

슈퍼컴퓨터 기반 한국형 기상 수치 모델 개발

(재)한국형수치예보모델개발사업단
김정한



목 차

1. 사업단(KIAPS) 소개

1. 설립 목적
2. 수치예보시스템의 구성
3. 고성능 컴퓨팅 자원
4. 수치예보시스템 개발

2. 고성능 수치예보시스템 개발

1. 모델링 프레임워크 개발
2. 모델 격자의 병렬화
3. 고성능 입출력 라이브러리(KIO) 개발
4. KIM 모델의 병렬 확장성
5. KIM 모델의 예보 성능

3. 요약

• 재단법인 한국형수치예보모델개발사업단

- 일기예보 역량: 관측 자료(32%), **모델 성능(40%)**, 예보관(28%)
- 현재 예보 모델: 영국 UM(Unified Model), 약 년 2억
- 목표: 차세대 전지구예보모델 개발 (세계 5위 수준)
- 2010년: 한국 수치예보모델 개발을 위한 상세 기획 연구 수행, 예비타당성 조사 심사 통과
- 2011년: (재) 한국형수치예보모델개발사업단 설립
- 예산: 약 년100억, 모델명: KIM (Korean Integrated Model)

1 단계

기반구축 및 기초기술개발

- 한국형수치예보모델 개발을 위한 사업단 설립과 연구개발 체계구축
- 역학코어, 물리과정, 자료동화 분야에서의 기초기술 개발



2011-2013

2 단계

시험예보시스템 개발

- 독자기술을 기반으로 한 핵심 모듈(역학코어, 물리과정, 자료동화) 완성
- 핵심모듈의 결합을 통한 한국형수치예보시스템 시험 버전 개발
- 시험버전의 실시간 운영 및 현업 모델과의 성능 비교 수행



2014-2016

3 단계

현업예보시스템개발

- 한국형수치예보시스템 버전 1.0 개발 및 준현업 운영
- 준현업 검증을 통한 한국형 수치예보 시스템의 안정화
- 한국형수치예보시스템의 국제화



2017-2019

• 재단법인 한국형수치예보모델개발사업단

- 일기예보
- 현재 예보
- 목표: 차세대
- 2010년: 한
- 2011년: (지
- 예산: 약 년

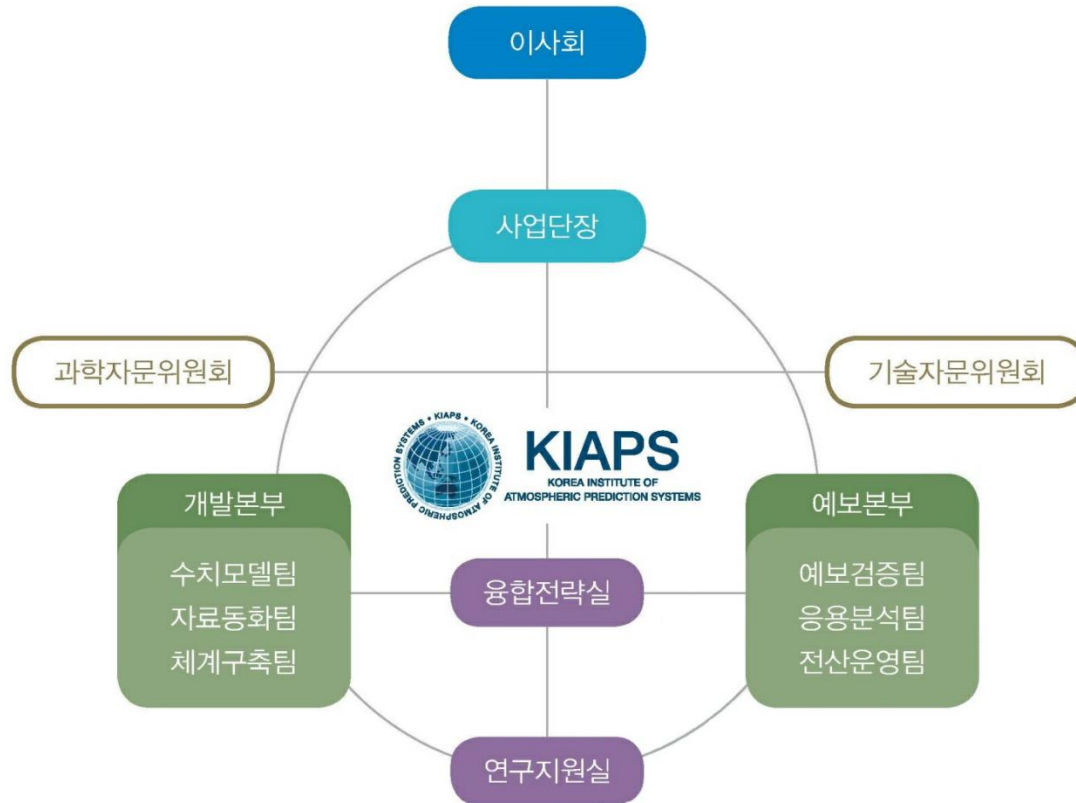
1 단계

기반구축

- 한국형수치예보사업단 설립과 연
- 역학코어, 물리분야에서의 기초기술 개발



2011-2013



보시스템개발

보시스템 버전 1.0 개발 및
통한 한국형 수치예보

- 시험버전의 실시간 운영 및 현업 모델과의 성능 비교 수행



2014-2016

- 한국형수치예보시스템의 국제화



2017-2019

수치예보시스템의 구성

역학 코어

운동 방정식: $\frac{dV}{dt} = -(2\Omega \times V) - \frac{1}{\rho} \nabla p + g + v + F_v$

