

다수/다차원 격자형데이터를 이용한 해상도 변환의 효율적 방안 연구

The Effective Method for Changing the Resolution of the Grid Environment Data

김 창 진* 오 광 백* 나 영 남**
Chang-Jin Kim Gwang-Beak Oh Young-Nam Na

ABSTRACT

In counterfire warfare, it is important to detect and attack enemy targets faster than the enemy using sensing. The grided environmental data is usually provided by the numerical simulation coupled with a data assimilation technique and various inter- or extrapolation algorithms, both of which are based on the observation spanning from simple equipments to satellites. In order to employ the gridded environmental data in the M&S system frequently cutting area and changing its resolution, interpolation algorithms such as linear, cubic spline, IDW, and Kriging methods are necessary to apply. These methods, however, require much time in the M&S system. This paper introduces a technic to reduce time to change the resolution of data. using the binary search method, which finds a point to interpolate quickly and interpolate data in the vicinity of. We also show the efficiency of proposed methods by way of measuring the respective elapsed times.

Keywords : NetCDF, Interpolation

1. 서론

실 환경데이터(해양, 기상)는 선진국의 고급정보였으나, 정보화시대를 맞아 실 측정데이터를 인터넷 상에

서 개방하는 추세이다. 국방기술도 무기체계 M&S 모델링의 결과에 대한 신뢰성을 향상시키기 위해 실 측정데이터에 대한 활용도가 높아져가고 있는 추세이다. 이렇게 공개된 측정데이터를 무기체계에 활용하기 위해서는 작전구역 상황에 맞는 해상도 변환이 필수적이다. 그러나 대부분의 데이터가 격자형의 대용량 데이터로서 해상도 변환의 시간이 오래 걸리게 된다. 본 논문에서는 격자형 데이터의 해상도변환 시 효율적인 방안을 제안한다.

† 2013년 1월 24일 접수~2013년 3월 15일 게재승인

* 삼성탈레스 해양시스템연구소 기반체계그룹

** 국방과학연구소(ADD)

책임저자 : 김창진(cj81@samsung.com)

2. 격자형 환경데이터

NOAA등에서 인터넷 상에 공개적으로 제공되는 환경데이터는 netCDF(network Common Data Format), Grib, HDF(Hierarchical Data Format)등 경도, 위도, 깊이, 시간 정보를 축으로 가지는 격자형의 대용량데이터이다. 본 논문에서는 이중 가장 많이 사용되는 netCDF 형식의 환경데이터를 사용한다. netCDF는 array 기반의 과학 데이터를 저장하는 데이터 포맷이다^[1].

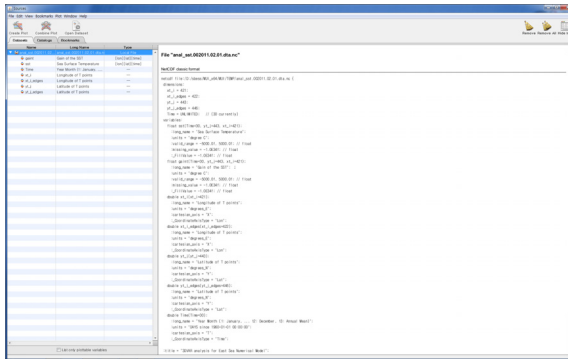


Fig. 1. A example of netCDF format

3. 해상도 변환 알고리즘 소개

본 논문에서 해상도 변환을 위해 사용하는 알고리즘은 다음과 같다.

가. Multi-Linear 보간법

Rick Wagner가 소개한 multi-Linear Interpolation 방법을 사용하여 가장 가까운 점을 통해 선형 함수를 통해 해를 찾는다. 다음은 n차원 데이터에서 주변의 입력치를 이용하여 해를 구하는 방정식을 보여준다^[2].

$$y = x_1 \cdot N_1 + x_2 \cdot N_2 + \dots + x_n \cdot N_n \quad (1)$$

단, N_n 은 다음과 같다.

$$N_n = \frac{(x_{11} - x_{12}) \cdot (x_{21} - x_{22}) \dots \cdot (x_{n1} - x_{n2})}{(x_{11} - x_{10}) \cdot (x_{21} - x_{20}) \dots \cdot (x_{n1} - x_{n0})}$$

나. Cubic Spline 보간법

주변의 값과 자연스럽게 보간하기 위해 3차 다항식

을 이용하여 계산하는 방법이다. 이를 위해 두 곡선이 미분 값과 이차미분 값이 같음을 이용하여 보간값을 구하는 방식이다^[3].

