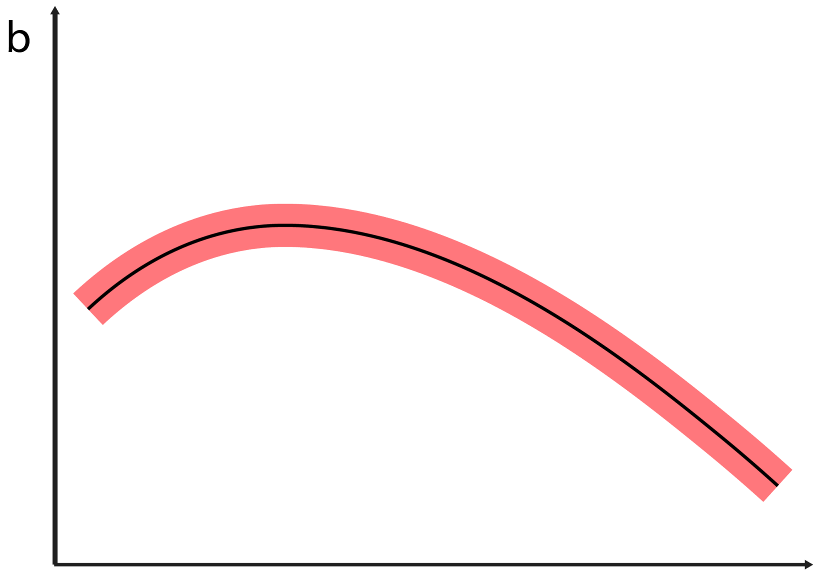


Figure 1: The search space for b^{exact}, the black line is the measured b with noise, the red region is the search sapce, with a width 2η

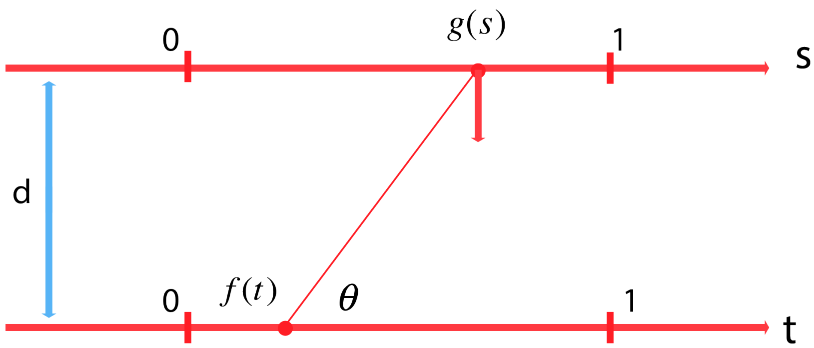


Figure 2: The geometry of the gravity surveying model problem: $f(t)$ is the mass density at t. and g(s) is the vertical component of the gravity field at s.

The two functions f and g are related via a Fredholm integral equation of the first kind. The gravity field from an infinitesmally small part of f(t), of length dt, on the axis is identical to the field from a point mass at t of strength f(t)dt. Hence, the magnitude of the gravity field along s is $f(t)dt/r^{2x}$, where $r = \sqrt{d^2 + (s-t)^2}$ is the distance between the source point at t and the field point at s. The direction of the gravity field is from the field point to he source point, and therefore the measured value of g(s) is

$$dg = \frac{\sin\theta}{r^2} f(t)dt$$

where θ is the angle shwon in Figure. 2. Using that $\sin\theta = d/r$, we obtain

$$\frac{\sin\theta}{r^2} f(t)dt = \frac{d}{\left(d^2 + (s-t)^2\right)^{3/2}} f(t)dt$$

The total value of g(s) for any $0 \leq s \leq 1$ consists of contributions from all mass along the t axis (from 0 to 1). and it is therefore given by the integral

$$g(s) = \int_0^1 \frac{d}{\left(d^2 + (s-t)^2\right)^{3/2}} f(t)dt$$

This is the forward problem and writing it as

$$\int_0^1 K(s,t)f(t)dt = g(s), \ \ 0 \leq s \leq 1 \tag{10}$$

where the function K, which represents the model, is given by

$$k(s,t) = \frac{d}{\left(d^2 + (s-t)^2\right)^{3/2}} \tag{11}$$

and the righ-hand side g is what we are able to measure. The function K is the vertical component of the gravity field, measured at s, from a unit point source locate at t. From K and g we want to compute f, and this is the inverse problem.