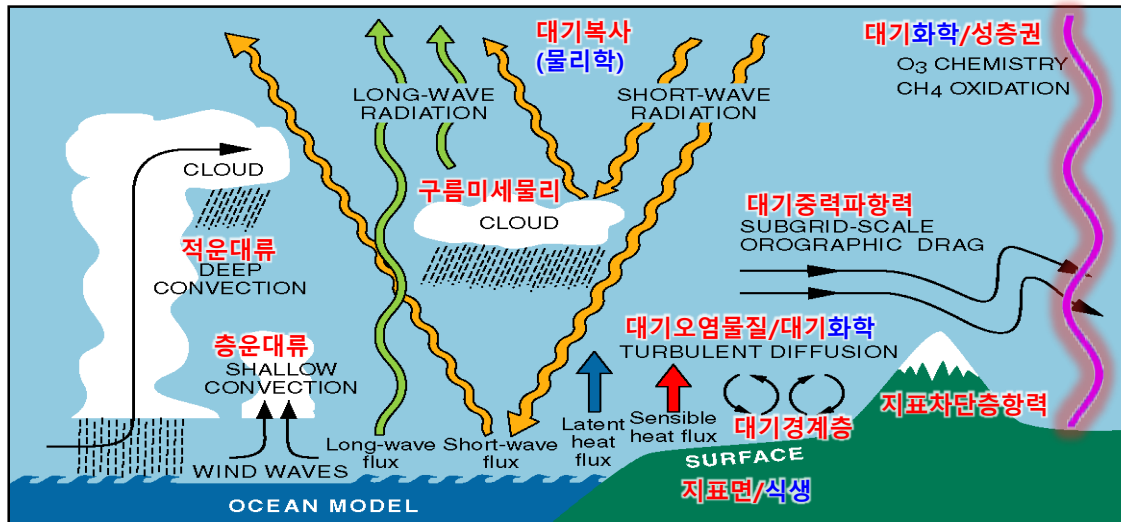
연속 방정식: $\frac{\partial \rho}{\partial t} = -\nabla \cdot (\rho V)$

열역학적 에너지 방정식: $c_p \frac{dT}{dt} = \frac{1}{\rho} \frac{d\rho}{dt} + v_T + Q$

상태 방정식: $p = \rho RT$

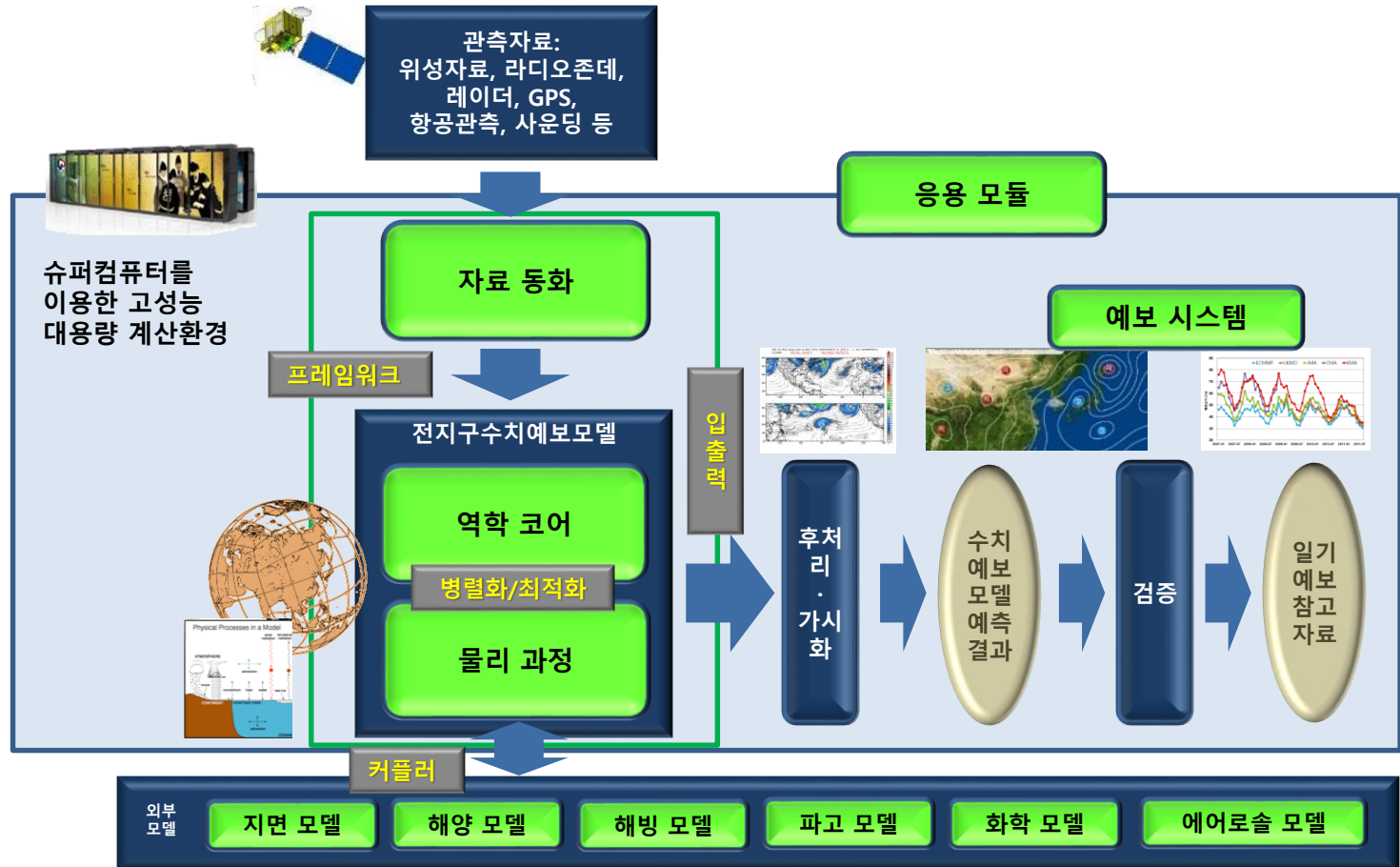
- 해석해 (analytic solution)가 존재하지 않음
⇒ 수치적인 방법으로 풀어야 함
- 비선형 방정식
⇒ 초기값의 작은 오차가 결과를 많이 바꿈
- 미분연산자
⇒ 병렬화 필요

