

ITERATIVE FAULT ENHANCEMENT WORKFLOW

Contents

Introduction	1
Iterative Fault Enhancement Workflow	1
Step 1. Defining input data and data parameters used by all programs in the workflow	/3
Step 2. Defining application program parameters	4
Step 3: Execute the workflow	8
Examples	8

Introduction

A greater degree of filtering can be obtained using three methods: (1) applying more aggressive filter parameters using the original window, (2) applying the original less aggressive filters but using a larger filter window, or (3) cascading the original less aggressive filters using the original filter window in an iterative manner. Iterative application has three advantages over using a larger window size: (1) cascaded filters will implicitly provide a tapered response, (2) cascaded filters adapt to structure, and (3) cascaded filters are more economic to apply than a single filter of the same effective length.

Iterative Fault Enhancement Workflow

The AASPI iterative fault enhancement workflow GUI can be invoked from the main **aaspi_util** by first selecting the AASPI Workflows tab:

🗙 aaspi_util GUI - Post Stack Utilities (Release Date: 24 January, 2018) -	×
Eile Geometric Attributes Spectral Attributes Single Trace Attributes Formation Attributes Volumetric Classification Image Processing	Help
Attribute Correlation Tools Display Tools Other Utilities Set AASPI Default Parameters	
SEGY to AASPI format conversion (multiple files) AASPI to SEGY format conversion (single file) AASPI QC Plotting AASPI Workflows	
AASPI Volumetric Attribute Workflows	
Geometric attribute workflow (Computes common structural, amplitude, and spectral component attribute volumes) AASPI Geometric Attribute Workflow	
Footprint suppression workflow (Adaptively subtracts footprint patterns identified in kx-ky domain) AASPI Footprint Suppression Workflow	
Iterative structure-oriented filtering workflow AASPI Iterative Structure-Oriented Filtering Workflow	
Iterative fault enhancement workflow (Preconditions selected attribute, iteratively applies dLoG filters, and then applies skeletonization AASPI Iterative Fault Enhancement Workflow	

The following workflow GUI will then be displayed (see next page).

