Background

The aim of this code is to implement SIRT algorithm with different variants. It is especially design for hydraulic tomography inversion usage. To run the code, user should also download MathNet package from NuGet.

Classes

12 classes were declared:

1, *Inv_Main* is the main class.

2, *Inv_SRInfoTable* is a class contains geometrical information, for instance, signals, senders, receivers, travel time parameters, etc.

> 3, *Inv_SRInfoLine* is a description of a signal, which contains a sender, a receiver, the travel time, etc. This class is used only in Inv_SRInfoTable.
>
> > 4, *Inv_SR3D* is a description of a 2D point, its object could be a sender, a receiver or other point type. This class is used only in Inv_SRInfoLine.

5, *InversionParameters* contains the initial diffusivity distribution and some constraints for the inversion, for example, the resolution, the cells with known diffusivity, iteration criteria, etc.

6, *NodesInversion* is a class of node, it is designed for inversion. This nodes class differ from Inv_SR3D and Nodes.

7, *Inv_StraightRayWithConstrain* is an implementation of straight ray tracing, which is a part of SIRT algorithm.

8, *Inv_NetworkWithConstrains* is an implementation of curved ray tracing, which is a part of SIRT algorithm.

9, *SIRT_Options* is the core of SIRT algorithm in matrix form provides different variant options, for example, CAV, DROP, Cimmino, etc.

10, *Dijkstra* is an implementation of Dijkstra algorithm, which finds the path from an origin (node) to a destination (node) with least distance (or time) in a graph.

> 11, *Edge* is a class of path between two nodes in a graph and is used only in Dijkstra. It contains origin, destination and distance properties.
>
> 12, *Nodes* is a simple class of node. It is only used in Dijkstra class and describe node property in a graph.

How to use

Step 1, initialize a 1D array to represent a 2D investigation area [x1, x2]x[z1, z2]:

> Float[4] domain, domain[0] = x1;domain[1] = x2;domain[2] = z1;domain[3] = z2;

Step 2, use a grid to discretize the investigation area into cells. The diffusivity value in cells will be stored in a List

List<double> d

Step 3, initialize objects of class Inv_SR3D with the geometrical data. Every object corresponds a pumping position (sender) or an observation position (receiver).

Step 4, create a signal (a object of class Inv_SRInfoLine) based on the Step3.

Step 5, create a table (a object of class Inv_SRInfoTable) based on Step 4.

Step 6, set parameters as InversionParameters properties.

Step 8, run SIRT() method in Inv_Main.

Example

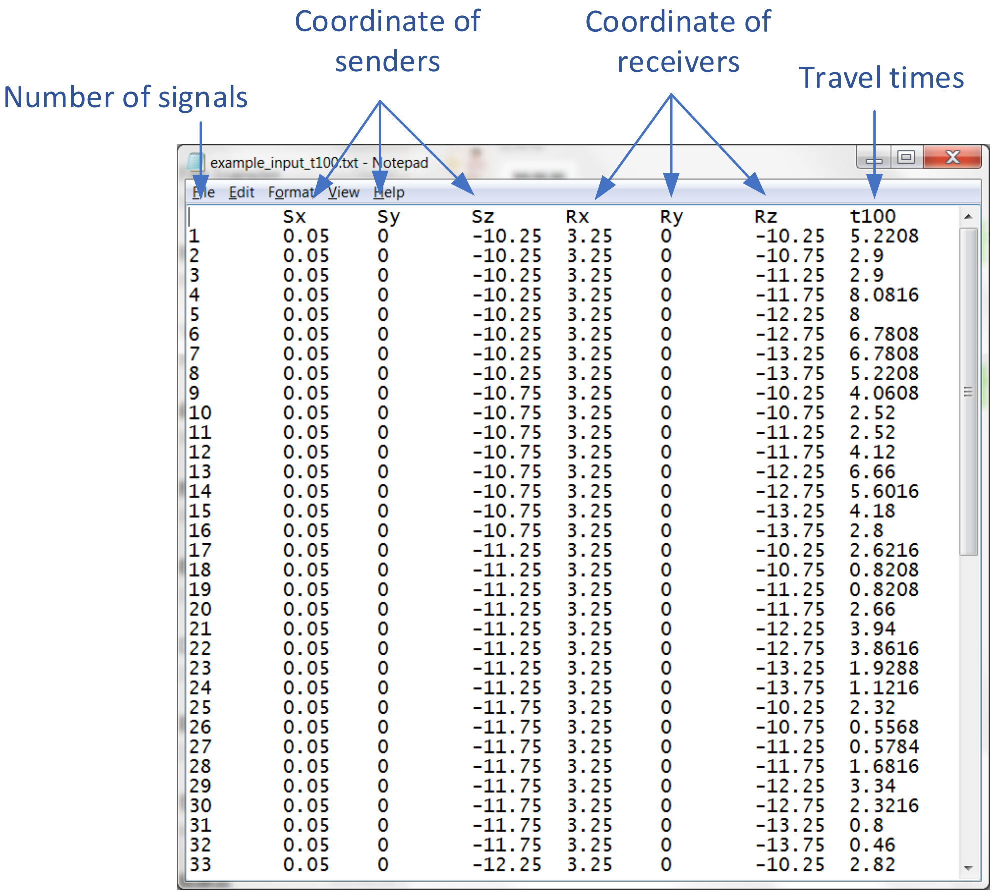
An input example “example_input.txt” is provided. It includes 8 senders, 8 receivers, and 64 signals with travel time (Fig.1). Peak travel time (t100) is used in this example.