물리 과정



수치예보시스템의 구성

• 수치예보시스템 구성



⇒ 슈퍼컴퓨터 성능(계산 자원)의 발전은 모델링 할 수 있는 정확도 혹은 규모를 결정

⇒ 관측 자료 처리, 자료 동화, 수치예보모델은 필수적으로 병렬화되어 있어야 함 (MPI 병렬화+OpenMP 병렬화)

- 연구/개발 및 준현업 테스트를 위한 컴퓨팅 자원

사업단 클러스터

이름	Gaon2
CPU	Xeon Sandy-bridge 2.9 GHz Xeon Ivy-bridge 3.0 GHz
코어 개수	2,240
이론 성능	52.6 TF
저장 장치	1 PB (IBM GPFS)
내부 네트워크	Infiniband

1. 프레임워크, 역학, 물리, 자료동화 등의 모듈 개발
2. 각 모듈들에 대한 단위 테스트 (디버깅)
3. 성능 및 속도 테스트
4. 결과 분석 및 가시화

기상청 슈퍼컴퓨터 4호기



이름	Nuri / Miri (Cray XC40)
CPU	Xeon Haswell 2.6 GHz
코어 개수	69,696 x 2
이론 성능	2.9 PF x 2
저장 장치	13 PB (Cray Lustre File system)
내부 네트워크	Aries interconnect

1. 고해상도 모델 실험
2. 계절 및 중기 실험
3. 시험 모델 혹은 준현업 모델 운영
4. 예보시스템 및 운영시스템 개발



수치예보시스템 개발 (소프트웨어)

- 기상학 관련 핵심 컴포넌트

- 역학 코어

- 격자 생성 및 미분 연산자 개발
 - 지배방정식, 점성 및 이류 기법 개발
 - 시간적분법 개발

- 물리과정, 자료동화 개발

- 병렬화 및 최적화, 고성능 컴퓨팅 기법 개발

- 모델링 프레임워크 개발

- 모델 격자의 병렬화 (격자와 미분연산자의 수치 기법)

- 고성능 입출력 시스템 개발

- 모델 커플링 기법 개발

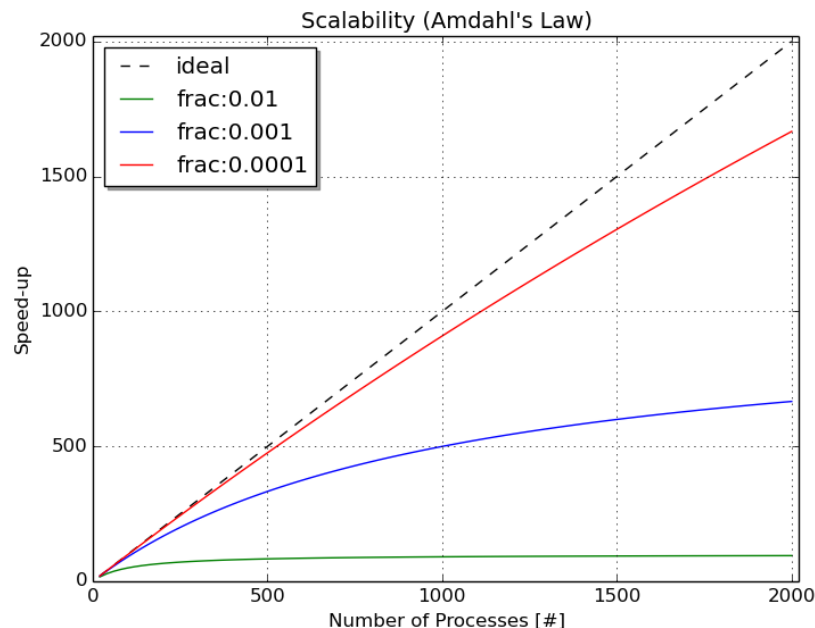
- 멀티 플랫폼 기반 예보 모델 개발

- 수치 기법 및 운영 스크립트

- 내삽 및 리맵핑 기법 개발 (1차원: 고차 보존 변환, 2차원: 보존)

- 이류 기법, 초기장 및 보조자료 생성, 모델 후처리, 자료동화 입출력, 커플링 시스템

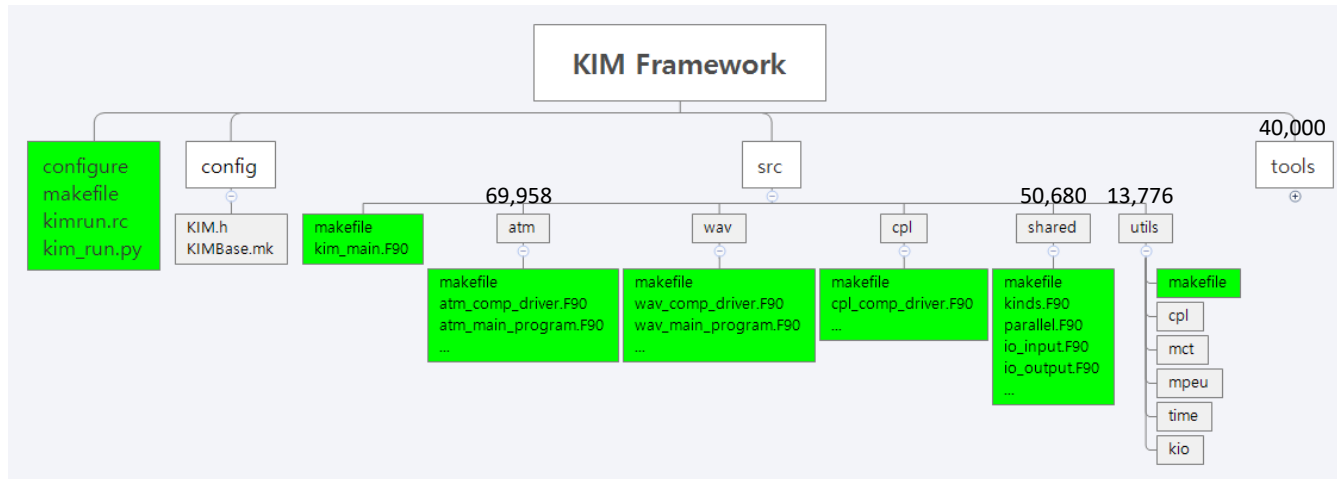
- Cylc 및 Python을 이용한 운영 소프트웨어 개발



모델링 프레임워크 소개

- 모델링 프레임워크

- 프레임워크 사전 조사 (2011년): ESMF (Earth System Modeling Framework), PRISM
- 자체 개발한 프레임워크 (2012년 ~)



