Using cocrs version 1.1: a manual

Bergler, S., Höcht, G., and Mann, J.

email: *Steffen.Bergler@gpi.uni-karlsruhe.de*

keywords: *Finite-Offset CRS, Offset stacking*

ABSTRACT

The Common-Offset Common-Reflection-Surface stack (CO CRS) has been developed in the last two years as an extension to the established zero-offset (ZO) CRS stack. It provides CO sections from multi-coverage data in a data-driven way. A C++ *code for testing and processing on real data has been developed. The CO CRS stack follows the philosophy of the ZO CRS stack and takes advantage of a multi-parameter stacking surface. However, it is computational more expensive than the ZO CRS stack. Therefore, the CO CRS stack is of interest in cases when target reflectors suffer from bad illumination by normal rays and the acquired data do not contain the necessary information for the simulation a good ZO section.*

INTRODUCTION

The Finite-Offset (FO) CRS stack can provide any FO section from multi-coverage data in a data-driven way as, e.g., CO, common-midpoint (CMP), or common-shot (CS) sections. For the stack into a FO section a five-parameter stacking surface in the midpoint-offset-traveltime data volume is used where the five stacking parameters are determined by means of coherence analysis. If a CO section is to be constructed in this way, we call this approach the CO CRS stack.

Data-driven ZO simulation techniques have proven to be successful in many difficult situations. This means they often yield better results in presence of complex subsurface structures and noisy data compared to conventional imaging methods like, for example, NMO/DMO/stack (see, e.g., Trappe et al., 2001). However, in cases like, e.g., subsalt imaging a standard application of data-driven ZO simulation techniques does not guarantee good imaging results of subsalt structures. Glogovsky et al. (2001) give an explanation for this. They state that the ZO section suffers from bad subsalt illumination by normal rays and, therefore, does not contain the necessary information for a good subsalt image. Bergler et al. (2002b) propose to overcome the problem of lack of subsalt reflection energy in near ZO sections and complex moveouts by the CO CRS stack. The CO CRS method does not rely on the illumination of a subsurface target by normal rays but expects only in the vicinity of the selected point of CO section an approximately hyperbolic moveout.

In this paper, we do not review the theory behind the CO CRS stack. The theoretical background can be found in detail in Zhang et al. (2001) and Bergler (2001). Instead, it is the aim to show the practical realization of the CO CRS stack. A sort of manual to the CO CRS processing software cocrs is given explaining the requirements on hard- and software, the input/output format, and processing parameters. The latest version of the cocrs is online available for sponsors of the WIT consortium on `www.wit-consortium.de` in the restricted area.

HARDWARE AND SOFTWARE REQUIREMENTS

The code should comply to the current C++ standards. As far as we know, no platform specific features are used. Despite the current standards, some environments might expect `.h` suffices for specific C++ header files. So far, the code has been successfully installed on the systems listed in Table 1.

Machine type	Operating system	C++ Compiler
Silicon Graphics Origin 3200	IRIX 6.5	MIPSpro 7.3.1.1m
Silicon Graphics O2	IRIX 6.5	MIPSpro 7.3.1.1m
Pentium IV PC	SuSE Linux 8.0	GNU 2.95.3

Table 1: System configurations with successful installations. Please report other successful installations (and possibly necessary changes of the code) to update this list.

Some efforts were made to reduce the memory resident amount of data—in most cases, only the data in the current aperture are kept in memory.

Input and output are in Seismic Un*x format. We strongly recommend to install the Seismic Un*x package for visualization, pre-processing etc. However, the CO CRS stack implementation itself does not require the Seismic Un*x package or any of its libraries. The Seismic Un*x package is freely available at `http://www.cwp.mines.edu/cwpcodes/index.html`.

INPUT/OUTPUT FORMAT

All input and output files are in Seismic Un*x format, a binary format consisting of records with 240 bytes trace header and up to 2^{15} floats representing the samples of the current trace. Please note that the CO CRS stack implementation expects all data in the native representation of the used platform. For the conversion from/to the native representation, refer to the Seismic Un*x utility suswapbytes. The conversion from Seismic Un*x format to SEG-Y and vice versa is provided by segywrite and segyread, respectively.