Figure.3 shows an example of the computation of the measured signal g(s), given the mass distribution f and three different values of the depth d.

3.1 Discretization of Liner Inverse Problems: Quadrature Methods

We compute approximations $\tilde{f}_j$ to the solution f solely at selected abscissas $t_1, t_2, ...t_n, i.e.,$

$$\tilde{f}_j = \tilde{f}(t_j), \qquad j = 1, 2, ..., n.$$

Quadrature methods–also called Nystrom methods–take their basis in the general quadrature rule of the form

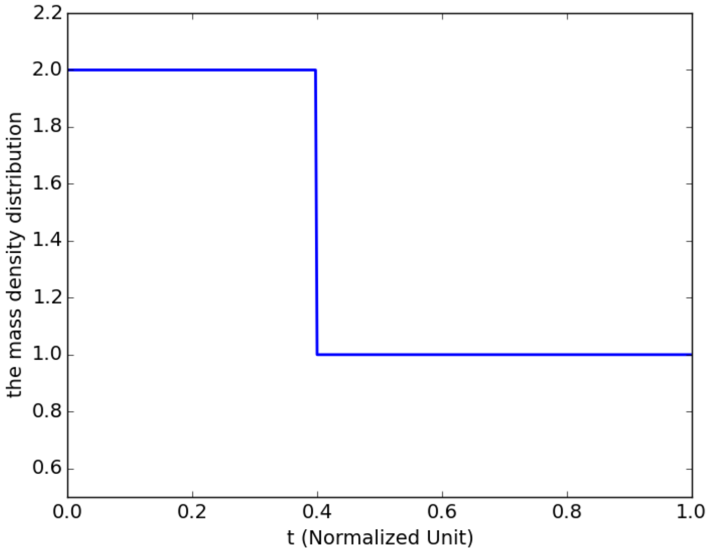
$$\int_0^1 \varphi(t)dt = \sum_{j=1}^{n} \omega_j \varphi(t_j) + E_n$$

where φ is the function whose integral we want to evaluate, E_n is the quadrature error, $t_1, t_2, ..., t_n$ are the abscissas for the quadrature rule, and $\omega_1, \omega_2, \omega_3, ..., \omega_n$ are the corresponding weights. For example, for the midpoint rule in the interval $[0,1]$ we have

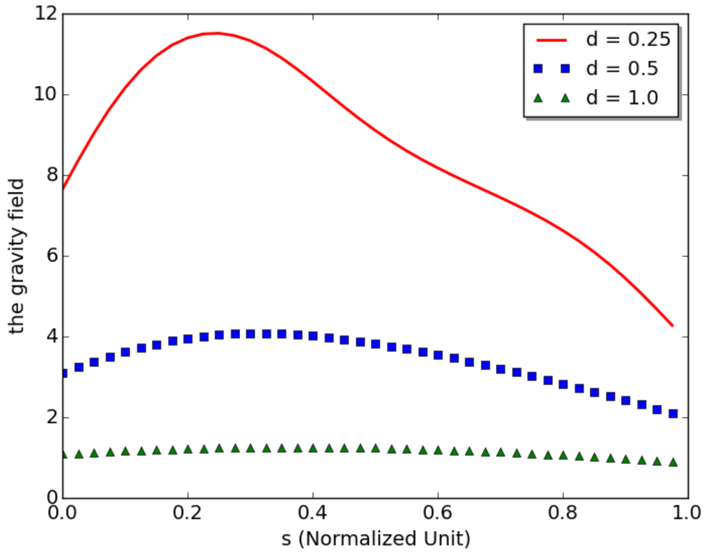
$$t_j = \frac{j - \frac{1}{2}}{n}, \qquad \omega_j = \tfrac{1}{n}, \qquad j = 1, 2, ...n. \tag{12}$$

