

기술보고

대화식 탄성파 자료 처리 수업을 위한 파이썬 패키지 개발

하완수^{1)*}

Development of a Python Package for Interactive Seismic Data Processing Classes

Wansoo Ha*

(Received 17 June 2015; Final version Received 22 July 2015; Accepted 24 August 2015)

Abstract : This paper presents a 2D data processing package for undergraduate seismic exploration classes. One can process seismic data interactively on IPython Notebook using this Python processing package. There is no commercial license problem, since this package is an open source software. Teachers can use this package as an alternative to commercial data processing packages. Students can practice visualization of seismic data, preprocessing, sorting, gain control, frequency filtering, deconvolution, velocity analysis, dynamic correction, static correction, stacking, and migration using this package.

Key words : Python, IPython notebook, Seismic data processing

요 약 : 본 연구에서는 대학 학부 탄성파 탐사 수업에서 사용할 수 있는 2차원 자료 처리 패키지를 개발하였다. 본 교육용 패키지는 파이썬 언어로 개발하였으므로 IPython Notebook을 이용하여 대화식으로 자료를 처리할 수 있고, 오픈소스이므로 라이선스 문제 없이 간편하게 사용할 수 있다. 교수자는 상용 자료 처리 프로그램들의 대안으로 본 패키지를 활용할 수 있다. 학생들은 본 패키지를 이용하여 탄성파 자료의 시각화, 전처리, 정렬, 이득 조절, 주파수 필터링, 곱풀기, 속도 분석, 동적 보정, 정적 보정, 겹쌓기 및 참반사 보정 과정을 실습할 수 있다.

주요어 : 파이썬, IPython Notebook, 탄성파 자료 처리

서 론

최근 교육계에서 실험 실습을 중요시하는 추세에 따라 탄성파 자료 처리 수업에서도 실습을 시도하는 경우가 많아지고 있다. 반사법 탄성파 탐사 자료의 처리는 과목 특성상 전산실에서 컴퓨터를 이용하여 실습을 진행하게 되며, 학생들은 책 또는 강의 자료의 처리 결과만 볼 때보다 직접 자료를 처리하며 해당 분야에 대해 더 깊이 이해할 수 있다. 전산실에서 자료를 처리할 때에는 자료 처리 소프트웨어를 이용하게 되는데, 자료 처리 프로그램은 크게 상용 프로그램과 오픈소스 프로그램으로 나누어 생각할 수 있다.

상용 프로그램은 자료 처리 소프트웨어 전문 개발 기업에서 작성하여 판매하는 프로그램으로, 대표적인 자료 처리 프로그램으로는 ProMax(Landmark software, 2015),

Geovation(CGG, 2015), Vista, Omega(Schlumberger software, 2015), Echos(Paradigm, 2015), Claritas(Globe Claritas, 2015) 등이 있다. 이러한 프로그램으로 자료 처리 실습을 수행하면 학생들이 졸업 후 해당 프로그램을 사용하는 직장에 진출했을 때 도움이 된다. 대부분의 학교 전산실 컴퓨터가 윈도우즈 운영체제로 설치되어 있는데, 상용 프로그램 중에는 전산실 윈도우즈 컴퓨터에 직접 설치하여 실행할 수 있는 프로그램도 있다. 리눅스 서버가 필요한 경우에도 실습 환경 구성을 위해 회사 전문가의 도움을 받을 수도 있다. 그러나 상용 프로그램의 경우 비싼 라이선스 비용이 들고, 프로그램 코드가 비공개이므로 자료 처리 알고리즘의 세부 구현 방법을 알기 힘들다. 또한 상용 프로그램을 사용하면 일반적으로 학생들이 학교 전산실 외에서는 실습을 할 수 없다.

오픈소스 프로그램으로는 대표적으로 Seismic Un*x (Stockwell, 1999; Sheen *et al.*, 2003), SEP 라이브러리 (Clapp *et al.*, 2004; Ji and Choi, 2010), Madagascar (Madagascar, 2015) 등이 있다. 이러한 프로그램을 실습에

1) 부경대학교 에너지자원공학과

*Corresponding Author(하완수)

E-mail; wansooaha@pknu.ac.kr

Address; Dept. of Energy Resources Engineering, Pukyong National University, Busan, Korea

사용할 경우 라이선스 비용이 들지 않고, 학생들이 궁금할 경우 알고리즘이 어떻게 적용되었는지 소스 코드를 확인할 수 있다. 반면에, 위의 라이브러리들은 모두 리눅스 기반으로 작성되어 있으므로 학교에서 직접 리눅스 서버를 구성하거나 관련 서비스를 이용해야 한다. 리눅스 서버를 사용할 경우 학생 계정 관리 뿐 아니라 학생들에게 기본적인 리눅스 명령어, 셸 스크립트 및 에디터 관련 교육이 필요하다는 부담도 있다. 또한 소스 코드가 공개되어 있기는 하지만, 많은 학생들이 어려워하는 C언어로 작성되어 있다는 점도 단점이다.