PROCESSING PARAMETERS

For processing with the cocrs code several parameters are required, other parameters are optional. For an overview, all parameters with expected type, default value, and short description are listed in Table 2. Parameters which are mandatory for processing and have to be user defined are marked by *required* as default value in Table 2. If cocrs is typed on the command line of the terminal window, with no options or redirects to files, the self-documentation is printed to standard error (the terminal). The self-documentation looks similar as Table 2 with a slightly more detailed description.

In the following, each processing parameter is explained. The explanations are divided into subsections as they appear in the self-documentation. The processing parameters are given in **boldface** with the expected data type in brackets.

Input/Ouput

The datapath and filename of the multi-coverage input data is read from the file **data (string)**. This parameter is required. The CO CRS stack uses the following trace header parameter from the input data:

- Shot and receiver coordinates (SX, GX, SY, GY)
- (Inverse) coordinate scaling factor (SCALCO)
- Temporal sampling and offset (DELRT, DT)
- CDP bin number (CDP).
- Field record number (FLDR)

The current implementation of the CO CRS stack is restricted to 2D data sets, i.e. data are acquired along one line on a plane surface. However, the program also supports quasi-2D data with shot and receivers distributed in a certain vicinity of a straight profile line which must point in either the x- or y-direction of a global coordinate system.

As mentioned before, the requested data format is in Seismic Un*x format. Some SEG-Y formats used in other seismic software differ only in a number of additional bytes at the beginning of the data file. In

Parameter=`value`	Default value	Description
Input/Output		
data=`string`	required	Name of data file to be processed.
data_skip_bytes=`int`	0	Number of bytes skipped at beginning of data file
basename=`string`	required	Prefix of name of output files
check_data		Check if data format is correct
Search strategy		
action=`int,int,int`	required	Number of search parameters in CMP,CO,CRS
icoher=`int`	0	Width of time gate in coherence analysis
Target window		
h=`float`	required	Offset of CO section to be simulated
tmin=`float`	from data	Lower time boundary of CO CRS stacked section
tmax=`float`	from data	Upper time boundary of CO CRS stacked section
dt=`float`	from data	Time sample spacing of CO CRS stacked section.
cdpmin=`int`	from data	Minimum CDP no. of CO CRS stacked section
cdpmax=`int`	from data	Maximum CDP no. of CO CRS stacked section
Velocities		
vs=`float`	required	Velocity at shot
vg=`float`	required	Velocity at geophone
Aperture		
ap_cmpcurv=`float`	from data	Aperture for curvature search in CMP
ap_cmpdip=`float`	from data	Aperture for dip search in CMP
ap_cocurv=`float`	from data	Aperture for curvature search in CO
ap_codip=`float`	from data	Aperture for dip search in CO
Including files		
cmp=`string`		Name of CMP files to include
co=`string`		Name of CO files to include
cs=`string`		Name of CS files to include

Table 2: Processing parameters with default values and short description.

order to process data sets different from Seismic Un*x data, these number of bytes must be omitted which can be specified by **data_skip_bytes (int)**. Default is zero. If the data are correctly read can be checked by the optional parameter **check_data (no type)**. If set the correct input data dimensions are prompted. In case check_data is set, the processing is aborted after reading the input data.

All files generated by the CO CRS stack have filenames of the kind prefix **basename (string)** followed by `.suffix` which uniquely describe their contents. All possible suffices together with the files' contents are listed in Table 3.

Target window

The target window at a selected (full) offset **h (float)** where a CO section is to be constructed can be specified by:

- the midpoint range, where **cdpmin (int)** is the minimum CDP no. and **cdpmax (int)** the maximum cdp no. to be processed. Defaults are the smallest and largest CDP nos. of the input data.
- the time range, where **tmin (float)** is the minimum time and **tmax (float)** the maximum time to be processed. **dt (float)** specifies the time sampling rate. The number of samples per trace are therefore given by the integer of (tmax-tmin)/dt. Defaults of these parameters are the respective properties of the input data.