Ji Ele AASPL_Utilities Iterative fault enhancement workflow: Step 1: Select Input edge attribute, reflector dip volumes, and processor nodes (these parameters will be used for all workflow programs) Input edge file name (*.H): small/energy_ratio_similarity_d_mig_GSB_small_20_ms_window_broadband.H Optional weight filename (*.H): arf2925/projects/GSB_small_total_energy_d_mig_GSB_small_20_ms_window.H Input inline dip file name (*.H): arf2925/projects/GSB_small/crossline_dip_d_mig_GSB_small_semblance_L2.H Input crossline dip file name (*.H): fs/marf2925/projects/GSB_small/crossline_dip_d_mig_GSB_small_semblance_L2.H Unique Project Name: GSB_small Suffix: semblance_L2 Verbose: Imput crossline dip file name (*.H): Jack 20 processors per node: 2 Node List: Number of iterations: 3 Keep all iteration-generated files? Save workflow environment parameters Step 2 : Save parameter files for desired programs Define parameters for skeletonize3d Define parameters for skeletonize3d gkeletonize3d Step 3 : Execute the iterative fault enhancement workflow Execute Iterative Fault Enhancement Workflow	AASPI - iterative_fault_enhancement		- U	,
Iterative fault enhancement workflow: Step 1: Select Input edge attribute, reflector dip volumes, and processor nodes (these parameters will be used for all workflow programs) Input edge file name (*.H): small/energy_ratio_similarity_d_mig_GSB_small_20_ms_window_H Optional weight filename (*.H): arf2925/projects/GSB_small/total_energy_d_mig_GSB_small_20_ms_window.H Input infine dip file name (*.H): arf2925/projects/GSB_small/coms_dip_d_mig_GSB_small_semblance_L2.H Bro Bro Input erosisine dip file name (*.H): arf2925/projects/GSB_small/crossline_dip_d_mig_GSB_small_semblance_L2.H Bro Bro Inique Project Name: GSB_small Suffix: semblance_L2 Verbose: Image: Semblance_L2 Node List: Image: Semblance_L2 Number of iterations: 3 <	<u> </u>			He
Input edge file name (*.H): _small/energy_ratio_similarity_d_mig_GSB_small_20_ms_window_broadband.H Bro Optional weight filename (*.H): arf2925/projects/GSB_small/total_energy_d_mig_GSB_small_20_ms_window.H Bro Input inline dip file name (*.H): aff2925/projects/GSB_small/inline_dip_d_mig_GSB_small_semblance_L2.H Bro Unique Project Name: GSB_small Suffix: semblance_L2 Verbose: Use MPI: Ø Processors per node: 2 Node List: jade:16 kwiatkowski:16 hematite:16 Number of iterations: 3 Keep all iteration-generated files? Step 2 : Save parameter files for desired programs Define parameters for fallt_enhancement Define parameters for skeletonize3d Step 3 : Execute the iterative fault enhancement workflow Execute Iterative Fault Enhancement Workflow	Iterative fau Step 1: Select Input edge attribut (these parameters will	lt enhancement workflow: e, reflector dip volumes, and processor nodes be used for all workflow programs)		
Optional weight filename (*.H): arf2925/projects/GSB_small/total_energy_d_mig_GSB_small_20_ms_window.H Bro Input inline dip file name (*.H): 6/marf2925/projects/GSB_small/crossline_dip_d_mig_GSB_small_semblance_L2.H Bro Unique Project Name: GSB_small GSB_small Bro Suffix: semblance_L2 Bro Verbose: Image: Comparison of the semblance_L2 Bro Node List: image: Comparison of the semblance of	Input edge file name (*.H):	small/energy_ratio_similarity_d_mig_GSB_small_20_ms_window_broadban	nd.H Br	owse
Input inline dip file name (*.H): \$6/marf2925/projects/GSB_small/inline_dip_d_mig_GSB_small_semblance_L2 H Bro Input crossline dip file name (*.H): harf2925/projects/GSB_small/crossline_dip_d_mig_GSB_small_semblance_L2 H Bro Unique Project Name: GSB_small Starf2925/projects/GSB_small/crossline_dip_d_mig_GSB_small_semblance_L2 H Bro Unique Project Name: GSB_small Starf2925/projects/GSB_small/crossline_dip_d_mig_GSB_small_semblance_L2 H Bro Unique Project Name: GSB_small Starf2925/projects/GSB_small/crossline_dip_d_mig_GSB_small_semblance_L2 H Bro Step 2: Save mode: 2 Step 2 : Save parameter files for desired programs Step 2 : Save parameter files for desired programs Define parameters for fault_enhancement fault_enhancement fault_enhancement gkeletonize3d Step 3 : Execute the iterative fault enhancement workflow fxecute Iterative Fault Enhancement Workflow fxecute Iterative Fault Enhancement Workflow	Optional weight filename (*.H):	arf2925/projects/GSB_small/total_energy_d_mig_GSB_small_20_ms_window	w.H Br	rowse
Input crossline dip file name (*.H): parf2925/projects/GSB_small/crossline_dip_d_mig_GSB_small_semblance_L2.H Unique Project Name: GSB_small Suffix: semblance_L2 Verbose: □ Use MPI: ✓ Processors per node: 2 Node List: jade:16 kwiatkowski:16 hematite:16 Number of iterations: 3 Xeep all iteration-generated files? Save workflow environment parameters Step 2 : Save parameter files for desired programs Define parameters for filter_single_attribute filter_single_attribute fault_enhancement parameters for skeletonize3d skeletonize3d Step 3 : Execute the iterative fault enhancement workflow Execute Iterative Fault Enhancement Workflow	Input inline dip file name (*.H):	6/marf2925/projects/GSB_small/inline_dip_d_mig_GSB_small_semblance_L	2.H Br	rowse