다른 종류의 오픈소스 자료처리 프로그램으로, Matlab으로 작성된 프로그램들이 있다(Matlab, 2015; Mousa and Al-Shuhail, 2011; SeismicLab, 2015). 이러한 종류의 프로그램들은 앞의 자료 처리 전용 패키지와는 달리, Matlab에서 불러서 쓸 수 있는 함수 형태로 되어 있다. 주로 교육을 위해 개발된 패키지들로, 전문 자료 처리 패키지에 비해 기능이 제한적이기는 하지만, 실습용으로 사용하기 위해 필요한 기능들은 대부분 갖추어져 있다. 이 경우 학생들은 C언어로 작성된 프로그램에 비해 소스 코드에 쉽게 접근할 수 있다. 그러나 오픈소스 자료 처리 패키지와 달리 이를 실행하기 위한 Matlab은 상용 프로그램이기 때문에 라이선스 비용이 발생하고, 학생이 개인적으로 Matlab을 구매하지 않을 경우 학교 내에서만 실습 가능하다는 한계가 있다.

본 연구에서는 앞에서 언급한 문제점들을 고려하여, 대학의 탄성과 탐사 수업에서 IPython Notebook(IPython,

2015)을 이용해 2차원 자료 처리 실습을 수행할 수 있도록 파이썬 패키지를 개발하였다. 패키지 개발 시 교육을 위해 교수자와 학생들이 쉽게 접근할 수 있도록 하고, 필요할 경우 학생들이 알고리즘 구현 및 적용 방식을 쉽게 이해할 수 있도록 만드는데 중점을 뒀다. 파이썬 패키지를 이용해 자료 처리 실습을 할 경우 인터넷 서비스를 이용하거나 다양한 운영 체제에 설치하여 사용할 수 있기 때문에 별도의 서버를 구성하지 않아도 된다. 라이선스 문제가 없으므로 학생들은 학교 뿐 아니라 집에서도 실습을 해볼 수 있고, 필요할 경우 파이썬 언어로 작성된 소스 코드도 쉽게 읽거나 수정할 수 있다. 논문에서는 먼저 IPython Notebook에 대해 설명하고, 본 연구에서 개발한 자료처리 패키지로 수행할 수 있는 자료처리 예시를 보인다.

파이썬과 IPython Notebook

파이썬은 파이썬 해석기를 통해 작동하는 범용 프로그래밍 언어로, 간결한 문법으로 다양한 기능을 지원한다(Beazley, 2009). 과학, 공학 수치해석 분야에서도 많이 사용되고 있고, 소규모 탄성과 자료 처리에도 생산적으로 사용할 수 있다(Ha, 2014). 파이썬은 C와 같은 정적 자료형 언어에 비해 배우기 쉬운 언어로, 프로그래밍 경험이 있는 학생이라면 본 패키지 사용에 필요한 지식을 몇 시간만에 습득할 수 있다.

IPython Notebook은 파이썬 언어를 웹 브라우저 상에서

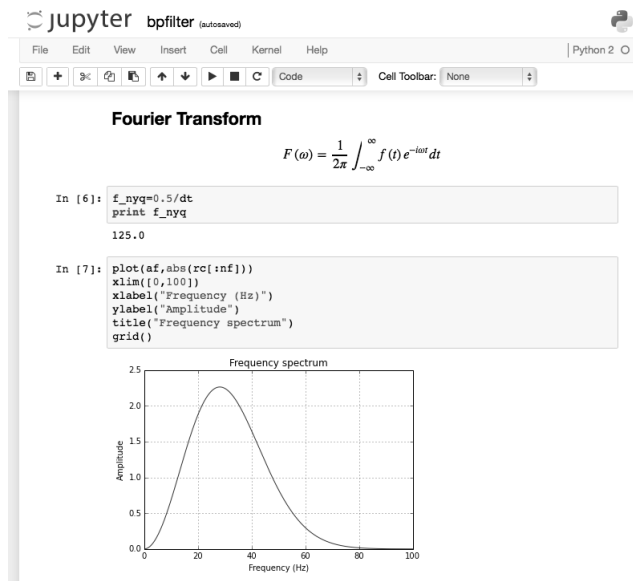


Fig. 1. An IPython Notebook running on a web browser.

