Geol 335.3

Lab #6: Common mid-point method

In this lab, you will study the principle of the common-midpoint (CMP) reflection method by making simple Matlab simulations. Tools and functions from the previous labs will be useful in this exercise.

Theory

Unlike the common-shot seismic data studied in the previous lab, CMP records are collected by moving both sources and receivers in opposite directions so that their midpoint remains constant while the source-receiver distance increases. CMP gathers are usually presented in the form of time-offset seismic sections.

The primary use of CMP gathers is for *stacking velocity analysis*. There exist many ways to calculate the "semblance" function, which is used to determine the optimal values of stacking velocity V at variable reflection times t_0. In its simplest form, velocity analysis is performed by trying a set of and for each of them, stacking the energy along the reflection hyperbola:

$$Semblance(t_0,V) = \frac{1}{N}\sum_{i=1}^{N} u_i^2\left(\sqrt{t_0^2 + \left(\frac{x_i}{V}\right)^2}\right) \quad , \tag{1}$$

where $u_i(t)$ is the signal it i^{th} channel, and x_i is the source-receiver offset at its location. However, in this lab, we use a different measure that provides smoother peaks:

$$Semblance(t_0,V) = const \times smooth\left\{\left[\sum_{i=1}^{N} u_i\left(\sqrt{t_0^2 + \left(\frac{x_i}{V}\right)^2}\right)\right]^2\right\}, \tag{2}$$

i.e. smoothed squared stack of the waveforms evaluated along the reflection hyperbola.

For a horizontal reflector, the stacking velocity equals the averaged (in the sense discussed in class) velocity above it. When reflector dip α is present, the stacking velocity increases:

$$V_{\text{with dip}} = \frac{V_{\text{no dip}}}{\cos\alpha} . \tag{3}$$

Code

In another copy of your `model.m` file (from lab 5), rename `sources` to `midpoints`. For each midpoint number `n`, you can obtain a CMP section by using a small modification of function `reflection(…)` called `reflection_CMP(…)`. In this function, source positions are variable for each receiver, so that the midpoint is fixed. The resulting section will be produced by

```
sec = reflection_CMP(midpoints(1),receivers,layer,2.0);
```

and plotted by

```
plot_section(offsets(1),sec,’Offset (m)’,'b-')
```

where the relative source-receiver distances (called “offsets”) can be found as (verify this!):

```
offsets = 2*(receivers - midpoints(1)).
```

To compute velocity spectra in eq. (2), we provide function `semblance()`. Look into its code. Note that it is constructed very similarly to `reflection()`, by first initializing a blank semblance and then adding to it contributions from all traces using `interp1()`. The output of `semblance()` also represents a trace section, which can be plotted by `plot_section()`:

```
plot_section(V, sec,’V(km/s)’,'b-'),
```

where `V` is the array of trial stacking velocities.

Assignments

1. [10%] Put only one point into array `midpoints` (in the middle of the model). Execute the modified script `model.m` and pick 2 points to make a horizontal reflecting boundary. Place the boundary in the bottom half of the section.
2. [20%] Add to your program another call to `picklayer()` and pick two more layers (named `layer2`, and `layer3`, for example) shallower than the first one. If you like, also try a third layer.
3. [10%] Generate reflection sections (by using `reflection()`) with different velocities $V = 2$, 1.5, and 1 km/s for the layers, and sum the three sections together.
4. [10%] Using the resulting CMP gather for midpoint #2 (at the middle), compute and plot the velocity spectrum (eq. 2) for stacking velocities `v=0.7:0.05:2.5 m/ms` (note that these units are the same as km/s).
5. [10%] Determine whether the velocities an t_0 set for the three reflectors are correctly determined by the peaks in velocity spectrum. From the plots, describe the velocity resolution (width of velocity peaks) varies with t_0 and velocity.
6. [20%] Run `model.m` again and pick 2 points to make a dipping reflecting boundary close to the depth of the middle reflector in the preceding test. Plot the common-midpoint gather. How does it differ from the horizontal-reflector case? Is it shifted up-dip or down-dip (or not shifted)? What happens if the dip is changed?
7. [10%] Plot the velocity spectrum for the dipping interface case. How do the optimal stacking velocity change? Why? Compare the result to the prediction from formula (2).

8. [10%] Try summarizing the differences of the horizontal and dipping reflector cases.

Hand in:

Zipped directories containing:

1. All Matlab codes (“m-files”);
2. Screen captures or Postscript/PDF figures;
3. Discussion in a Word file.