If the specified extremal values of CDP and time range exceed the extremal values of the input data, these parameters will be automatically adapted to the input data. The above described quantities are illustrated in Figure 1.

Suffix	Contents
`cmp`	Stacked CO section of CMP stack
`cmpcoher`	Coherence section of CMP stack
`cmpnot`	Number of traces contributing to CMP stack
`cmpdip`	Local dip of event in CMP gather
`cmpcurv`	Local curvature of event in CMP gather
`co`	Stacked CO section of CO stack
`cocoher`	Coherence section of CO stack
`conot`	Number of traces contributing to CO stack
`codip`	Local dip of event in CO gather
`cocurv`	Local curvature of event in CO gather
`crs`	Stacked CO section of CRS stack
`crscoher`	Coherence section of CRS stack
`crsnot`	Number of traces contributing to CRS stack
`csdip`	Local dip of event in CS gather
`cscurv`	Local curvature of event in CS gather
`bs`	Angle section
`bg`	Angle section
`k1`	Wavefront curvature section
`k2`	Wavefront curvature section
`k3`	Wavefront curvature section

Table 3: Possible suffices of files generated by the CRS stack. Each group in the table is roughly related to a specific procedure discussed in the subsection search strategy.

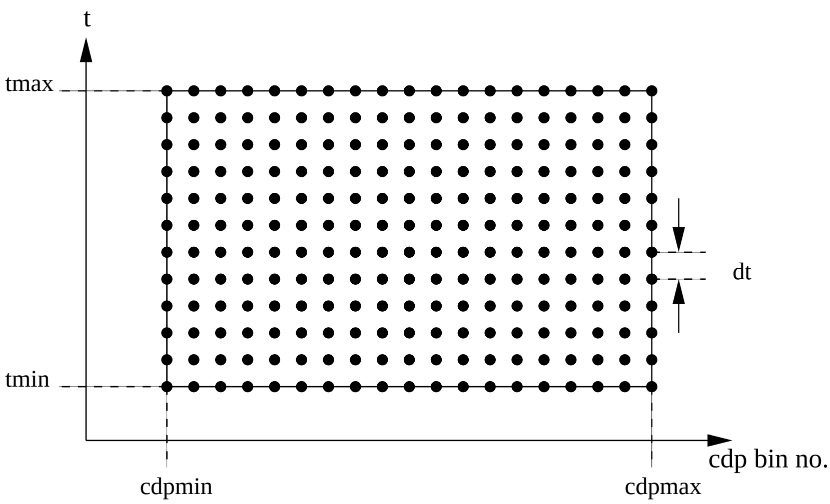

Figure 1: Target window specifications for the CO section to be constructed.

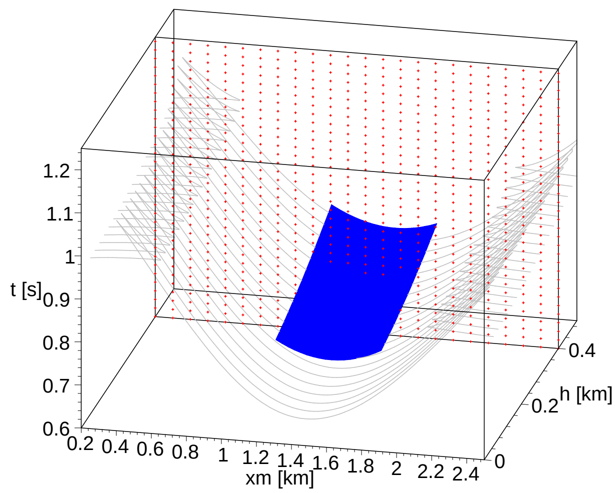

Figure 2: Data volume with traveltime curves of primary reflections of one reflector in depth, grid of CO samples and best fitting stacking surface for one sample.

Velocities

The only velocity information which is required are the near-surface velocities **vs (float)** and **vg (float)** at shot and receiver, respectively. They are necessary to derive the kinematic wavefield attributes (see appendix A) and to constrain the maximal and minimal possible dip of an event (see next subsection).