Apply the quadrature methods on Eq.10, we arrive at the relations

$$\sum_{j=1}^{n} \omega_j K(s_j, t_j)\tilde{f}_j = g(s_i), \qquad i = 1, ...n. \tag{13}$$



(a) f(t)



(b) g(s)

Figure 3: (a) shows the function f(the mass density distribution), and (b) shows the measured signal g (the gravity field) for three different values of the depth d in Figure. 2

The realtion in Eq.18 are just a linear system, which can be also written as

$$\begin{pmatrix} \omega_1 K(s_1,t_1) & \omega_2 K(s_1,t_2) & ... & \omega_n K(s_1,t_n) \\ \omega_1 K(s_2,t_1) & \omega_2 K(s_2,t_2) & ... & \omega_1 K(s_2,t_n) \\ \vdots & \vdots & & \vdots \\ \omega_1 K(s_n,t_1) & \omega_1 K(s_n,t_2) & ... & \omega_1 K(s_n,t_n) \end{pmatrix}$$

or simply $Ax = b$, where A is an $n \times n$ matrix. The elements of the matrix A, the right-hand side b, and the solution vector x are given by

$$\left.\begin{aligned} a_{ij} &= \omega_j K(s_i,t_j) \\ x_j &= \tilde{f}(t_j) \\ b_i &= g(s_i) \end{aligned}\right\} \quad i,j = 1,...,n. \tag{14}$$

3.2 The Singular Value Decompostion

While the matrices that we encountered in the previous section are squre, it can be aslo apply to general case. Hence, we assume that the matrix is either square or has more rows than columns. Then, for any matrix $A \in R^{m\times n}$ with mn, the SVD takes the form

$$A = U\Sigma V^T = \sum_{i=1}^{n} u_i \sigma_i v_i^T \tag{15}$$

Here, $\Sigma \in R^{n\times n}$ is a diagonal matrix with the signluar values, satisfying

$$\Sigma = diag(\sigma_1,...,\sigma_n), \ \ \sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_n \geq 0$$

The matrices $U \in R^{m\times n}$ and $V \in R^{m\times n}$ consistt of the left and right singular vectors

$$U = (u_1,...,u_n) \qquad V = (v_1,...,v_n)$$

and both matrices have orthonormal columns:

$$U^T U = V^T V = I$$

Here we apply the svd to the inverse problem $x = A^{-1}b$, we immediately see that the “naive” solution is given by

$$x = A^{-1}b = \sum_{i=1}^{n} \frac{u_i^T b}{\sigma_i} v_i \tag{16}$$

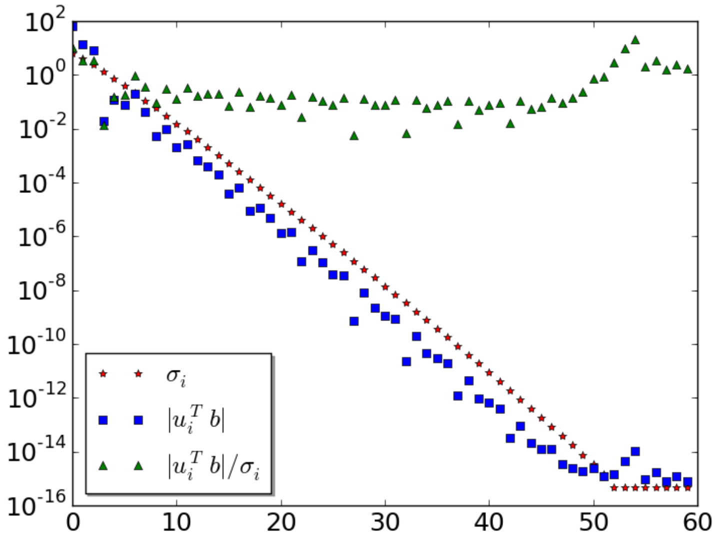
3.3 SVD Analysis and the Discrete Picard Condition

The Discrete picard Condition: Let τ denote the level at which the computed singular values σ_i level off due to rounding errors. The discrete Picard condition is satisfied if, for all singular values larger than τ, the correspoing coefficients $\left|u_i^T b\right|$, on average, decay faster than the σ_i.