다. IDW 보간법(Inverse Distance Weighted Interpolation)

보간지점에 가까운 입력치가 큰 가중치를 가지도록 하는 보간법이다. weight function은 다음과 같다.

$$w_i = \frac{\left[\frac{R - h_i}{Rh_i}\right]^2}{\sum_{j=1}^n \left[\frac{R - h_j}{Rh_j}\right]^2} \quad (2)$$

단, R은 보간지점과 가장 거리가 먼 포인트와의 거리이고 h_i 는 보간지점과 각 포인트와의 거리를 나타낸다.

라. Kriging 보간법

Kriging은 보간지점 주위의 데이터의 공간적 공분산의 weight function에 의한 보간값을 결정하는 기법이다^[4].

Multi-Linear와 Cubic Spline은 산술적인 보간 알고리즘이고 IDW/Kriging은 위치 기반의 보간 알고리즘이다.

4. 범위 제한 윈도우와 binary Search를 이용한 방안

netCDF는 격자형 데이터를 위한 포맷으로 dimension을 공유하는 데이터로 이루어져 있다. 데이터의 축은 dimension으로 저장되고, 대부분 경도, 위도, 고도, 시간 축으로 2차, 3차, 4차등의 배열 구조로 되어 있다. 해당 Variable의 데이터는 차원이 증가됨에 따라 기하급수적으로 커지게 된다. 예를 들어 경도, 위도 100×100 의 데이터가 깊이로 100, 시간 축으로 10이 증가하게 되면 variable은 총 10,000,000개라는 데이터가 되고 이를 경도 축, 위도 축으로 2배씩만 해상도 조절해도 기존 데이터 4배에 해당하는 데이터가 결과로 발생하게 된다. 이를 해상도 변환 시 전체 Time Complexity는 다음 식과 같다.

$$\text{Time Complexity} = 40,000,000 \times T \quad (3)$$

단, T는 한 Point 해상도 변환 시 소요시간이다

본 논문에서는 이러한 대용량 데이터 해상도 처리 시 소요시간 단축을 위해 제한된 입력치 구성을 위한 범위 제한 윈도우 방안과 보간 지점에 가까운 입력치를 찾기 위한 Binary Search방안을 제안한다.

가. 범위 제한 윈도우 적용

시간이 걸리는 주된 원인은 많은 데이터의 양에 있다. 임의의 보간 지점 한 점을 추정하는데 전체의 데이터를 이용하여 계산하게 되면 입력치 데이터가 증가함에 따라 소요시간은 기하급수적으로 증가하게 된다. IDW와 Kriging의 알고리즘은 위치기반의 알고리즘으로서 랜덤한 위치의 관측된 소수의 관측치를 가지고 임의의 점의 값을 추정하는 방식이다. 이러한 특성상 알고리즘 그대로 격자형 대용량 데이터에 적용하게 되면 모든 점에 대한 거리를 계산하므로 많은 시간이 소요된다.

그러나 IDW나 Kriging의 알고리즘은 가까운 지점의 값이 더욱 큰 가중치를 가지므로 이러한 특성을 고려하여 가까운 점들로 입력치를 제한하여 수행하게 되면 수행시간은 단축되나, 모든 입력치를 고려하여 계산하는 방법과 유사한 결과를 얻을 수 있을 것이다. 다음의 그림은 크기가 4인 윈도우를 적용하여 보간지점 상하좌우 2개의 입력치를 이용하여 해상도 조절처리 하는 과정을 보여준다.