실행하며 결과를 확인할 수 있도록 만든 대화식 개발 환경으로, 파이썬 코드 실행 뿐 아니라 텍스트와 수식 입력 및 멀티미디어를 지원한다(Fig. 1). 파이썬 언어가 과학 기술 분야에서 많이 사용됨에 따라 IPython Notebook도 자료 분석이나 통계 분야를 중심으로 널리 사용되고 있다. IPython Notebook에서는 모든 입력과 출력값을 저장하기 때문에 재현 가능한 연구를 위해서도 많이 사용되고 있다(IPython, 2015). 최근에는 Notebook 기능을 파이썬 외에 다른 언어로도 확장해 나가고 있는 중이다(Jupyter, 2015).

Fig. 1에 IPython Notebook 실행 화면을 보였다. Notebook은 셀들로 구성되어 있다. 셀에는 헤더 텍스트, 일반 텍스트, 마크다운(Markdown, 2015), 코드 등의 타입을 지정할 수 있고, 그에 따라 셀의 실행 결과가 달라지게 된다. 마크다운 셀에는 LaTeX(Latex, 2015) 수식도 넣을 수 있다. 코드 셀은 실행하면 셀 아래에 실행 결과가 출력되는데, Matplotlib과 같은 모듈을 이용하여 그림을 출력할 수도 있다. 이러한 특성들을 이용하면 실시간으로 결과를 확인하며 자료 처리 단계들을 수행할 수 있다.

수업을 위한 준비

IPython Notebook을 포함하여 본 연구에서 개발한 자료 처리 패키지를 사용하기 위해서는 크게 세 가지 방법이 있다. 설치가 필요 없는 가장 간편한 방법은 IPython Notebook 개발 환경을 지원하는 인터넷 사이트를 이용하는 방법이다. 대표적인 사이트로 Sage Math Cloud(Sage, 2015)와 Wakari(Wakari, 2015)가 있다. 학생들은 각자 사이트에 계정을 만들고 패키지와 처리할 자료를 올려 실습을 수행할 수 있다. 수업 시간 외에도 인터넷 사용이 가능한 곳에서 자유롭게 실습이 가능하다. 이 경우 자료 처리 패키지 대부분의 기능을 실행할 수 있지만, 별도의 창을 생성하는 속도 분석 부분은 실행할 수 없다. 또한 CPU, 메모리, 하드 디스크, 저장할 수 있는 파일 크기 등 사용할 수 있는 계산 자원이 제한적이라는 단점이 있다.

두 번째 방법으로는 학생들이 실습용 컴퓨터에 직접 파이썬 개발 환경을 설치하는 방법이 있다. Anaconda(Anaconda, 2015) 또는 Canopy(Canopy, 2015) 패키지를 사용하면 윈도우, 맥, 리눅스 운영 체제에서 자료 처리 패키지 실행에 필요한 파이썬 모듈들을 간편하게 설치할 수 있다. 이 경우 실습용 컴퓨터의 계산 자원을 컴퓨터 이용자가 모두 사용할 수 있고, 자료 처리 패키지의 모든 기능을 실행할 수 있다. 별도 서버 관리 문제가 없으므로 실습용 컴퓨터 사양이 뒷받침될 경우 가장 좋은 환경이라 할 수 있다. 학생들의 개인 컴퓨터에서도 동일하게 실행 가능하다.

필요할 경우 별도의 리눅스 서버를 구축할 수도 있다. 이 경우에도 학생들은 웹 브라우저를 통해 서버에 접속하기

때문에 리눅스 서버 명령어들을 새로 배울 필요는 없다. 단, 학생 계정 관리가 필요할 경우에는 별도의 서버 설정과 접속 설정이 필요하다. 또한 학교 정책에 따라 학생이 학교 외에서 실습을 해보기 위해서는 교외 접속 허용 설정이 필요할 수 있다.

파이썬 자료 처리 패키지

본 자료 처리 패키지는 2차원 탐사 자료 처리 교육을 위한 패키지로, 개발을 위해 기본적으로 파이썬 언어와 Numpy, Matplotlib, SciPy 모듈을 사용하였다(Ha, 2014). 속도 분석을 위한 님블(semblance) 계산, 동적 보정, 송곳 곱풀기(spiking deconvolution), 정적 보정, 주시 계산과 키르히호프 참반사 보정 등은 계산량이 많으므로 Cython 모듈을 사용하여 가속하였다(Ha, 2014).