Search strategy

In the prestack volume, where data are sorted with respect to midpoint (x_m), half-offset (h), and traveltime (t), the implemented CO CRS stacking operator describes a five-parameter surface of the form

$$t^2(\Delta x_m, \Delta h) = (t_0 + a_1\,\Delta x_m + a_2\,\Delta h)^2 + b_{11}\,\Delta x_m^2 + b_{22}\,\Delta h^2 + b_{12}\,\Delta x_m \Delta h\,, \tag{1}$$

where

$$\Delta x_m = x_m - x_0 \qquad \text{and} \qquad \Delta h = h - h_0\,.$$

For each time sample (given by t_0) along each trace (defined by its midpoint x_0) of a CO section to be constructed (defined by its half-offset h_0), the five-parameters a_1, a_2, b_{11}, b_{22}, and b_{12} have to be found that make the CO CRS stacking operator fitting best to the respective reflection event of the data. An example is depicted in Figure 2. There, the gray CO traveltime curves represent the traveltimes of primary reflections with different x_m and h of one reflector in depth. The grid points, in a chosen CO section at half-offset h_0, are locations where the CO CRS stacking operator is searched for. For one point (x_0, h_0, t_0), the five-parameter surface best adjusted to the gray curves in the vicinity of this particular point is displayed.

Mathematically speaking, the search of the CO CRS stacking operator is a non-linear, global, five-parameter optimization problem. Because of the large computation time necessary to perform a simultaneous five-parameter search, the parameter determination is split and conducted in subsets of the data volume. In each gather (CMP, CO, CS, ...) the five-parameter CO CRS formula (1) reduces to a two-parameter hyperbola,

$$t^2_{\mathrm{gat}}(y) = (t_0 + a_{\mathrm{gat}} y)^2 + b_{\mathrm{gat}} y^2\,, \quad \mathrm{gat} = \mathrm{CMP}, \mathrm{CO}, \mathrm{CS}, \ldots\,, \tag{2}$$

with the appropriate meaning of the variable y. The two parameters a_{gat} and b_{gat} control the dip and curvature of the CO CRS operator in the respective gather at the particular sample. A physical explanation of the parameters in terms of wavefront attributes for different gathers is given in Bergler (2001). The formulas relating the parameters to the wavefront attributes are represented in appendix A.

The principle search approach at a particular sample now is the following:

- find the dip and curvature in the offset direction to fit the CMP hyperbola best, i.e. a_{CMP} and b_{CMP}

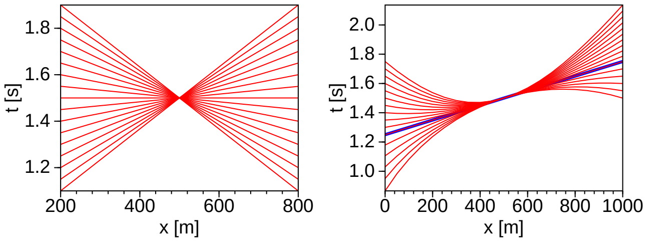

Figure 3: Two one-parameter searches: on the left side the dip determination, on the right side the curvature determination with the found dip value (indicated by the bold line).

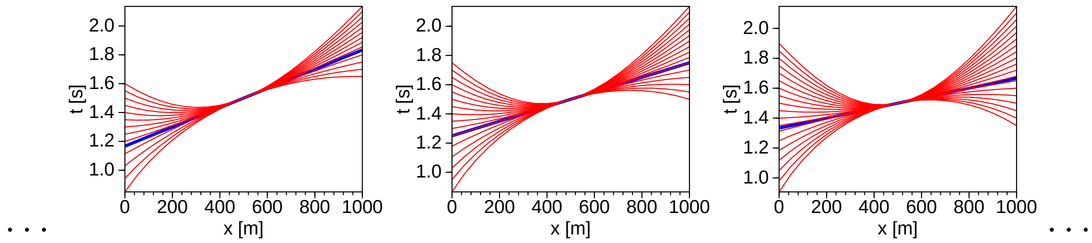

Figure 4: One two-parameter search: a set of hyperbolic curves are tested for each dip value. The dots indicate that many more curves associated with lower and higher dips are part of the search.