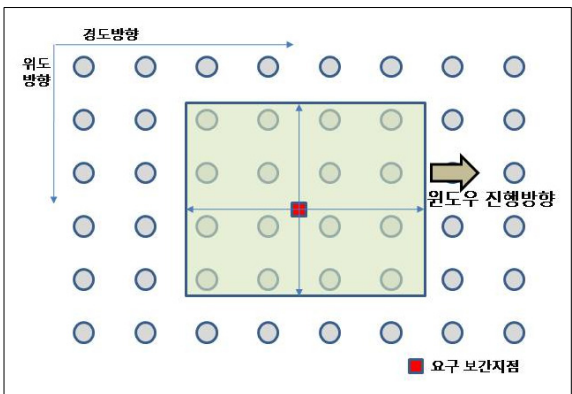


Fig. 2. A example to use the range-limited window

나. Binary Search를 통한 위치 탐색

범위 제한 윈도우를 적용하기 위해서는 가까운 입력치들을 가져올 수 있어야 한다. 임의의 점에서 근접한 입력치를 찾기 위해 모든 입력치에 대해 거리

를 계산하거나 순차적으로 해당 위치까지 검색해야 한다. 격자형데이터의 위치는 경도, 위도, 심도, 시간 축이고 이들은 오름차순 혹은 내림차순의 순서로 저장되어 있다. 이러한 데이터의 특성을 이용하여 binary Search를 통해 근접점을 찾게 되면 소요시간을 줄일 수 있다. binary Search는 오름차순 내림차순의 sorting된 데이터에서 임의의 데이터가 존재하는 위치를 빠르게 찾는 방법이다. 오름차순의 데이터일 경우 중간 지점의 데이터를 확인하여 비교 데이터 보다 클 경우 나머지 우측의 데이터만 비교하고, 작을 경우 나머지 좌측의 데이터만 비교함으로써 비교 데이터의 횟수를 획기적으로 줄이는 방법이다. sequential의 경우 N개의 데이터를 비교할 때 최대 N번 비교해야 하나, binary의 경우 모든 경우에 대해 log n번으로 비교횟수가 수렴된다. 특히 임의의 지점에 대한 추정으로 보간 알고리즘 사용 시 sequential 방안은 임의의 점 위치에 대해 최소 1번에서 최대 N번 검색으로 검색횟수가 불규칙하나, binary 방안은 위치에 상관없이 항상 log n의 검색 횟수를 가지는 장점을 가지게 된다. 아래의 그림은 35라는 값이 오른쪽 배열에 위치하기 위해 데이터 비교횟수를 보여주는 예시이다. Sequential search방안은 총 9회 비교해야하나, binary search의 경우 3회 비교를 통해 위치를 찾게 된다.

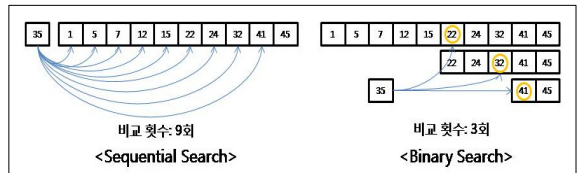


Fig. 3. A example of the difference between sequential Search and binary Search

다음의 표는 보간 알고리즘 별 범위 제한 윈도우 방안과 binary search 방안의 적용여부를 보여준다.

Table 1. The checklist of method per the algorithms

	범위제한 윈도우	binary search
Linear	X	O
Spline	O	O
IDW	O	O
Kriging	O	O

5. 시험결과

시험환경은 HP Z600 모델의 workstation급 PC(CPU : Intel Xeon E5507 2.27Ghz, 2.26Ghz, RAM : 6GB)를 사용했으며, 운영체제는 window 7(64bit)이다.

시험자료 구성은 해수면 수온자료 netCDF파일로서 위도방향 20도에서 25도, 경도 방향 140도에서 145도 자료를 추출하여 총 60×60×30(위도 데이터개수×경도 데이터개수×시간단편개수) 포함 108000개의 자료가 입력치로 구성된다.

출력치는 위도 경도 방향으로 각각 2배 해상도를 가지도록 해상도 처리하였으며, 이 경우 총 출력치 데이터 개수는 119×119×30(위도 데이터개수×경도 데이터개수×시간단편 개수)으로 총 424,830개가 결과로 출력된다.