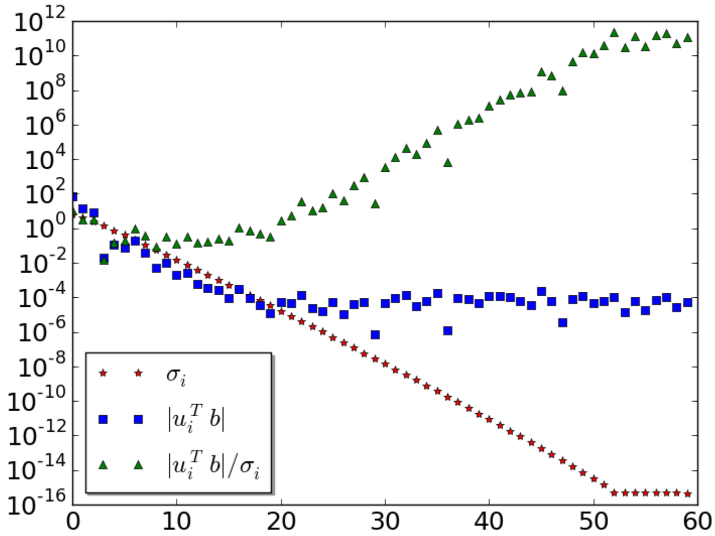
Figure.4 shows the picard plot fro the discretized gravity surveing problem with different noise level.

3.4 Truncated SVD

Figure.5 shows tsvd reconstructed f for the discretized gravity surveing problem with differernt noise level.

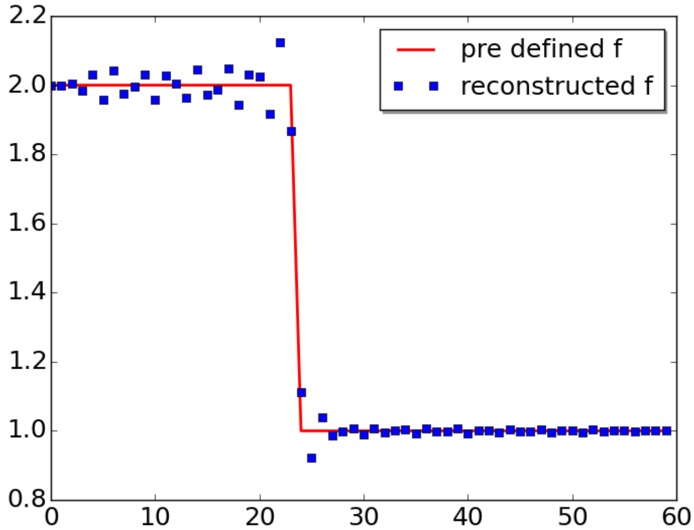


(a) With no noise in the right hand side (only discretization noise)

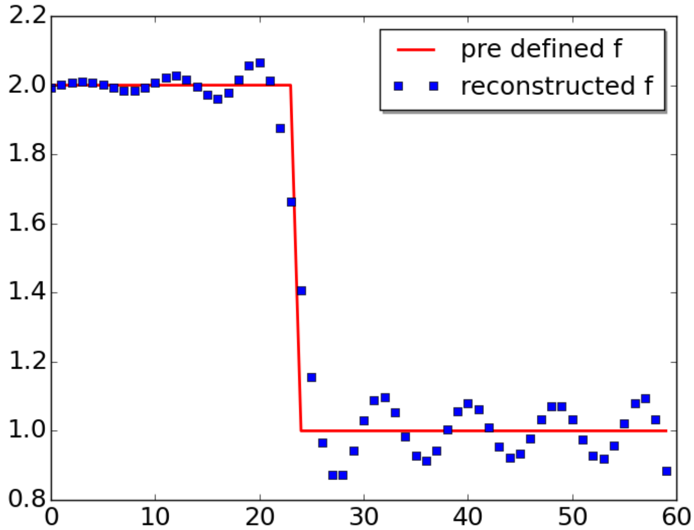


(b) With added noise in the right hand side

Figure 4: The Picard plots for the discretized gravity surveing problem with different noise level. (a) With no added noise (only discretization noise). (b) With an addional white noised added to the measured g (1E-4 random white noise)