Unique Project Name: GSB_small Suffix: semblance_L2 Verbose: I Use MPI: I Processors per node: 2 Node List: jade:16 kwiatkowski:16 hematite:16 Number of iterations: 3 Keep all iteration-generated files? Image: Save workflow environment parameters Step 2 : Save parameter files for desired programs Image: Single_attribute Define parameters for filter_single_attribute Image: Single_attribute Image: Step 3 : Execute the iterative fault enhancement workflow Image: Single_attribute Execute Iterative Fault Enhancement Workflow Image: Single_attribute	Input crossline dip file name (*.H):	harf2925/projects/GSB_small/crossline_dip_d_mig_GSB_small_semblance_L	2.H Bi	rowse
Suffix: semblance_L2 Verbose: I Use MPI: V Processors per node: 2 Node List: jade:16 kwiatkowski:16 hematite:16 Number of iterations: 3 Xeep all iteration-generated files? Save workflow environment parameters Step 2 : Save parameter files for desired programs Define parameters for filter_single_attribute [filter_single_attribute] Define parameters for fault_enhancement [fault_enhancement] Define parameters for skeletonize3d [skeletonize3d] Step 3 : Execute the iterative fault enhancement workflow Execute Iterative Fault Enhancement Workflow	Unique Project Name:	GSB_small		
Verbose: <pre></pre>	Suffix:	semblance_L2		
Use MPI: Image: Constraint of the second seco	Verbose:			
Processors per node: 2 Node List: jade:16 kwiatkowski:16 hematite:16 Number of iterations: 3 Step all iteration-generated files? - Save workflow environment parameters - Step 2 : Save parameter files for desired programs - Define parameters for filter_single_attribute	Use MPI:			
Node List: jade:16 kwiatkowski:16 hematite:16 Number of iterations: 3 Keep all iteration-generated files? Save workflow environment parameters Step 2 : Save parameter files for desired programs Define parameters for filter_single_attribute Define parameters for fault_enhancement perine parameters for skeletonize3d Step 3 : Execute the iterative fault enhancement workflow Execute Iterative Fault Enhancement Workflow	Processors per node:	2		
Number of iterations: 3 Keep all iteration-generated files? Save workflow environment parameters Step 2 : Save parameter files for desired programs Define parameters for filter_single_attribute [filter_single_attribute] Define parameters for fault_enhancement Define parameters for skeletonize3d Step 3 : Execute the iterative fault enhancement workflow Execute Iterative Fault Enhancement Workflow [Execute Iterative Fault Enhancement Workflow]	Node List:	jade:16 kwiatkowski:16 hematite:16		
Keep all iteration-generated files? Save workflow environment parameters Step 2 : Save parameter files for desired programs Define parameters for filter_single_attribute Define parameters for fault_enhancement Define parameters for skeletonize3d Step 3 : Execute the iterative fault enhancement workflow Execute Iterative Fault Enhancement Workflow Execute Iterative Fault Enhancement Workflow	Number of iterations:	3		
Save workflow environment parameters Step 2 : Save parameter files for desired programs Define parameters for filter_single_attribute [Keep all iteration-generated files?			
Step 2 : Save parameter files for desired programs Define parameters for filter_single_attribute Define parameters for fault_enhancement Define parameters for skeletonize3d Step 3 : Execute the iterative fault enhancement workflow Execute Iterative Fault Enhancement Workflow	Save workflow environment p	arameters		
Step 2 : Save parameter files for desired programs Define parameters for filter_single_attribute Define parameters for fault_enhancement Define parameters for skeletonize3d Step 3 : Execute the iterative fault enhancement workflow Execute Iterative Fault Enhancement Workflow				
Step 2 : Save parameter files for desired programs Define parameters for filter_single_attribute Define parameters for fault_enhancement Define parameters for skeletonize3d Step 3 : Execute the iterative fault enhancement workflow Execute Iterative Fault Enhancement Workflow				
Define parameters for filter_single_attribute	Step 2 : Save parameter files fo	or desired programs		
Define parameters for fault_enhancement	Define parameters for filter_single	_attribute 🗖 _filter_single_attribute		
Define parameters for skeletonize3d <pre>skeletonize3d </pre> Step 3 : Execute the iterative fault enhancement workflow <pre>Execute Iterative Fault Enhancement Workflow </pre> <pre>Define parameters for skeletonize3d </pre>	Define parameters for fault_enhar	ncement 🗖 fault_enhancement		
Step 3 : Execute the iterative fault enhancement workflow Execute Iterative Fault Enhancement Workflow (a) 2000 2010 AACDI. The University of Oldshame	Define parameters for skeletonize	3d 🗖 skeletonize3d		
Execute Iterative Fault Enhancement Workflow	Step 3 : Execute the iterative fault	t enhancement workflow		
(c) 2000 2010 AACDL. The University of Oldshame	Execute Iterative Fault Enhan	comont Worldow		
	Execute iterative Fault Enhan	cement worknow		
(a) 2000 2010 AACDL. The University of Oldshame				