- Fortran90 프로그래밍, 사업단 코딩 규칙
- make 유틸리티를 이용한 계층적 빌드시스템, python 기반 실험 설정 스크립트
- 컴포넌트 기반 모델링 프레임워크 (객체지향적 구조): 4 APIs
- utils: 달력, 입출력, 외부 모델 커플러
- shared: 병렬 환경 및 함수, 입출력 제어, 격자 생성, 미분 연산자
- atm: 역학코어, 물리과정 (기상&대기 연구자)
- svn repository 관리

모델링 프레임워크 소개

- 모델 설치 및 수행

```
> svn co svn://111.222.333.444/tags/3.1 .
```

⇒ svn repository

```
> export KIMDIR=`pwd`
```

⇒ 모델 설치 경로 설정

```
> ./configure -c intel -s nuri -p --single
```

⇒ 설치 정보 (-c: 컴파일, -s: 시스템, -p: 병렬 입출력)

```
> make install
```

⇒ 소스 코드 컴파일 및 설치

```
> vi kim_run.rc
```

⇒ 모델 실험 정보 (입력 파일, 예보 시간, 출력, ...)

```
> ./kim_run.py
```

⇒ 모델 수행 (PBS 명령어 포함)

- 1) 사용자 편의 설치 및 운영 구조
- 2) 지속적인 개선 및 사용자 (연구원 및 기상청) 의견 반영
- 3) GUI ?

모델링 프레임워크 소개

코딩 규칙 및 모듈 구조

```
#include <KIM.h>
!-----
! module atm_main_program
!-----
! abstract : main driver for KIAPS global model
!
! history log :
!   2014-11-01   junghan kim       first working version (with sang-yoon jun)
!
!-----
! use kinds      , only: i4, r4, r8
! use parallel, only: kim_par, par_initialize, par_finalize
! use time       , only: atm_time, time_set
! use dynamics, only: dyn_set=>set, dyn_ini=>ini, dyn_run=>run, dyn_fin=>fin
! use physics   , only: phy_set=>set, phy_ini=>ini, phy_run=>run, phy_fin=>fin
!
! implicit none
!
! private
!
! public :: set, ini, run, fin
!
! contains
!
!-----
!
!-----
! subroutine set(comm, res)
!-----
! implicit none
! integer(i4), optional, intent(in  ) :: comm
! integer(i4),          intent(inout) :: res
! local variables
! real(r8) :: err
!
! initialize parallel environment
! call par_initialize()
! calendar & time set
! call time_set(atm_time,gregorian,year=2018,month=5,day=16,hour=13,minute=00,&
!                                     second=00)
!
! dynamics & physics
! call dynamics_set(kim_par)
! call physics_set(dtime=dtime)
!
! return
! end subroutine set
!-----
!
!-----
!
!-----
! subroutine ini()
```

코딩 규칙: 코드 공유 & 버그 감소

- 1) Fortran90 형식: module 단위 프로그램
- 2) 소문자와 underscore
- 3) 코드 설명 및 history
- 4) 라인의 시작 행: 3
- 5) 들여쓰기 칸수: 2
- 6) 라인 길이: 80
- 7) 명시적 fortran kind 및 변수 선언
- 8) intent(in), intent(out), intent(inout) 명시
- 9) contains, subroutine, function 사이의 구분
- ...

4 API (Application Program Interface) 기반 모듈

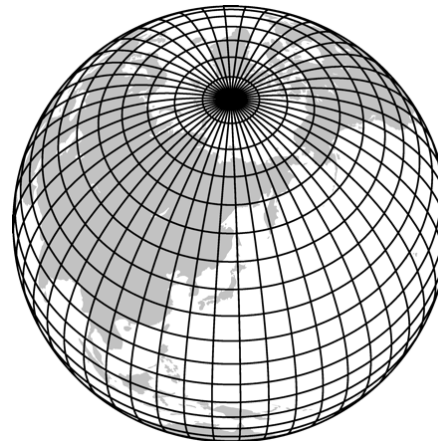
- 1) Set: 모듈 설정 (적분 간격, 함수 인자, 설정 인자)
 - 2) ini: 메모리 할당 및 변수 초기화 (allocate, var=)
 - 3) run: 메인 코드
 - 4) fin: 메모리 해제 (deallocate)
- ⇒ 하부 모듈들이 이 규칙하에서 작성이 되면 메인 함수에서 프로그램 진행 및 시작 적분 수행
- ⇒ 메모리 및 계산의 효율성 증가

모델 격자의 병렬화

• 역학 코어의 계산 격자

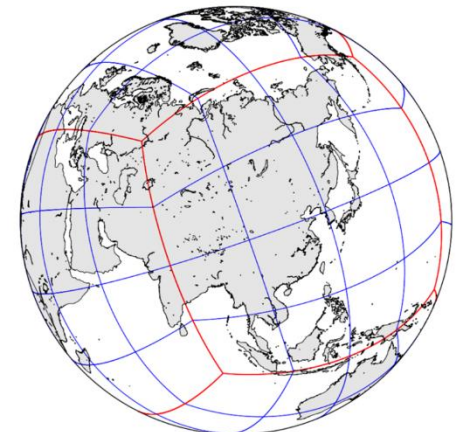
역학 코어의 수치 기법

방정식	비정역학 (Perturbation form)
격자	육면체구 격자 (Cubed-sphere grid)
수평 미분	분광 요소법 (Spectral Element Method)
연직 미분	유한 차분법 (Finite Difference Method)
시간 적분	Split-explicit RK3