(a) With no noise in the right hand side (only discretization noise) truncated number k = 48



(b) With added noise in the right hand side (white noise 1E-4), truncated number k = 18

Figure 5: TSVD f reconstruction

3.5 Tikhonov Regularization

Figure.6 shows Tikhonov Regularization reconstructed f for the discretized gravity surveing problem with differernt noise level.

3.6 Fitness of the Reconstruction: Generalized Cross Validation

The fitness is calculated as:

$$\min_{\lambda} ||Ax_{\lambda} - b^{exact}||_2^2 \tag{17}$$

However, we can't calculate it since b^{exact} is not available. Generalized Cross Validation is a classical statistical technique that comes into good use here [3].

Using Generalized Cross Validation(GSV), the fitness can be calculated as:

$$\min_{\lambda} \frac{||Ax_{\lambda} - b||_2^2}{\left(m - \sum_{i=1}^{n} \varphi_i^{[\lambda]}\right)^2} \tag{18}$$

3.7 Self Consistent Regularization With Cross Validation

4 Cellular Evolutionary Algorithm

4.1 Evolutionary Algorithms

Evolutinary algorithms (EAs) are a family of heuristic search methods that are often used nowadays to find satisfactory solutions to difficult optimization and machine learning problems. EAs are lossely based on a few fundamental evolutionary ideas introduced by Darwin in the niewteenth century. These concepts revolve around the notion of populations of organisms adapting to their enviroment through genetic inheritance and survivial of the fittest.

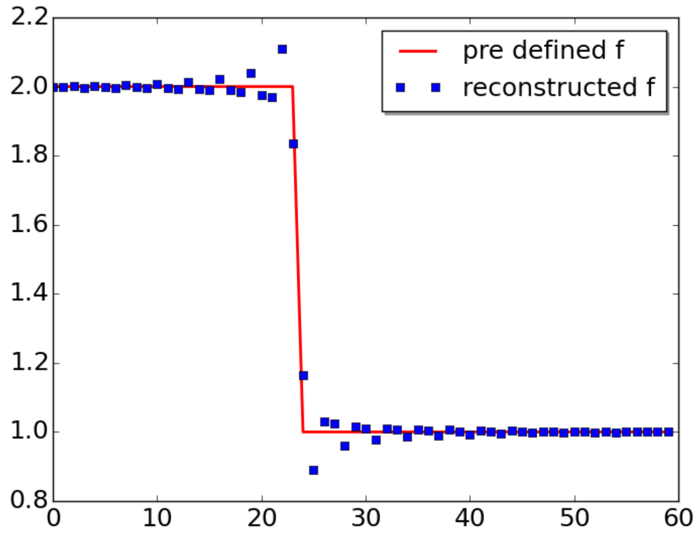
Each individual may be viewed as a representation, according to an appropriate encoding, of a particular solution to an algorithmic problem. To implement an EAs for solving a given problem, at least approximately, the user must provide the following pieces of information.

4.1.1 Representation

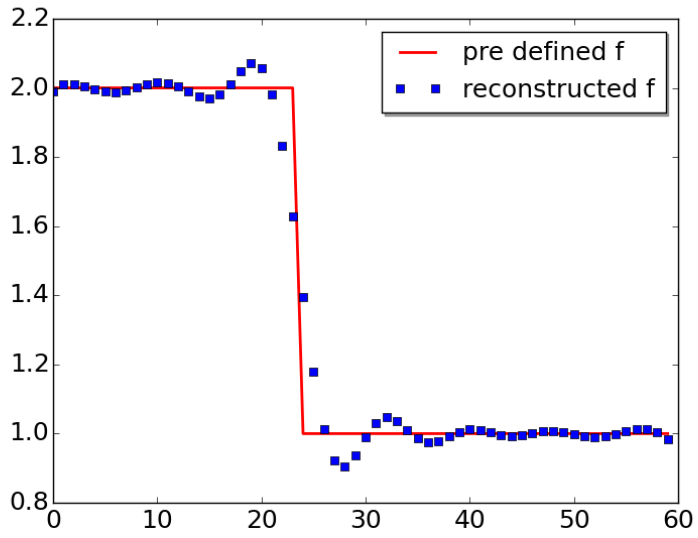
Individuals in the population represent solutions to a pronlem and must be encoded in some way to manipulated by the EA processes. They may be just string of binary digits, which is a widespread and universal representation, or they may be any other data structure that is suitable for the problem at hand such as string of integers, alphabetic characters, strings of real numbers, and even trees or graphs.