- find the dip and curvature in the midpoint direction (CO gather) to fit the CO hyperbola best, i.e. a_{CO} and b_{CO}
- vary the last remaining parameter, which is chosen to be b_{CS}, to fit the CO CRS stacking operator best.

The best fit is determined by means of a coherence analysis with coherence measure semblance. The search steps in the CMP and CO sections can be alternatively performed as two one-parameter searches or one two-parameter search.

For the two one-parameter searches, we firstly determine the dip of the reflection event in the respective gather with the formula

$$t_{\mathrm{gat}}(y) = t_0 + a_{\mathrm{gat}} y\,, \quad \mathrm{gat} = \mathrm{CMP}, \mathrm{CO}, \mathrm{CS}, \ldots\,. \tag{3}$$

Afterwards, b_{gat} in formula (2) is determined with the found, now fixed, value a_{gat}. Figure 3 shows the two one-parameter searches for one sample located at x_0=500m and t_0=1.5s. Please note that the variable x can in fact stand for x_m and h or a linear combination of both. The dip determination is constrained by two criteria. First, the extremal dip may not exceed $\pm(1/v_S + 1/v_G)$. Larger or smaller dips are unphysical. Second, all straight lines tested must completely pass through the search window around (x_0, t_0). This is necessary to make the coherence values associated with different dip values comparable. The extensions of the search window at a sample (x_0, t_0) are specified in lateral directions by the search aperture (see subsection on apertures), and in time direction by either the extremal dip of the event or the extremal time values of the input data. The tested curvature values are also constrained by two criteria. First, the hyperbolic curves must lie inside the area constrained by the straight lines with dip $\pm(1/v_S + 1/v_G)$. Second, all hyperbolic curves must again completely pass through the search window around (x_0, t_0) for the same reasons as stated above.

For the one two-parameter search, the values of a_{gat} and b_{gat} are varied simultaneously. Figure 4 illustrates the situation. For one dip value a set of curvature values are tested. The same procedure is repeated for all possible dip values. The dip values and curvature values are thereby confined by the same criteria as given above.

The selected search strategy can be set by **action ([0,1,2],[0,1,2],[0,1])**. The first value given stands for the strategy in the CMP gather, the second for the strategy in the CO gather, and the last for the final parameter search. The values have the following meaning for the CMP and CO gather:

- 0: no search
- 1: two 1-parameter searches
- 2: one 2-parameter search .

The last value of **action ([0,1,2],[0,1,2],[0,1])** determines if either the search for the last parameter will be performed or not. If one of the values of **action ([0,1,2],[0,1,2],[0,1])** are zero, sections containing the respective $a_{\rm gat}$ and $b_{\rm gat}$ can be included (see subsection on including files).

We recommend to use a search strategy defined by $\texttt{action} = 2, 1, 1$ for real data. The search in the CMP gather is performed with (noisy) prestack data, where a simple dip determination by tested straight lines might fail. The CO dip and curvature values are, however, searched in the CMP stacked CO section. This section normally has a strongly reduced noise level and the two one-parameter searches are mostly sufficient. Moreover, the two one-parameter is computational far less expensive than the one two-parameter search.

The parameters that lead to the best fitting curves and surfaces are determined by coherence analysis with the coherence measure semblance (Neidell and Taner, 1971). To account for the temporal extension of the source wavelet, the coherence criterion is applied to a temporal band centered around the stacking operator. The size of this band can be specified by **icoher (int)** and is given by 2 icoher+1. Default is 0.

With $f_{i,j}$ denoting sample no. j in trace no. i of M contributing traces, the stacking operator represented as $k(i)$, and a symmetric temporal band of width $W + 1$, the semblance coefficient can be written as follows:

$$S_C = \frac{\sum\limits_{j=k(i)-W/2}^{k(i)+W/2} \left(\sum\limits_{i=1}^{M} f_{i,j(i)} \right)^2}{M \sum\limits_{j=k(i)-W/2}^{k(i)+W/2} \sum\limits_{i=1}^{M} f_{i,j(i)}^2} . \tag{4}$$

Therefore, semblance can be interpreted as the normalized ratio of output (stacked trace) to input energy (prestack data).