향후 언급되는 윈도우 사이즈는 근접 입력치의 개수를 의미하는 것으로 각각 방향별 개수이다. 예를 들어 윈도우 사이즈 5는 상하좌우 각각방향으로 5개의 입력치를 사용함을 의미한다.

가. Linear 알고리즘 결과

Linear 알고리즘은 검색 방법만 적용되어 수행되므로 시험환경의 입력치 만으로는 큰 차이를 보이지 않았다. 시험환경 입력치보다 4배 입력치를 증가하여 수행하면 다음과 같은 수행시간의 차이를 보이게 된다.

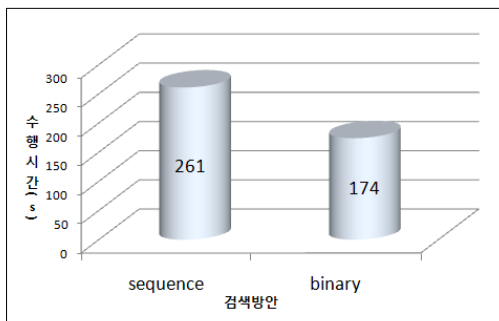


Fig. 4. Time costs per searching methods in linear interpolation

binary 검색방안이 30%정도의 시간단축을 가져옴을 보여준다. 검색방법에 대한 결과 차이는 없으므로 생략한다.

나. Spline 알고리즘 결과

다음은 spline 알고리즘을 각각 윈도우 사이즈 5, 10,

15 경우와 윈도우 미적용 시 소요시간을 비교한 그래프이다.

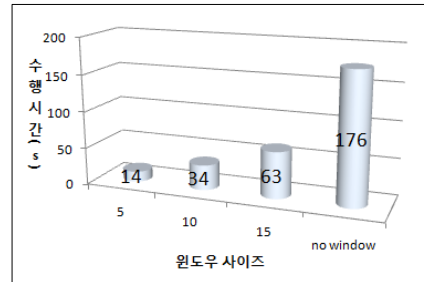


Fig. 5. Time costs per window sizes in spline interpolation

윈도우 사이즈 5일 때와 윈도우 미적용 시를 비교하면 10배 이상의 시간 단축을 보여준다. 반면 아래의 그림과 같이 출력결과는 거의 차이가 없음을 보여준다.

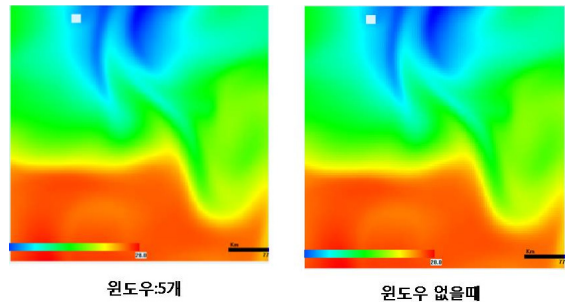


Fig. 6. The comparison of results in spline interpolation

다. IDW 알고리즘 결과

다음은 IDW 알고리즘을 각각 윈도우 사이즈 5, 10, 15때와 윈도우 미적용 시의 소요시간을 비교한 그래프이다.

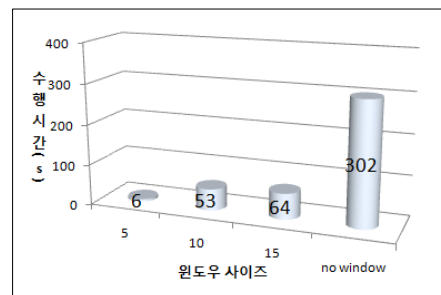


Fig. 7. Time costs per window sizes in IDW interpolation

윈도우 사이즈가 10에서 15로 늘었을 때의 소요시간의 차이는 크지 않으나, 사이즈 5일 때와 없을 때와 비교하면 거의 50배가량의 효율을 보여준다. 다음의 그림은 윈도우사이즈 5일 때와 윈도우 미적용 시 출력 결과를 비교한 것이다.