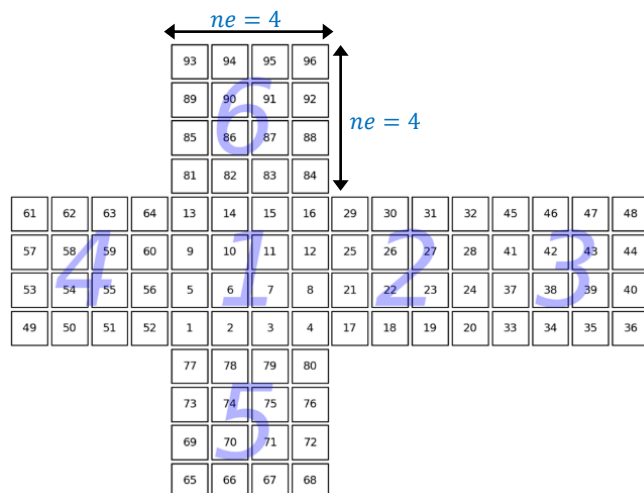
위경도 격자

- SPH (Spectral Harmonics)
- 극 특이점 문제 – 극 필터
- 고차정확도를 위해 많은 통신량이 필요



육면체구 격자

- **SEM** (Spectral Element Method), FVM, FEM
- 극 필터가 필요 없음
- **고차정확도를 위한 통신량이 적음**



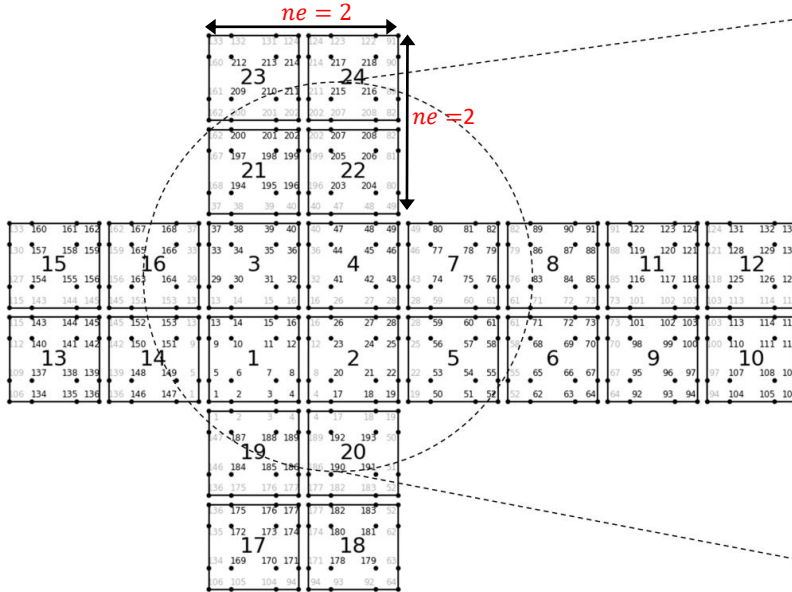
육면체구(Cubed-Sphere Grid) 격자에서 분광요소법 :

- 1) 6개의 face로 구성된 정육면체를 구면에 투영한 격자
(Equi-angular Projection)
- 2) 각 face 각 분광요소법(Spectral-Element Method)을 계산하기 위한
요소(Element)들로 구성
- 3) 전체 요소 개수: $6 \times ne \times ne$

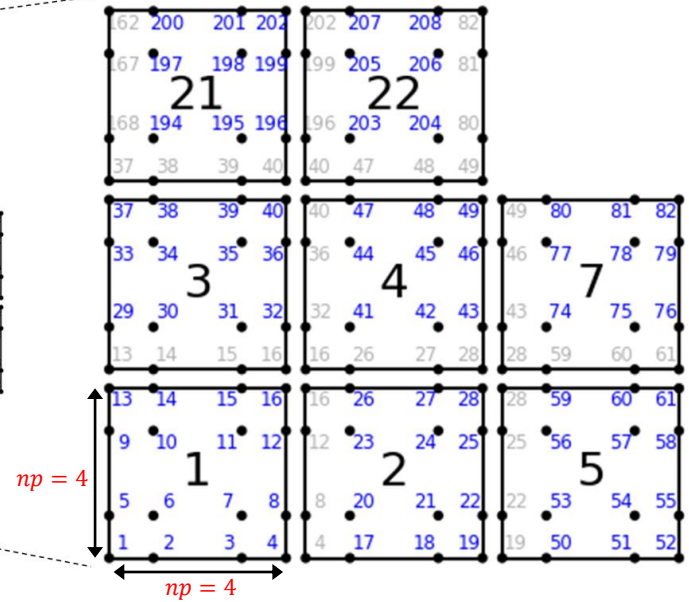
모델 격자의 병렬화

- 분광요소법에서의 GLL 점들과 변수 표현

(a)