Aperture

The CO CRS stacking operators are approximations of the kinematic reflection response of curved interfaces in a paraxial vicinity of the central CO ray under consideration. Therefore, it is necessary to define an appropriate aperture inside of which the approximation is sufficiently accurate.

For the dip and curvature search in the CMP gather the apertures are defined by **ap_cmpdip (float)** and **ap_cmpcurv (float)**, respectively. **ap_cmpdip (float)** and **ap_cmpcurv (float)** give the full aperture with the center at the central point (x_0, h_0, t_0). The same applies accordingly for the search in the CO section where the apertures for the dip and curvature search are defined by **ap_codip (float)** and **ap_cocurv (float)**. The aperture for the last parameter search which is also the CO CRS stacking aperture, has an elliptical shape in the offset-midpoint domain with its center at (x_0, h_0). One half-axis is defined by **ap_cmpcurv (float)** in the offset direction and by **ap_cocurv (float)** in midpoint direction. Please note that all aperture parameters are time-invariant in the current implementation. This will be improved in the following versions.

Including files

If results of single search steps are already available, they can be included. For example, if $a_{\rm cmp}$ and $b_{\rm cmp}$ are available, the corresponding sections can be considered by setting the parameter **cmp (string)**. In this case, $\texttt{cmp} = \texttt{basename.cmp}$. The same applies if the CO and CS search parameters should be included, which can be done by setting **co (string)** and **cs (string)**, respectively.

CONCLUSIONS AND FUTURE WORK

The current practical implementation of the CO CRS stack has been shown and the processing parameters explained. Various points of the code are yet expandable, mainly with respect to the choice of apertures. Time-variant apertures will therefore be implemented in the next versions. Another topic which is currently investigated is an AVO dependent coherence measure. Phase shifts and polarity reversals along events are inadequately handled by semblance. These, however, occur when processing far-offset sections with overcritical reflection events. Coherence measures which perform better in such situations are discussed in the recent geophysical literature. These coherences measure will also enter into the next versions of the code.

PUBLICATIONS

Consideration of the CO CRS stack and converted waves have been published in Bergler et al. (2002a).

ACKNOWLEDGEMENTS

This work was kindly supported by the sponsors of the *Wave Inversion Technology (WIT) Consortium*, Karlsruhe, Germany. Especially, we thank Dr. Glyn Jones, ChevronTexaco, for cocrs user's feedback.

REFERENCES

Bergler, S. (2001). The Common-Reflection-Surface Stack for Common Offset - Theory and Application. Master's thesis, Universität Karlsruhe.

Bergler, S., Duveneck, E., Höcht, G., Zhang, Y., and Hubral, P. (2002a). Common-Reflection-Surface stack for converted waves. *Stud. geophysica et geodaetica*, 46:165–175.

Bergler, S., Mann, J., Höcht, G., and Hubral, P. (2002b). The Finite-Offset CRS stack: an alternative stacking tool for subsalt imaging. In *Extended Abstracts, SP 1.8*. 72nd Annual Internat. Mtg., Soc. Expl. Geophys.

Glogovsky, V., Landa, E., and Pfaffenholz, J. (2001). Integrated approach for subsalt depth imaging: a synthetic cas e study. In *Workshop Programme, Session W-5*. 71st Annual Internat. Mtg., Soc. Expl. Geophys.

Neidell, N. S. and Taner, M. T. (1971). Semblance and other coherency measures for multichannel data. *Geophysics*, 36(3):482–497.

Trappe, H., Gierse, G., and Pruessmann, J. (2001). Case studies show potential of Common-Reflection-Surface stack—structural resolution in the time domain beyond the conventional NMO/DMO stack. *First Break*, 19(11):625–633.

Zhang, Y., Bergler, S., and Hubral, P. (2001). Common-Reflection-Surface (CRS) stack for common-offset. *Geophys. Prosp.*, 49(6):709–718.