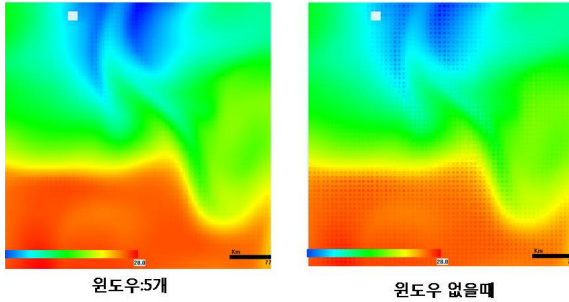


Fig. 8. The comparison of results in IDW interpolation

윈도우 미적용 시 입력치 부분이 더욱 강하게 나타나는 현상이 보여지나, 해상도 변환의 흐름에는 큰 차이를 보여주지는 않는다.

라. Kriging 알고리즘 결과

다음은 Kriging 알고리즘을 각각 윈도우 사이즈 5, 10, 15일 때와 윈도우 미적용 시의 소요시간을 비교한 그래프이다.

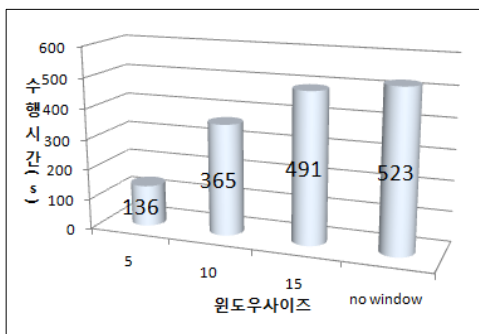


Fig. 9. Time costs per window sizes in Kriging interpolation

윈도우 사이즈 5일 때와 윈도우 미적용 시 소요시간 차이는 약 5배 정도로 나타난다. 윈도우 사이즈 15와 윈도우 미적용 시 차이가 크지 않음은 Kriging 내부에 입력치를 제한하는 부분이 존재하여 일정 이상의 입력치 데이터를 사용하지 않음으로 인해 발생하는

것으로 예상된다. 다음의 그림은 윈도우 사이즈 5일 경우와 윈도우 미적용 시의 출력 결과를 보여주는데 출력결과도 거의 차이가 없음을 알 수 있다.

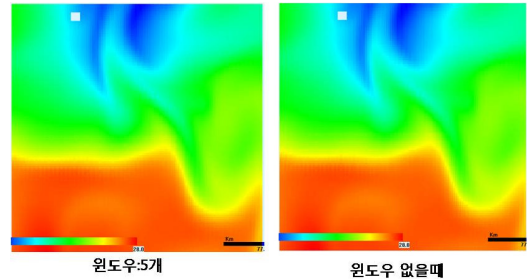


Fig. 10. The comparison of results in Kriging interpolation

6. 결론

환경데이터를 해상도 변환 시 데이터의 양으로 인해 많은 시간이 소요된다. 본 논문이 제안하는 binary search를 통해 보간 지점을 빠르게 찾고, 범위 제한 윈도우를 이용하여 입력치 제한함으로써 해상도 변환 소요시간을 단축하는 효과를 가져왔다.

제한적인 입력데이터를 사용하여 해상도 변환하는 것은 전체 입력데이터를 사용한 것과 다른 결과를 가져올 수 있다. 그러나 임의의 점을 보간 하는 것도 여러 입력 값을 고려하여 계산한 추정치이고, 제한적인 입력데이터가 가까운 거리의 데이터라면 전체 입력데이터를 사용한 것과 큰 차이를 보이지 않았다. 본 논문이 제안하는 방안을 이용하여 알고리즘의 수행시간을 줄이는 것이 알고리즘을 활용하는 데 있어 유용할 것으로 보인다.

후 기

본 논문은 국방과학연구소의 합성전장모의체계 사업의 일환으로 본문의 IDW, Kriging 알고리즘은 국방과학연구소에서 제공받은 알고리즘을 사용하였습니다.

References

[1] UCAR Community <http://www.unidata.ucar.edu/>

- [2] Rick Wagner, Multi-Linear Interpolation, Beach Cities Robotics FIRST Team 294.
- [3] William H. Press, Saul A. Teukolsky, William T. Vetterling, Brian P. Flannery, Numerical Recipes in C, CAMBRIDGE UNIVERSITY PRESS.
- [4] Geoff Bohling, Kriging, Kansas Geological Survey.