본 자료 처리 패키지는 Matlab 자료 처리 패키지들(Mousa and Al-Shuhail, 2011; SeismicLab, 2015)을 참고

Table 1. Seismic data processing functions available in the package

Processing step	Detail
Visualization	Source, receiver geometry
	Header key values
	Wiggle trace, image plot Frequency spectrum, F-K spectrum
Preprocessing	SU input/output
	Seismic data object input/output
	Header editing, trace editing Windowing
Sorting	By header key values
Gain	Time-power
	Exponential
	Automatic Gain Control
Filtering	Butterworth bandpass filtering
Deconvolution	Spiking deconvolution
Velocity analysis	Semblance picking
Normal moveout correction	NMO correction
	Stretch muting
Static correction	Surface consistent static correction
Stacking	CDP stack
Post-stack migration	Stolt migration
Traveltime calculation	Eikonal equation
Prestack migration	Kirchhoff migration

하여 Matlab 코드를 파이썬 코드로 변환하였는데, 파이썬 언어 특성에 맞게 변환하며 관련된 파이썬 내장 함수나 Numpy, SciPy 모듈의 함수가 있을 경우에는 해당 모듈의 함수들을 사용하였다. 본 패키지로 할 수 있는 자료 처리는 Table 1과 같다. 파이썬 언어를 배우면 필요에 따라 새로운 기능을 추가하거나 기존 기능을 확장해서 사용할 수 있다.

시각화

본 패키지에서는 Matplotlib 모듈을 이용하여 시각화 함수들을 구현하였다(Fig. 2). 그림에는 해상 자료인 Marmousi 자료를 보였다(Versteeg, 1994). 탄성과 채널 기록의 시각화를 위해 단순 채널 기록과 이미지 방법을 지원한다. 이미징 방법에서는 흑백 외에도 Matplotlib에서 지원하는 다양

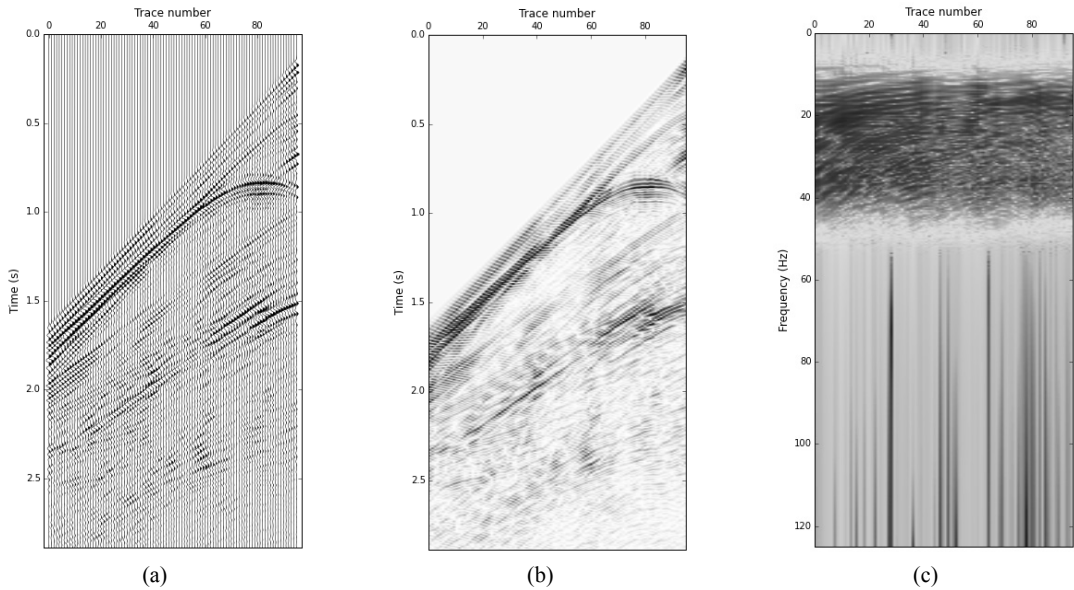


Fig. 2. Plotting examples of the processing package. (a) A wiggle plot, (b) an image plot, and (c) a frequency spectrum of a shot gather.