Domain as follows: [0.05, 3.25]x[-13.75, -10.25].

Dimfactor 4, P=100, f_alpha_d=1.

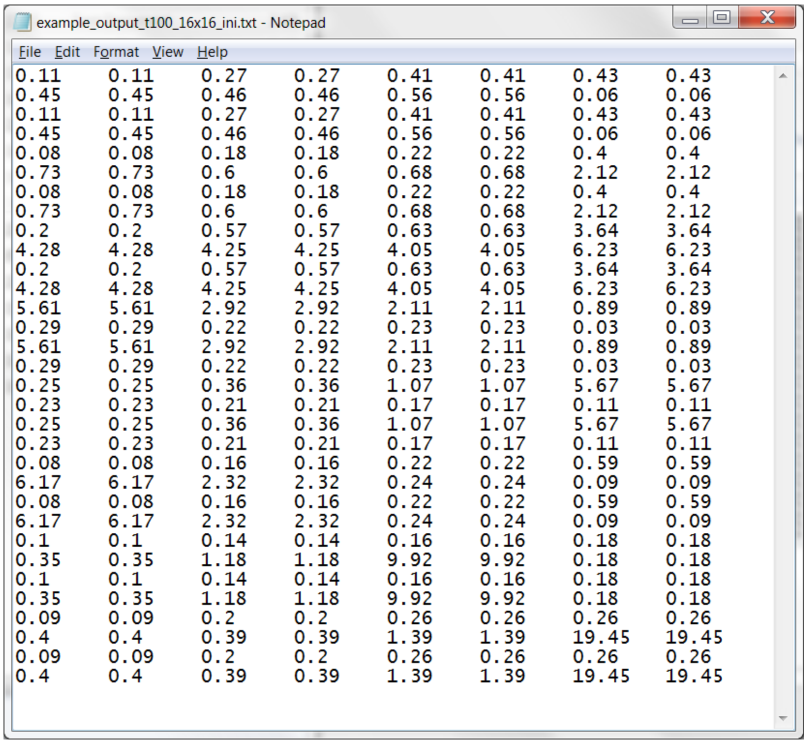
nx=8, nz=8, StrRay=1, CurRay=50.

Use other parameters as default value.