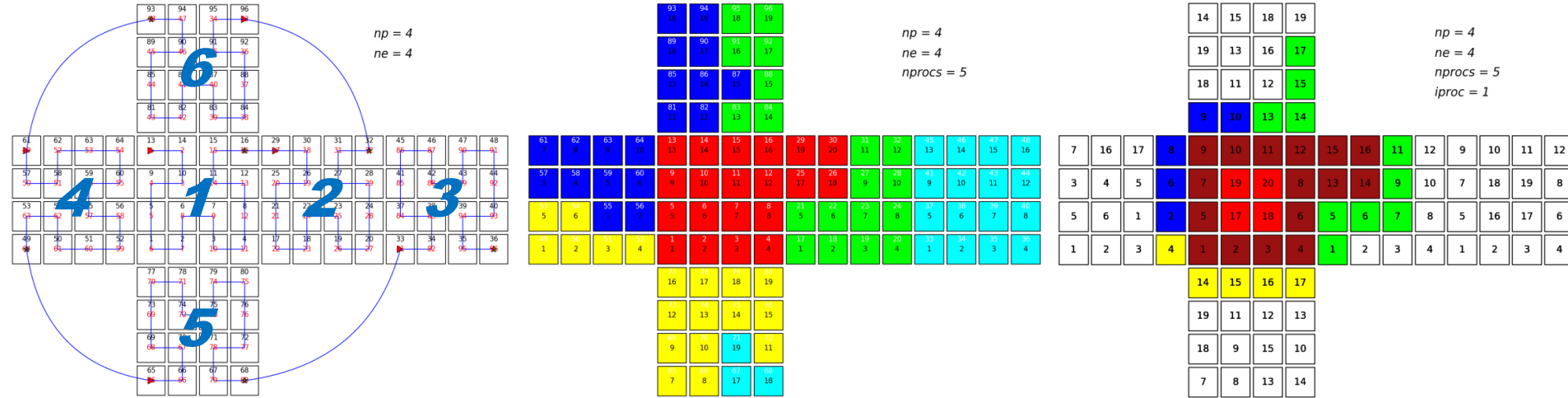
(b)



- 1) 각 요소 (element)들은 GLL 점 (Gauss-Lobatto-Legendre point)들로 구성 ($np \times np$)
- 2) 각 요소들에서 GLL 점들을 이용해 역학의 지배방정식이 계산되고 요소들의 경계는 C^0 연속성을 가짐
- 3) 전체 점들을 EP(Entire Point)라 부르고 중복된 점들이 제외된 점들을 UP(Unique Point)라고 부름
- 4) 출력 순서: face number, element number (x, y), GLL point number (x, y)

모델 격자의 병렬화

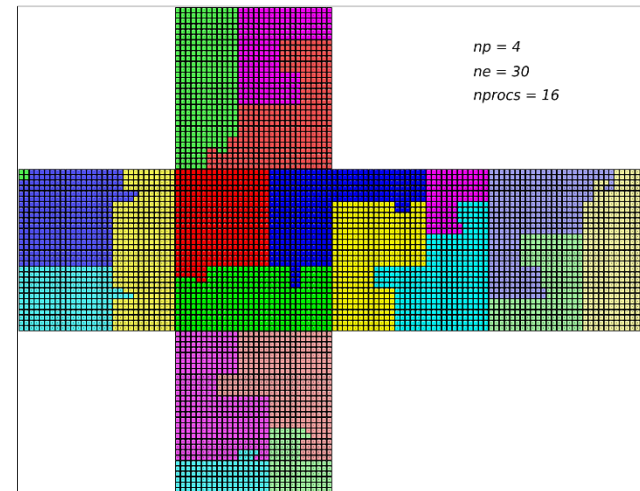
영역 분할과 병렬화



- 1) 육면체구 격자에 대한 모든 요소(element)들을 space-filling curve로 한붓 그리기
- 2) Curve의 1차원 요소들을 프로세스 개수로 도메인 분할
- 3) 프로세스별 수신해야 되는 index와 정보를 송신해야 되는 index를 table로 만들
- 4) table을 기반으로 MPI 송수신 루틴을 만들
- 5) OpenMP 병렬화: 요소들을 균등 분배

12 km 해상도: 약 3억개 격자

모델 출력 파일: 15GB x 4회/일 x 10일 x 4사이클/하루 = 2.4 TB /하루



고성능 입출력 라이브러리 개발 (KIO)

병렬 입출력 라이브러리 설계

파일 형식 (format)

- ASCII format
- Typical binary format
- General purpose I/O library format
 - HDF4, HDF5, **NetCDF3**, **NetCDF4**, GRIB1, GRIB2, ...

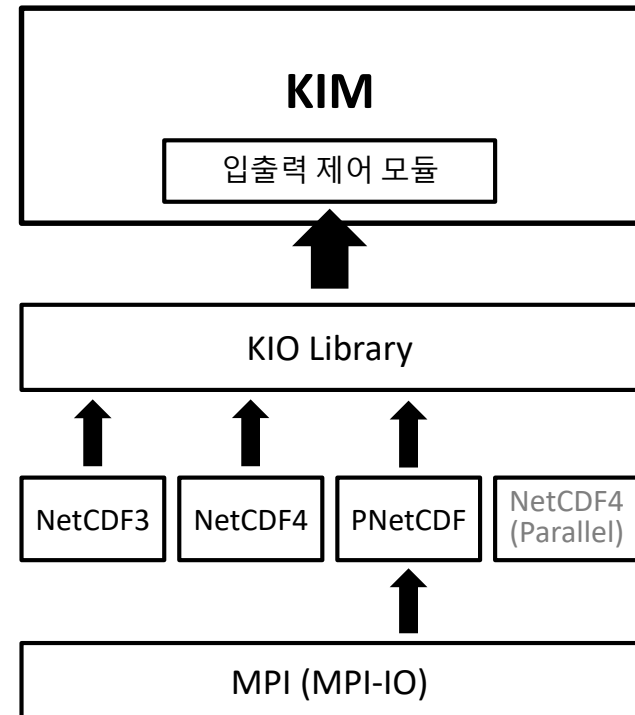
NetCDF3, NetCDF4

- 순차 및 병렬 I/O 기법을 제공
 - 순차 I/O: NetCDF3, NetCDF4 라이브러리
 - 병렬 I/O: PNetCDF, parallel NetCDF4 라이브러리
- 다른 기관과 공유가 쉬움

파일 시스템 구조:

- 모델에 직접 개발하는 것은 범용성이 떨어짐
- 라이브러리(KIO library) 형식으로 개발
- 모델에서 제어 모듈을 따로 개발
- back-end로 NetCDF 라이브러리 사용