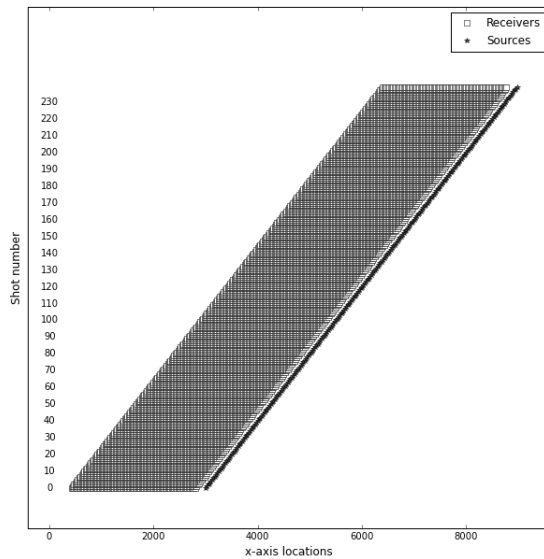


Fig. 3. A chart showing the source and receiver geometry.

한 색상을 이용할 수 있다. 또한 주파수 스펙트럼과 주파수 공간파수 스펙트럼, 자기 상관도, 힐버트 변환을 이용한 데시벨 그림 등을 확인할 수 있다. 탄성파 기록 외에도 Matplotlib 함수들과 탄성파 자료의 헤더 정보를 이용하여 송수신기의 좌표(Fig. 3), 높이, 공통 송신원 모음당 트레이스 개수 등의 정보를 시각화할 수 있다. 이러한 기능들은 자료 처리 시작 전 자료에 대한 정보를 얻기 위해 사용할 수도 있고, 자료 처리 중간에 처리 결과를 확인하며 품질 관리에도 사용할 수 있다. 시각화한 그림은 필요할 경우 다양한 형식의 그림 파일로 저장이 가능하다.

탄성파 자료 객체와 전처리

본 패키지에서 탄성파 자료는 트레이스 헤더와 시계열 자료를 포함하는 객체로 정의되어 있다. 편의상 한 파일의 모든 트레이스를 하나의 객체에 저장한다. 트레이스 헤더에 들어가는 정보는 SU의 트레이스 헤더에 들어가는 정보와 동일하고, 시계열 자료는 배열도 실수로 저장한다. 객체 내에는 자료 처리 과정과 사용한 변수들을 기록하여 필요할 경우 자료 처리 과정을 살펴보거나 동일한 과정을 재현할 수 있도록 하였다. 본 패키지를 사용하면 SU 파일을 읽거나 쓸 수 있다. 파이썬 객체 자체를 파일로 출력하여 자료 처리 중간 과정을 저장한 후 다시 읽어 들여 처리할 수도 있

다. 파이썬 객체를 사용하면 SU 파일에는 기록할 수 없는 자료 처리 과정과 사용한 변수들, 속도 분석에서 선택한 속도 등의 정보를 기록할 수 있고, 탄성파 자료도 배열도 실수로 저장하게 된다. 객체 내의 헤더와 시계열 자료는 Numpy의 구조적 배열을 사용하여 저장하므로 Numpy 명령을 사용하여 헤더 및 트레이스 편집과 같은 다양한 처리를 수행할 수 있다. 이러한 기능들 외에도 필요에 따라 특정 헤더 키 값을 얻거나 수정할 수 있고, 키 값을 이용해 윈도우를 적용하거나 트레이스들을 분리할 수 있다. 특정 헤더 키에 몇 개의 서로 다른 값이 사용되었는지, 키 값의 범위는 어떻게 되는지도 간편하게 알 수 있다.

정렬과 이득 조절

정렬 기능을 이용하면 동시에 여러 개의 SU 키를 이용하여 정렬이 가능하며 각각의 키에 대해 오름차순, 내림차순을 선택할 수 있다. 일반적인 자료 처리를 위해서는 공통 송신원별로 정렬된 자료를 속도 분석과 동적 보정을 위해 공통 중간점 모음 자료로 정렬할 수 있다. 이득 조절은 시간의 거듭 제곱 형태, 지수 함수를 사용하는 형태와 자동 이득 제어를 지원한다. 이득 조절 기능은 시간에 따라 탄성파 신호의 진폭이 작아지는 현상을 보정하기 위해 사용할 수 있다.