Step 1. Defining input data and data parameters used by all programs in the workflow

The input data consists of (1) an attribute that is sensitive to edges, typically one of the similarity (coherence) volumes, or total aberrancy if we wish to enhance faults that appear as flexures. If we wish to enhance axial planes, one would choose the most-positive principal curvature, k_1 . The confidence in an edge attribute can be sensitive to the amplitude of the seismic reflectivity or to some other measure of signal-to-noise ratio. In this example, we have used energy-ratio coherence as the input edge attribute and (2) total energy as the optional weight. If no weight file is chosen, all edge attribute values are treated equally (i.e. the weight is set to 1.0). The algorithms used in this workflow are sensitive to structure, requiring the use of (3) inline and (4) crossline components of reflector dip.

The unique project name, suffix, and verbose options are the same as in most AASPI applications. The use mpi and number of processors options are common to all AASPI programs that run in parallel (which is the case for **filter_single_attribute**, **fault_enhancement**, and **skeletonize3d** used in this workflow). In this example, we are running across (5) three different workstations, each with 16 processors, for a total of 48 processors in all.

Improved smoothing and sharpening is achieved by defining (6) the *number of iterations*. We've found that for the default, smaller windows, only minimal changes occur beyond three iterations.

If this is your first time running the iterative fault enhancement workflow, you may wish to check (7) *keep all generated iteration-generated files*. In this case with three iterations, there will be a file for fault dip azimuth, fault dip magnitude, and fault probability with the strings "__1", "__2", and "__3" appended to their names. The default is to not keep the intermediate iteration output in order to save disk space.

Finally, (8) click *Save workflow environment parameters*. These parameters will be shared with the successive programs. Once saved, *Step 2* becomes activated, application program by program.

Step 2. Defining application program parameters

Once the parameters are saved, the first application program in the workflow, **filter_single_attribute**, becomes active:

Save workflow environment parameters	
Step 2 : Save parameter files for desired programs	
Define parameters for filter_single_attribute	
Step 3 : Execute the iterative fault enhancement workflow	
Execute Iterative Fault Enhancement Workflow	
(c) 2008-2018 AASPI - The University of Oklahoma	Reset selections

Clicking the **filter_single_attribute** button invokes the corresponding GUI (see next page):