I/O 시스템 구조

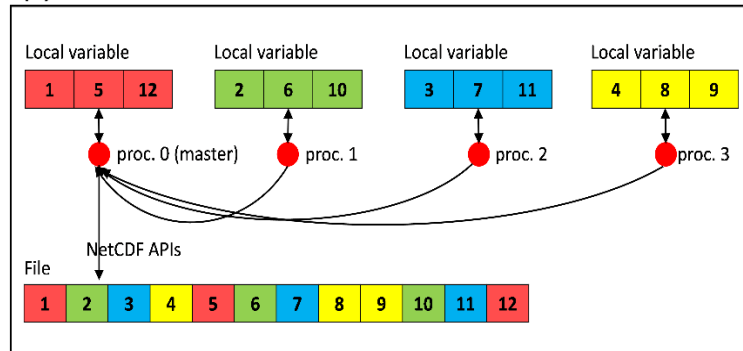


고성능 입출력 라이브러리 개발 (KIO)

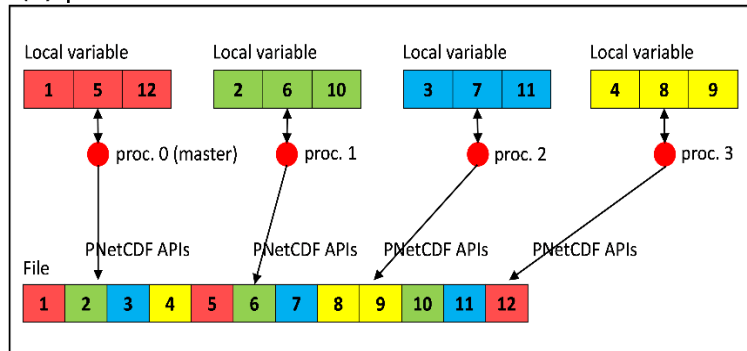
입출력 모드

- 순차 입출력, 병렬 입출력, I/O 분할

(a) serial I/O

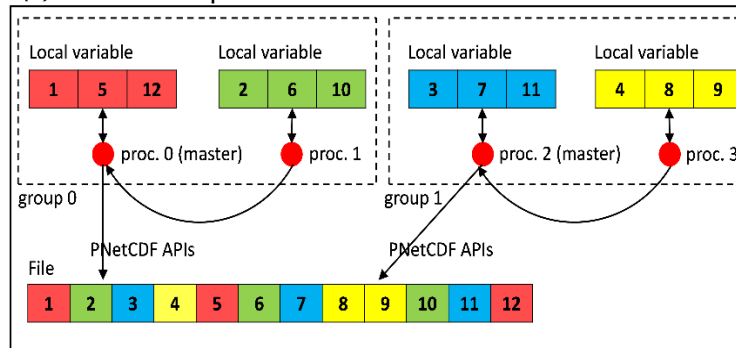


(b) parallel I/O



- 고전적인 입출력 방식
- All-to-1 통신으로 인한 부하
- 임시 버퍼가 필요

(c) I/O decomposition method

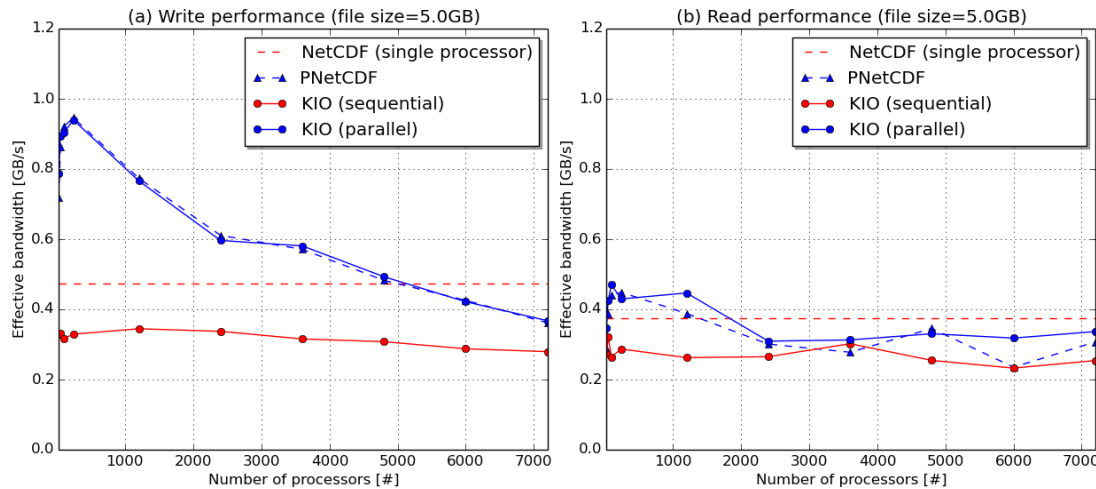


- MPI-I/O를 이용한 병렬 입출력
- 특정 프로세스 개수 이상에서
동시 파일 접근으로 인한 부하

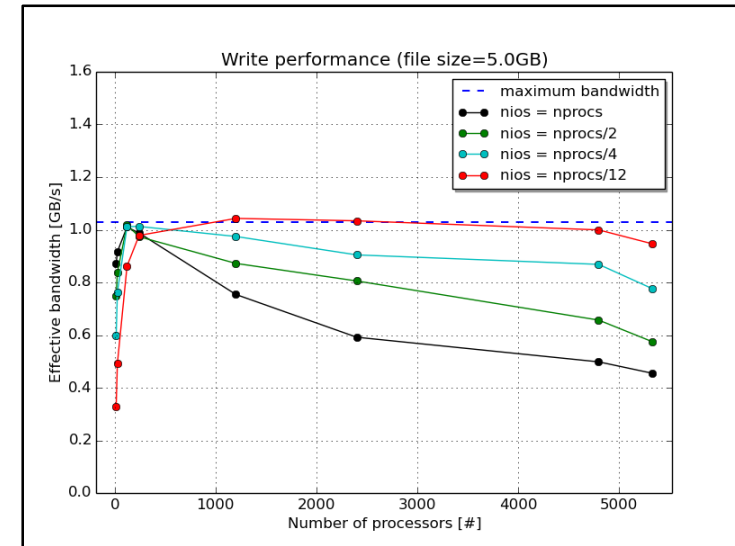
- 순차 입출력과 병렬 입출력의 하이브리드(hybrid) 방식
- 프로세스 그룹을 나누어 통신 부하를 줄이고 그룹의 master에서 병렬 입출력을 수행