APPENDIX A

The parameters $a_{\rm gat}$, $b_{\rm gat}$ as well the variable y of equation (2) can be expressed as follows:

i) for the CMP gather $y = \Delta h$ and

$$a_{\rm CMP} = \frac{\sin\beta_G}{v_G} - \frac{\sin\beta_S}{v_S}\,, \tag{5a}$$

$$b_{\rm CMP} = t_0\left(K_3\,\frac{\cos^2\beta_G}{v_G} - K_2\,\frac{\cos^2\beta_S}{v_S}\right)\,, \tag{5b}$$

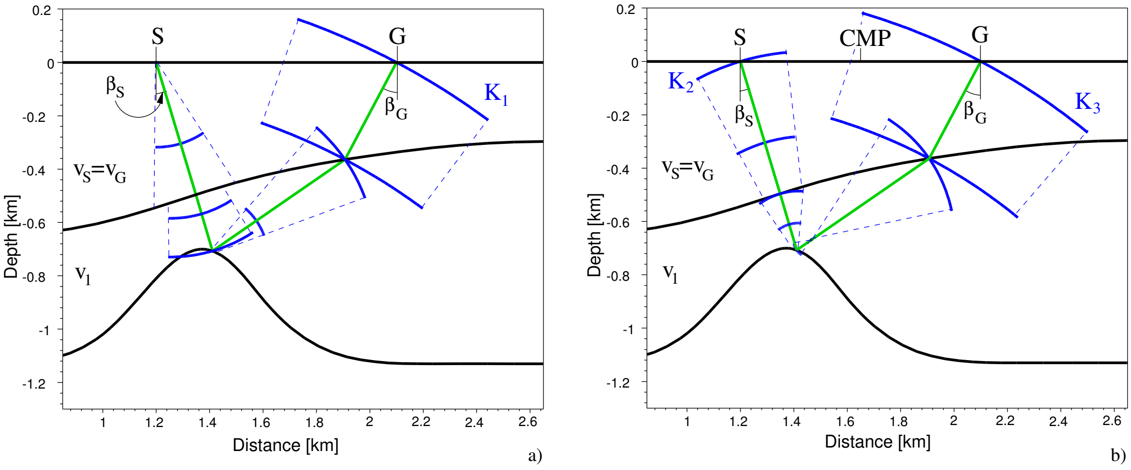

Figure 5: a): CS experiment, b): CMP experiment.

ii) for the CO gather $y = \Delta x_m$ and

$$a_{\mathrm{CO}} = \frac{\sin\beta_G}{v_G} + \frac{\sin\beta_S}{v_S}\,, \tag{6a}$$

$$b_{\mathrm{CO}} = t_0\left((4\,K_1 - 3\,K_3)\,\frac{\cos^2\beta_G}{v_G} - K_2\,\frac{\cos^2\beta_S}{v_S}\right)\,, \tag{6b}$$

iii) and for the CS gather $y = \Delta x_m + \Delta h$ and

$$a_{\mathrm{CS}} = \frac{\sin\beta_G}{v_G}\,, \tag{7a}$$

$$b_{\mathrm{CS}} = t_0\left(K_1\,\frac{\cos^2\beta_G}{v_G}\right)\,. \tag{7b}$$

The new quantities can be explained with Figure 5, where two experiments for a simple model are depicted. The first experiment is the so-called common-shot experiment for which a point source is placed in the shot point of the considered CO ray. The CS wavefront propagates along the down-going ray branch, is reflected at the reflector segment, and propagates back to acquisition surface along the up-going ray branch. This experiment defines three wavefield attributes, namely the curvature of the wavefront K_1 emerging at the receiver and the propagation direction along the FO ray at the source and the receiver, respectively. The propagation directions can be described by the angles β_S and β_G between the CO ray branches and the acquisition surface normal. The second experiment is the so-called CMP experiment: the initial curvature of the wavefront starting at the source is now no longer zero (as in the CS experiment) but takes a finite value K_2. This wavefront also propagates along the CO ray via the reflecting interface to the receiver and emerges with the curvature K_3 . The propagation direction along the CO ray is the same as for the CS experiment.