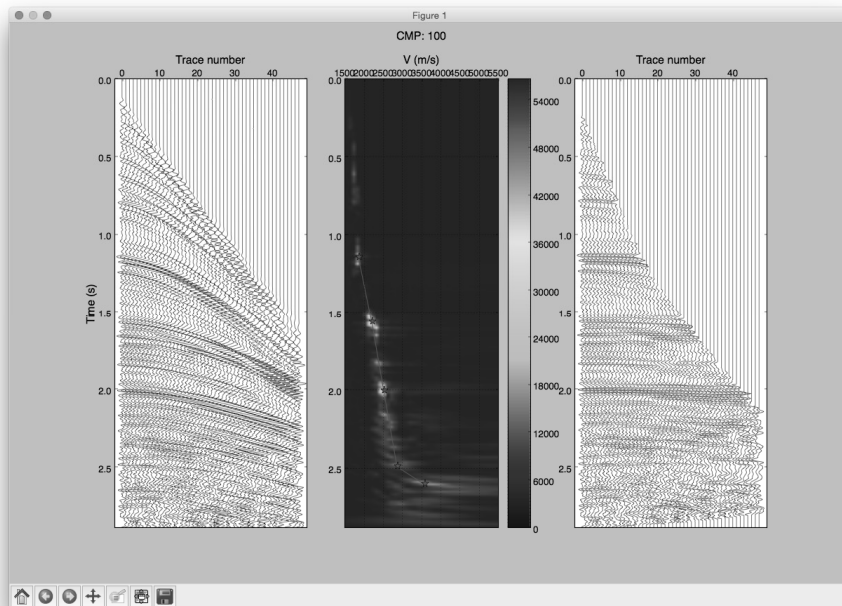


Fig. 4. A program for velocity analysis. The left panel shows an original common midpoint(CMP) gather. The middle panel is the semblance panel and users pick up the velocity on this panel. The right panel shows the normal moveout(NMO) corrected CMP gather using the picked velocities.

주파수 필터링과 곱풀기

본 패키지에서는 버터워스 필터(Butterworth, 1930)를 이용한 주파수 띠통과 필터 기능을 지원한다. 필터링 기능은 저주파 잡음을 제거하거나 원하는 주파수 대역의 신호를 얻기 위해 사용할 수 있다. 송곳 곱풀기 방식을 지원하며, 송곳 곱풀기는 자료에 따라 계산 시간이 오래 걸릴 수 있기 때문에 Cython 모듈을 이용하여 가속하였다. 예제에 사용한 Marmousi 자료의 경우 동일한 컴퓨터에서 파이썬 코드를 사용할 경우 7분 50초 이상 걸리던 작업이 Cython 코드를 통해 4초 이내로 줄어 100배가 넘는 속도 향상을 얻을 수 있었다.

속도 분석과 동적 보정

Matplotlib 모듈을 이용하여 속도 분석 프로그램을 작성하였다(Fig. 4). 본 패키지는 교육을 위한 패키지이므로 자동 속도 분석을 사용하지 않고, 속도 분석을 위해 별도 창에 띄운 답음 그림 상에서 마우스로 속도와 시간 좌표를 선택하도록 만들었다. 따라서 속도 분석은 웹 브라우저 상에서 하지 않고 별도의 명령행 파이썬 프로그램을 통해 수행한다. 선택한 속도는 간편하게 선형으로 보간하며, 인접 공통 중간점 모음의 속도들도 선형으로 보간하였다. 동적 보정 시 신호가 일정 비율 이상으로 늘어나는 부분은 제거하도록 하였다. 속도 분석 화면 가운데에는 답음 계산 그림, 왼쪽에는 동적 보정 전의 공통 중간점 모음 자료, 오른쪽에는 동적 보정 후의 공통 중간점 모음 자료를 제시하여 학생들이 동적 보정 결과를 참고하여 속도를 선택하고 수정할 수 있도록 하였다(Fig. 4). 속도 분석과 동적 보정 과정 또한 Cython을 이용하여 가속하였다.

겹쌓기, 정적 보정 및 참반사 보정

동적 보정을 통해 영 별림거리 신호로 보정한 자료들은 겹쌓기를 통해 신호대 잡음비를 높이게 된다. Fig. 5에 예시 자료의 겹쌓기 결과를 보였다. Fig. 6는 Fig. 5의 단면에 겹쌓기후 참반사 보정을 적용한 결과이다. 참반사 보정은 Stolt의 방법(Stolt, 1978)을 적용하였고, 3000 m/s의 상속도를 사용하였다. 겹쌓기 결과에 비해 참반사 보정 결과가 향상된 것을 볼 수 있다. 그러나 경사면이 많은 구조에서 경사 시간차 보정 없이 상속도 모델로 얻은 결과라 영상에서 단층 구조를 정확히 파악하기는 힘들다. 실제 속도 모델을 이용하여 겹쌓기 전 키르히호프 참반사 보정(Yilmaz, 2001)을 적용하면 향상된 결과를 얻을 수 있다(Fig. 7). Fig. 6의 단면을 얻기 위해 적용한 자료 처리는 시간 이득, 송곳 곱풀기, 자동 이득 제어, 정렬, 속도 분석, 동적 보정, 겹쌓기, 참반사 보정이다. 현장 육상 자료의 경우 필요에 따라 처리 과정 중에 주파수 필터링과 간단한 정적 보정(Hileman *et al.*, 1968; Taner *et al.*, 1974; Mousa and Al-Shuhail, 2011)을 적용할 수 있다.