🗙 aaspi_filter_single_attribute GU	l (Release Date: 24 January, 2018)	— C	x I
∬ <u>F</u> ile			<u>H</u> elp
filter_single_attribute - A suit (use program sof3d to filter s	e of image processing filters that can be applied any attribute along structural dip seismic amplitude or impedance data)		<u></u>
Attribute to be filtered (*.H):	_small/energy_ratio_similarity_d_mig_GSB_small_20_ms_window_broadband.H		
Inline Dip (*.H):	6/marf2925/projects/GSB_small/inline_dip_d_mig_GSB_small_semblance_L2.H		
Crossline Dip(*.H):	harf2925/projects/GSB_small/crossline_dip_d_mig_GSB_small_semblance_L2.H		
Unique Project Name:	GSB_small		
Suffix:	semblance_L2		
Verbose:			
Primary Parameters	arallelization parameters		
Filter to apply:	Alpha-trimmed mean filter along structure (averages values, but rejects of att	ribute out	liers) 🔻
Lower and Upper Percentile	, alpha: 50		
MSMTM range, q:	5		
Window half length ():	25.0208		
Window half width ():	50.0416		
Window half height ():	0		
Use rectangular_window?:			
Remove filtered trend from input?:			
Save filter_single_attribute	parameters for a subsequent workflow		
Save parameters and ret	urn to workflow GUI		
(c) 2008-2018 AASPI for Line	ux - The University of Oklahoma Execute filter	_single_a	attribute

Note that the input data at the top of the GUI are grayed out and cannot be changed. They were defined previously by the workflow GUI. The default parameters for this filter are to apply a 50% alpha trimmed-mean filter to the similarity volume using an elliptical window of radius 25 m by 50 m by 0 ms, resulting in a zero-sample thick median filter being applied to the similarity volume along structural dip. This median filter will preserve similarity anomalies parallel to structure and suppress thin anomalies (such as faults) cutting structure at steep angles. The checkmark indicates that this filtered trend will be subtracted from the input similarity volume, resulting in a volume that favor faults, but suppresses anomalies subparallel to structure. Detailed description of this program can be obtained by bringing up the documentation using the Help button in the upper right. After inspecting and perhaps modifying these parameters, simply *Save parameters and return to Workflow GUI*, to return.

At this point, one can select the next application, **fault_enhancement**, in the iterative fault enhancement workflow GUI:

Save worknow environment parameters	
Step 2 : Save parameter files for desired programs	
Define parameters for filter_single_attribute Image: filter_single_attribute Define parameters for fault_enhancement Image: filter_single_attribute Define parameters for skeletonize3d Image: skeletonize3d	
Step 3 : Execute the iterative fault enhancement workflow	
Execute Iterative Fault Enhancement Workflow	
(c) 2008-2018 AASPI - The University of Oklahoma	Reset selections

Clicking the *fault enhancement* button invokes the corresponding GUI:

🗙 aaspi_fault_enhancement GUI (Release	Date: 24 January, 2018)			_		×
<u>F</u> ile						Help
fault_enhancement - Compute the p If the input data are similarity/coher If the input data are curvature volu	probability, dip magnitude ence volumes, the output mes, the output will corre	dip azimuth, and dip strike of locally pla will correspond to fault planes. pond to axial planes of folds.	nar feature:	s in 3D	attrib	ute vo
Input edge attribute filename (*.H):	ratio_similarity_d_mig_GSI	3_small_20_ms_window_broadband_semb	lance_L2.H	Brows	e	
Input inline dip file name (*.H):	6/marf2925/projects/GS	3_small/inline_dip_d_mig_GSB_small_semb	lance_L2.H	Brows	e	
Input crossline dip file name (*.H):	harf2925/projects/GSB_sr	nall/crossline_dip_d_mig_GSB_small_semb	lance_L2.H	Brows	e	
Optional weight filename (*.H):	arf2925/projects/GSB_sn	nall/total_energy_d_mig_GSB_small_20_ms	_window.H	Brows	e	
Unique project name:	GSB_small					
Suffix:	semblance_L2					
Verbose:						
Primary Parameters Paralleli	zation parameters					
Dip1:	10					
Dip2:	25					
Fault Opacity:	-1					
ZNULL value for fault dip magnitud	de: 100					
ZNULL value for fault dip azimuth:	200					
Window half length (m):	75.0624					
Window half width (m):	75.0624					
Window half height (s):	0.0375312					
sigma1 (m):	75.0624					
sigma3 (m):	25.0208					
Use rectangular Window?						
compute_fault_dip_strike?						
Save fault_enhancement paramet	ers for a subsequent wor	<flow< td=""><td></td><td></td><td></td><td></td></flow<>				
Save parameters and return to	workflow GUI					
(c) 2008-2018 AASPI for Linux - Th	e University of Oklahoma		Execute fa	ult_enf	nance	ment