	Sx	Sy	Sz	Rx	Ry	Rz	t100
1	0.05	0	-10.25	3.25	0	-10.25	5.2208
2	0.05	0	-10.25	3.25	0	-10.75	2.9
3	0.05	0	-10.25	3.25	0	-11.25	2.9
4	0.05	0	-10.25	3.25	0	-11.75	8.0816
5	0.05	0	-10.25	3.25	0	-12.25	8
6	0.05	0	-10.25	3.25	0	-12.75	6.7808
7	0.05	0	-10.25	3.25	0	-13.25	6.7808
8	0.05	0	-10.25	3.25	0	-13.75	5.2208
9	0.05	0	-10.75	3.25	0	-10.25	4.0608
10	0.05	0	-10.75	3.25	0	-10.75	2.52
11	0.05	0	-10.75	3.25	0	-11.25	2.52
12	0.05	0	-10.75	3.25	0	-11.75	4.12
13	0.05	0	-10.75	3.25	0	-12.25	6.66
14	0.05	0	-10.75	3.25	0	-12.75	5.6016
15	0.05	0	-10.75	3.25	0	-13.25	4.18
16	0.05	0	-10.75	3.25	0	-13.75	2.8
17	0.05	0	-11.25	3.25	0	-10.25	2.6216
18	0.05	0	-11.25	3.25	0	-10.75	0.8208
19	0.05	0	-11.25	3.25	0	-11.25	0.8208
20	0.05	0	-11.25	3.25	0	-11.75	2.66
21	0.05	0	-11.25	3.25	0	-12.25	3.94
22	0.05	0	-11.25	3.25	0	-12.75	3.8616
23	0.05	0	-11.25	3.25	0	-13.25	1.9288
24	0.05	0	-11.25	3.25	0	-13.75	1.1216
25	0.05	0	-11.75	3.25	0	-10.25	2.32
26	0.05	0	-11.75	3.25	0	-10.75	0.5568
27	0.05	0	-11.75	3.25	0	-11.25	0.5784
28	0.05	0	-11.75	3.25	0	-11.75	1.6816
29	0.05	0	-11.75	3.25	0	-12.25	3.34
30	0.05	0	-11.75	3.25	0	-12.75	2.3216
31	0.05	0	-11.75	3.25	0	-13.25	0.8
32	0.05	0	-11.75	3.25	0	-13.75	0.46
33	0.05	0	-12.25	3.25	0	-10.25	2.82

Fig 1. Input example



0.11	0.11	0.27	0.27	0.41	0.41	0.43	0.43
0.45	0.45	0.46	0.46	0.56	0.56	0.06	0.06
0.11	0.11	0.27	0.27	0.41	0.41	0.43	0.43
0.45	0.45	0.46	0.46	0.56	0.56	0.06	0.06
0.08	0.08	0.18	0.18	0.22	0.22	0.4	0.4
0.73	0.73	0.6	0.6	0.68	0.68	2.12	2.12
0.08	0.08	0.18	0.18	0.22	0.22	0.4	0.4
0.73	0.73	0.6	0.6	0.68	0.68	2.12	2.12
0.2	0.2	0.57	0.57	0.63	0.63	3.64	3.64
4.28	4.28	4.25	4.25	4.05	4.05	6.23	6.23
0.2	0.2	0.57	0.57	0.63	0.63	3.64	3.64
4.28	4.28	4.25	4.25	4.05	4.05	6.23	6.23
5.61	5.61	2.92	2.92	2.11	2.11	0.89	0.89
0.29	0.29	0.22	0.22	0.23	0.23	0.03	0.03
5.61	5.61	2.92	2.92	2.11	2.11	0.89	0.89
0.29	0.29	0.22	0.22	0.23	0.23	0.03	0.03
0.25	0.25	0.36	0.36	1.07	1.07	5.67	5.67
0.23	0.23	0.21	0.21	0.17	0.17	0.11	0.11
0.25	0.25	0.36	0.36	1.07	1.07	5.67	5.67
0.23	0.23	0.21	0.21	0.17	0.17	0.11	0.11
0.08	0.08	0.16	0.16	0.22	0.22	0.59	0.59
6.17	6.17	2.32	2.32	0.24	0.24	0.09	0.09
0.08	0.08	0.16	0.16	0.22	0.22	0.59	0.59
6.17	6.17	2.32	2.32	0.24	0.24	0.09	0.09
0.1	0.1	0.14	0.14	0.16	0.16	0.18	0.18
0.35	0.35	1.18	1.18	9.92	9.92	0.18	0.18
0.1	0.1	0.14	0.14	0.16	0.16	0.18	0.18
0.35	0.35	1.18	1.18	9.92	9.92	0.18	0.18
0.09	0.09	0.2	0.2	0.26	0.26	0.26	0.26
0.4	0.4	0.39	0.39	1.39	1.39	19.45	19.45
0.09	0.09	0.2	0.2	0.26	0.26	0.26	0.26
0.4	0.4	0.39	0.39	1.39	1.39	19.45	19.45

Fig 2. Output example