실제 강의에서의 적용 사례

실제 대학 강의에서는 파이썬 언어를 모르는 학생들을 위해 전산실에서 3회에 걸쳐 파이썬, IPython Notebook, Numpy와 Matplotlib 사용법 강의를 진행한 후 자료 처리 실습을 진행하였다. 강의는 주2회 강의실과 전산실에서 번갈아가며 진행하였다. 강의실에서 기본적인 신호 처리, 단계별 탄성과 자료 처리 원리와 방법에 대해 설명한 후 전산실에서 IPython Notebook을 이용해 해당 단계의 실습을 진행하였다. 앞으로도 강의와 실습을 통해 본 패키지의 기능을 보완, 확장 및 개선할 것이다. 본 패키지 파일, 해상 및 육

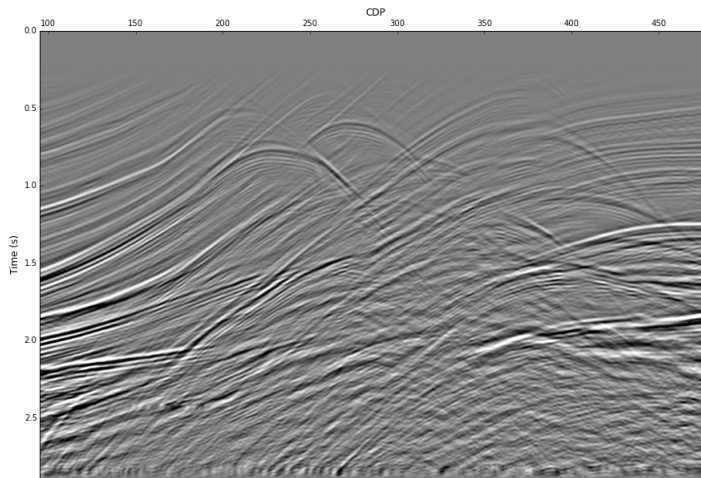


Fig. 5. A stacked section.

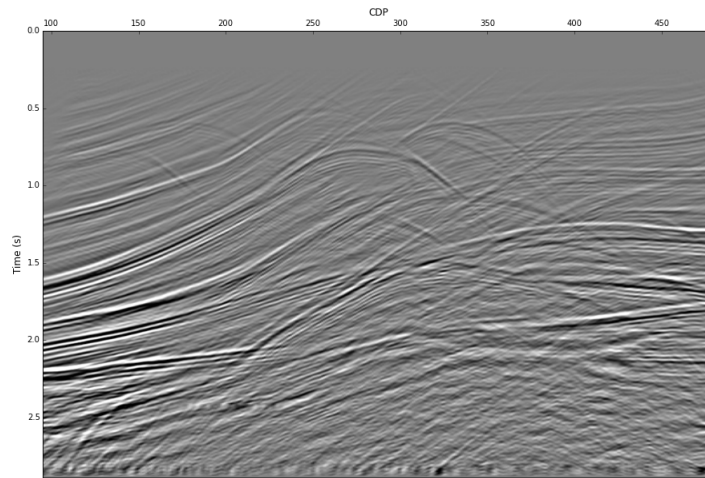


Fig. 6. A migrated section obtained using the post-stack Stolt migration algorithm(Stolt, 1978).

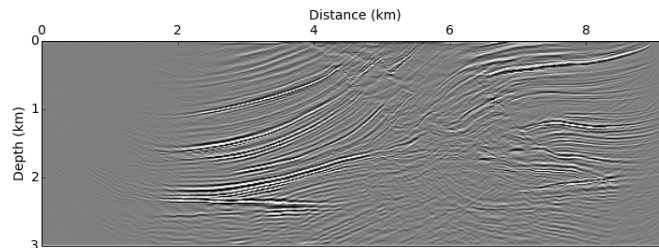


Fig. 7. A migrated section obtained using the prestack Kirchhoff migration algorithm(Yilmaz, 2001).

상 자료 처리 예시와 각각의 기능에 대한 설명은(<https://github.com/pkgpl/IPythonProcessing>)에서 찾을 수 있다.