고성능 입출력 라이브러리 개발 (KIO)

출력 성능



순차 및 병렬 입출력 성능

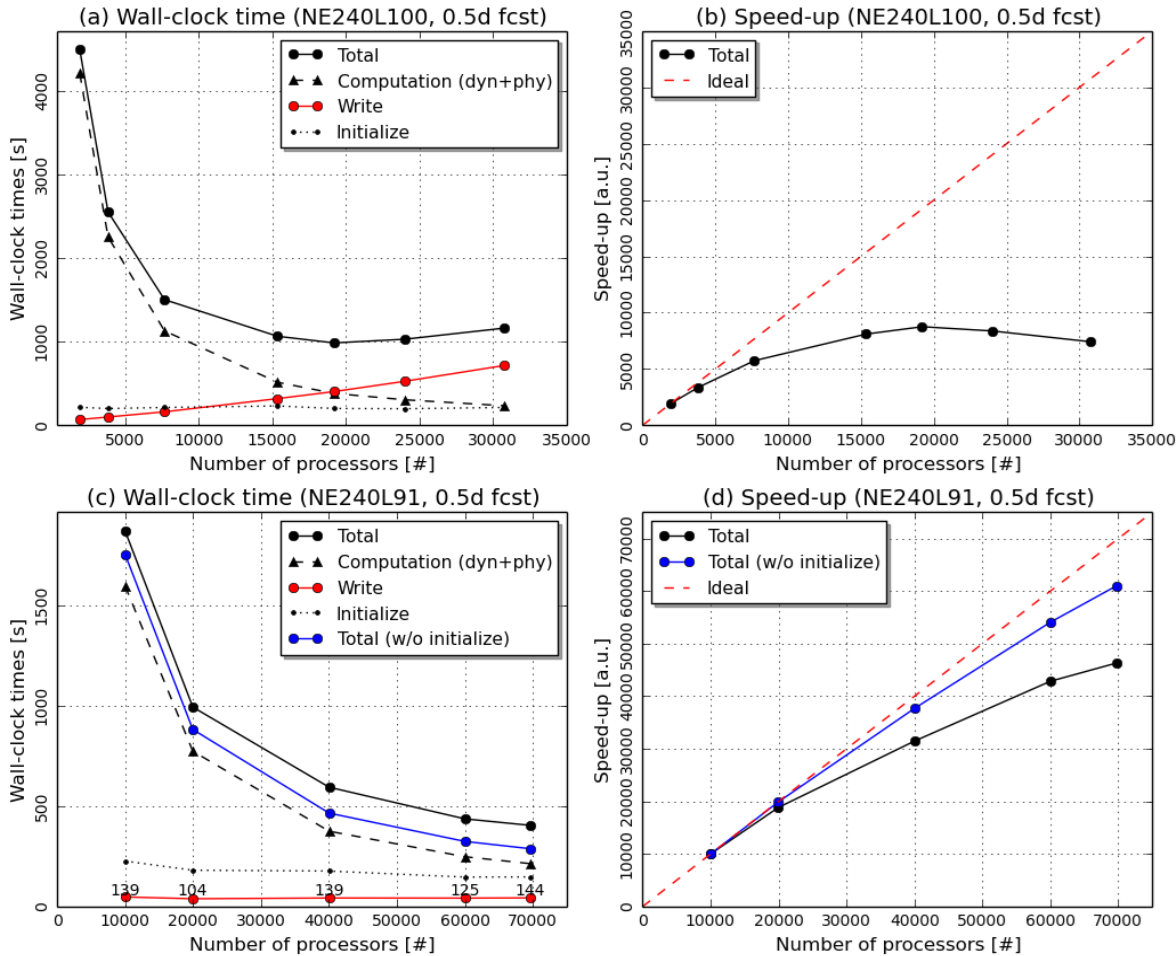


I/O 분할 기법의 출력 성능

- 1) 순차 입출력: 통신 부하로 인해 기본적인 성능이 낮음
- 2) 병렬 입출력: 프로세스 개수가 약 150개일 때 최대 성능을 갖고 이후 동시 파일 쓰기 부하로 성능이 떨어짐
- 3) I/O 분할 기법: 병렬 입출력의 쓰기 부하를 줄여 성능이 개선됨

KIM 모델의 병렬 확장성

모델의 병렬 확장성



- 1) 모델 병렬화: 모델 계산 부분은 $O(100,000)$ 개수까지 병렬확장성을 가짐
- 2) 병렬 입출력: 프로세스 개수가 증가할 때 동시 쓰기에 대한 부하로 약 10,000 프로세스 개수에서 성능이 감소
- 3) I/O 분할 기법: 출력에 참여하는 프로세스 개수를 100~150개로 고정하게 되면 출력에 대한 부하 낮아짐

• 슈퍼컴퓨팅 기반 고성능 수치 모델 개발

- 모델링 프레임워크 개발
 - 자체 개발한 계층적이고 컴포넌트 기반 구조의 프레임워크
 - 다양한 연구자 편의 도구: 병렬 환경, 달력, 입출력, 커플러 등
 - 역학과 물리의 연동
- 모델 격자의 병렬화
 - SFC 기반 도메인 분할
 - 요소들 간의 상호 작용을 위한 MPI 통신 주소 매핑
 - 요소 경계 간의 정보 전달
- 고성능 입출력 라이브러리 개발
 - 순차, 병렬, I/O 분할 기능 제공
 - 모델의 병렬 확장성 보장

• 기타 병렬화/최적화

- 계산-통신 중첩 기법을 이용한 성능 개선
- 중복되는 점들이 제외되는 통신 방법 개발
- 다중 해상도 계산을 위한 도메인 병렬화
- 예보 시스템에 대해 KIO 라이브러리 적용

감사합니다.