4.1.2 Genetic Operators

Usually Evolutionary Algorithm assume that the strucutre of the population is panmictic, which means that any inidividual may interact with any other individual in the population. However, this need not be always the case: we often see population sin the biological and social world in which individuals only interact with a subset of the rest of the population. This situation can usefully be depicated by using the concept of a population graph [4].

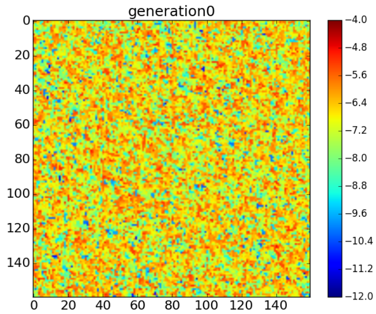


(a) With no noise in the right hand side (only discretization noise), $\lambda =$ 1E-12

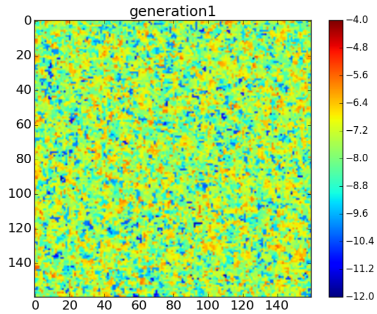


(b) With added noise in the right hand side (white noise 1E-4), $\lambda =$ 1E-3

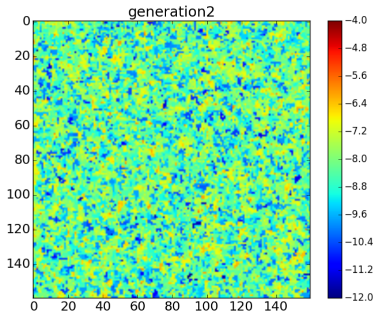
Figure 6: Tikhonov Regularization f reconstruction



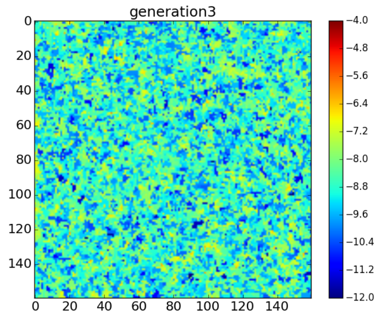
(a) lowest fitness = 2.60e-12



(b) lowest fitness = 1.65e-25



(c) lowest fitness = 1.65e-25



(d) lowest fitness = 1.65e-25

Figure 7: 4 generation of CEA simulation fitness map, we come to the exact de noise measurement of only 2 generation

References

[1] George B Arfken. *Mathematical methods for physicists.* Academic press, 2013.

[2] Wolfgang Hackbusch. *Integral equations: theory and numerical treatment*, volume 120. Birkhäuser, 2012.

[3] Per Christian Hansen. *Discrete inverse problems: insight and algorithms*, volume 7. Siam, 2010.

[4] Alfons G Hoekstra, Jiri Kroc, and Peter MA Sloot. *Simulating complex systems by cellular automata.* Springer, 2010.

[5] Arnold Neumaier. Solving ill-conditioned and singular linear systems: A tutorial on regularization. *SIAM review*, 40(3):636–666, 1998.

[6] Lloyd N Trefethen and David Bau III. *Numerical linear algebra*, volume 50. Siam, 1997.