결 론

본 연구에서는 대학 학부에서의 반사법 탄성파 자료 처리 실습을 위한 파이썬 패키지를 개발하였다. 본 패키지는 상용 자료 처리 프로그램이나 리눅스용 전문 오픈소스 프로그램들에 비해 기능은 제한적이지만 기본적인 자료 처리 흐름에 필요한 기능들을 갖추고 있고, 복잡한 설정 없이 쉽게 사용할 수 있다. 학생들은 자료 처리 패키지와 IPython Notebook을 이용해 2차원 탐사 자료 처리를 실습해볼 수 있다. 탄성파 자료 처리 수업에서 자료 처리 이론 강의와 본 패키지를 이용한 실습을 병행할 때 학생들의 자료 처리 이해도가 향상될 것으로 기대된다. 본 패키지는 오픈 소스이므로 원할 경우 사용자가 기능을 추가해서 사용할 수 있다. 관련 연구자들은 자료 처리 알고리즘 시제품 개발에도 활용할 수 있을 것이다.

사 사

이 논문은 부경대학교 자율창의학술연구비(2014년)에 의하여 연구되었음.

References

- Anaconda, 2015.6.16., <https://store.continuum.io/cshop/anaconda>.
- Beazley, D.M., 2009, Python essential reference, 4th ed., Developer's Library, pp. 5-24.
- Butterworth, S., 1930, "On the theory of filter amplifiers", *Experimental Wireless and the Wireless Engineer*, Vol. 7, pp. 536-541.
- Canopy, 2015.6.16., <https://www.enthought.com/products/canopy>.
- CGG, 2015.7.16., <http://www.cgg.com/default.aspx?cid=4-5820-1925>.
- Clapp, R.G., Prucha, M.L., Sava, P., Dellinger, J. and Biondi, B., 2004, SEP Manual, *Stanford Exploration Project*, p. 7.
- Globe Claritas, 2015.6.16., <http://www.globeclaritas.com>.

- Ha, W., 2014, "Productive small-scale data processing using Python", *J. Korean Soc. Miner. Energy Resour. Eng.*, Vol. 51, No. 5, pp. 705-714.
- Hileman, J.A., Embree, P. and Pflueger, J.C., 1968, "Automated static corrections", *Geophysical Prospecting*, Vol. 16, No. 3, pp. 326-358.
- IPython, 2015.6.16., <http://ipython.org/notebook.html>.
- Ji, J. and Choi Y.G., 2010, "3D seismic data processing methodology using public domain software system", *Jigu-Mulli-wa-Mulli-Tamsa*, Vol. 13, No. 2, pp. 159-168.
- Jupyter, 2015.6.16., <http://jupyter.org>.
- Landmark software, 2015.6.16., <https://www.landmarksoftware.com/Pages/SeisSpaceProMAX.aspx>.
- Latex, 2015.6.16., <http://www.latex-project.org>.
- Madagascar, 2015.6.16., <http://www.ahay.org>.
- Markdown, 2015.6.16., <http://daringfireball.net/projects/markdown>.
- Matlab, 2015.6.16., <http://www.mathworks.co.kr/products/matlab>.
- Mousa, W.A. and Al-Shuhail, A.A., 2011, *Processing of seismic reflection data using MATLAB*, Morgan & Claypool Publishers, pp. 1-71.
- Paradigm, 2015.6.16., <http://www.pdgm.com/solutions/seismic-processing-and-imaging>.
- Sage, 2015.6.16., <https://cloud.sagemath.com>.
- Schlumberger software, 2015.6.16., <http://www.software.slb.com/products/platform/Pages/omega.aspx>.
- SeismicLab, 2015.6.16., <http://seismic-lab.physics.ualberta.ca>.
- Sheen, D.-H., Ji, J. and Lee, D.-S., 2003, "Development of a PC-based 3-D seismic data processing system for the underground investigation", *Journal of the Korean Society for Geosystem Engineering*, Vol. 40, No. 1, pp. 48-57.
- Stockwell, J.W., 1999, "The CWP/SU: Seismic Un*x package", *Computers & Geosciences*, Vol. 25, pp. 415-419.
- Stolt, R.H., 1978, "Migration by Fourier transform", *Geophysics*, Vol. 43, 23-48.
- Taner, M.T., Koehler, F. and Alhilali, K.A., 1974, "Estimation and correction of near-surface time anomalies", *Geophysics*, Vol. 39, No. 4, pp. 441-463.
- Versteeg, R., 1994, "The Marmousi experience: velocity model determination on a synthetic complex data set", *Leading Edge*, Vol. 13, pp. 927-936.
- Wakari, 2015.6.16., <https://www.wakari.io>.
- Yilmaz, O., 2001, *Seismic data analysis, 2nd ed.*, Society of Exploration Geophysicists, p. 788.

하 완 수

현재 부경대학교 에너지자원공학과 조교수
(本學會誌 第52卷 第1号 参照)