As in the previous application, the input data at the top of the GUI are grayed out and cannot be changed and were defined previously by the workflow GUI. The default parameters for this

filter is to apply a directional derivative of a Gaussian that smooths along the hypothesized fault plane using a Gaussian smoother with σ_1 =75 m, and to sharpen perpendicular to this same hypothesized fault plane using a dLOG sharpening/smoothing parameter of σ_3 =25 m. Detailed description of this program can be obtained by bringing up the documentation using the Help button in the upper right. After inspecting and perhaps modifying these parameters, simply *Save parameters and return to Workflow GUI*, to return.

Save workflow environment parameters	
Step 2 : Save parameter files for desired programs	
Define parameters for filter_single_attribute Image: filter_single_attribute Define parameters for fault_enhancement Image: filter_single_attribute Define parameters for skeletonize3d Image: skeletonize3d	
Step 3 : Execute the iterative fault enhancement workflow	
Execute Iterative Fault Enhancement Workflow	
(c) 2008-2018 AASPI - The University of Oklahoma	Reset selections

At this point, one can select the next application, **skeletonize3d**, in the iterative fault enhancement workflow GUI:

X aaspi_skeletonize3d GUI (Re	lease Date: 24 January, 2018)		_		\times
∬ <u>F</u> ile					Help
skeletonize3d - Compute t	he 3D skeletonized fault image and segmented faults for fault interpretation				^
Fault probability (*.H):	fault_probability_GSB_small_workflow.H	Browse			
Fault dip magnitude (*.H):	fault_dip_magnitude_GSB_small_workflow.H	Browse			
Fault dip azimuth(*.H):	fault_dip_azimuth_GSB_small_workflow.H	Browse			
Unique Project Name:	GSB_small				
Suffix:	workflow				
Verbose:					
Primary Parameters	Parallelization parameters				
Save skeletonize3d para	meters for a subsequentorkflow				-1
Save parameters and	return to workflow GUI				
(c) 2008-2018 AASPI for L	inux - The University of Oklahoma	Exec	ute sk	eletor	nize3d

Note that there are no parameters to change in this GUI. Simply click *Save parameters and return to Workflow GUI* and find that *the Execute iterative fault enhancement workflow* button is now activated.

Step 3: Execute the workflow

Save workflow environment parameters	
Step 2 : Save parameter files for desired programs	
Define parameters for filter_single_attribute filter_single_attribute Define parameters for fault_enhancement fault_enhancement Define parameters for skeletonize3d skeletonize3d	
Step 3 : Execute the iterative fault enhancement workflow	
Execute Iterative Fault Enhancement Workflow	
(c) 2008-2018 AASPI - The University of Oklahoma	Reset selections

At this point simply click Execute Iterative Fault Enhancement Workflow button to submit the suite of processes.

Examples

Detailed examples on the generation of each of these steps can be found in the documentation for **filter_single_attribute**, **fault_enhancement**, and **skeletonize3d**. In this documentation, we simply display the input similarity volume, the results of subtracting the median filtered similarity from the input, the results of three iterations of fault enhancement, and the final skeletonized image.

Vertical slices through these four volumes appear as follows:











Time slices at t=1.28 through these four volumes appear as follows:





