

AI 및
클라우드 컴퓨팅 시대를 위한
고성능 컴퓨팅



NVIDIA

AI 및 클라우드 컴퓨팅 시대를 위한 고성능 컴퓨팅

HPC용 NVIDIA 프레임워크	2
HPC - 새로운 과제, 동일한 목표	4
HPC(High-Performance Computing)의 목표	4
인사이트 극대화	4
값비싼 리소스의 효율성 극대화	5
무엇이 최적의 성능 지표인가?	6
애플리케이션 처리량과 작업 성능 비교	6
병렬 처리의 비효율성 성능	6
지표 측정	9
HPC의 트렌드	11
에너지 효율성 강조	11
탄소 배출 감축	12
수냉 및 액침 냉각 HPC용 ARM64	13
HPC용 ARM64	15
GPU 파워 튜닝	16
직류 전원 데이터 센터	17
데이터 기반 검색	18
전용 실리콘 가속 기술의 부상	19
HPC의 데이터 소스(에지) 배치	20
양자 컴퓨팅이란 무엇인가?	22
메타버스와 디지털 트윈의 부상	23
HPC 핵심 요소	25
HPC 애플리케이션 지원 및 적용	25
범위 HPC와 AI의 융합	27
시뮬레이션 가속화	27
방정식 향상 머신 러닝 HPC 혁신	31
HPC 혁신	33
코히어런트 시스템과 가속기 메모리	34
컴퓨터 네트워크의 발전	36
스케일링을 위한 네트워크 대역폭	36
다중 GPU 및 다중 노드 가속화를 위한 상호 연결	38
클라우드 네이티브 슈퍼 컴퓨팅	42
SmartNIC 오프로드	44

컨버지드 가속기와 네트워크 인터페이스 카드 HPC	45
개발자 에코시스템	46
활발한 커뮤니티	46
표준, 개방형 및 포터블 병렬 프로그래밍 모델 클라우드, VM,	47
컨테이너	52
결론	54

HPC - 새로운 과제, 동일한 목표

지구 생명의 기원을 탐색하거나 자동차를 안전하게 자율 주행하기 위한 스키마를 구축할 때, 혹은 금융 거래에서 사기 행위를 탐지할 때 [HPC\(High-Performance Computing\)](#)를 사용하면 새로운 인사이트를 생성 및 입증할 수 있습니다.

이렇게 도출된 인사이트는 대체로 새로운 것이기에 상당한 가치가 있습니다. 당연히 HPC 시스템의 주요 목표는 비용을 최소화하면서도 귀중한 지적 재산(IP)을 최대한 빠르게 생성하는 것입니다.

데이터 센터 관리자, 신진 HPC 개발자, 베테랑 연구원 등 이 전자책을 읽는 독자라면 누구나 종사하는 분야에 관계없이 많은 정보를 찾을 수 있을 것입니다. HPC 처리량을 높이는 요인을 알아내는 것 외에도 HPC의 현재 분위기를 둘러싼 메커니즘과 환경을 심층적으로 이해하고, 업계 사례를 통해 이렇게 끊임없이 진화하는 HPC 분야에 대해 배울 수 있습니다.

NVIDIA의 향후 HPC 이니셔티브가 어떤 모습일지 궁금하신 분들은 페이지 곳곳에 나와 있는 챕터와 예제, 사용 사례 및 블로그 링크를 주기적으로 확인하시기 바랍니다.

HPC(High-Performance Computing)의 목표

인사이트 극대화

HPC에서는 대체로 지적 재산이 "작업(job)"이라고 하는 워크로드 태스크를 완료함으로써 생성됩니다. 따라서 HPC 시스템의 가치를 최적화하기 위해 가장 간단히 극대화할 수 있는 것이 바로 처리량입니다.

그러나 사용자가 여러 명인 다학제 HPC 센터와 실험실에서는 워크로드 태스크가 고유하게 정의된 단일 엔터티가 아닙니다. 태스크 다양성 스펙트럼은 특성 분석 중인 HPC 환경의 유형과 보통 상관 관계가 있습니다.

[도쿄 대학의 TACC Frontera 및 Wisteria](#)와 유사한 중앙 집중식 HPC 환경을 지칭하는 아카데미 시스템은 대체로 가장 광범위한 분야를 지원합니다. 이는 일반적으로 해당 분야의 주요 조사 및 탐색 톨로 HPC를 사용하는 연구 팀과 부서 인력이 다양하기 때문입니다.

상업용 HPC 센터는 HPC 사용 범위를 제품의 정의, 인증 및 지원으로 대폭 제한하는 경향이 있습니다.

이는 집적 회로 설계([삼성의 SSC-21](#)), [소비재](#), [항공우주 산업\(일본 항공우주 연구개발기구의 TOKI-SORA\)](#), [에너지\(ENI S.p.A의 HPC5\)](#) 및 자동차 등 다양한 영역에도 해당하는 이야기입니다. 이들 각 시스템은 다양한 분야를 지원하는 데 사용됩니다.

예를 들어, 소비자 제품 HPC 시스템은 전체 워크로드의 일부만으로도 유체 흐름, 구조적 무결성, 열 역학, 각종 화학 공정, 품질과 비용에 대한 제조 공차, 그리고 제품 수명을 시뮬레이션할 수 있습니다. 이처럼 광범위한 분야에서 시뮬레이션을 수행하기 위해서는 HPC 센터가 최소한 수십 개의 애플리케이션을 지원해야 합니다.

분명한 사실은, 인사이트나 처리량을 극대화하려면 각종 HPC 애플리케이션의 속도를 최대한 높여야 한다는 것입니다.

값비싼 리소스의 효율성 극대화

NVIDIA는 HPC 시스템을 IP 생성기로 포지셔닝했지만, 이렇게 중요한 기능을 수행하려면 다수의 입력이 필요하다는 사실을 견지해야 합니다. 기계이기 때문에 에너지, 유지보수, 소프트웨어 및 냉각이 필요하지만 이 외에 정보도 필요합니다. 정보는 HPC 애플리케이션이 수용할 수 있는 방식으로 정리되어야 하며, 사람이 입력과 애플리케이션 출력을 모두 이해하고 검증할 수 있는 방식으로 구성되어야 합니다.

따라서 HPC 시스템의 효율성을 최적화하려면 이러한 요소와 관련 비용을 고려해야 합니다. 아래 목록에는 비용이 높은 것에서 낮은 순으로 나와 있고, 이점을 극대화하기 위해 주의를 기울여야 하는 영역이 제시되어 있습니다. 아래 목록의 순서는 기업, 시장 및 관련 기술에 따라 약간씩 조정될 수 있습니다.

- 연구원/엔지니어/과학자 시간
- 소프트웨어 라이선스 시간 - 일부 HPC 소프트웨어는 상업용
- 기기 시간 - 입자 가속기(collider), 현미경, 시퀀서, 망원경 등
- 컴퓨팅 리소스 시간

무엇이 최적의 성능 지표인가?

병렬 처리로 인해 발생하는 비효율에 대응하면서 애플리케이션 처리량과 작업 성능을 고려하는 지표가 최적의 성능 지표입니다.

애플리케이션 처리량과 작업 성능 비교

주어진 HPC 애플리케이션과 환경 세트에서 사용자가 다른 선택을 함으로써 HPC 시스템의 효율성에 상당한 영향을 미치는 경우가 종종 있는데,

바로 처리량과 개별 작업 성능을 적절히 조율하는 방법을 선택하는 것입니다.

이는 때에 따라 쉽게 선택할 수 있습니다. 개중에는 단일 스레드만 실행할 수 있는 애플리케이션이 있는데 이는 전자 설계와 유전체학에서 흔히 볼 수 있는 것으로, 스케일링과 관련된 병렬 처리의 비효율성이 드러나지 않습니다.

HPC 센터는 생산 기한이나 비즈니스 목표, 발행일, 기타 외부 효과 등의 인사이트를 생성하기 때문에 다른 요소의 요구사항을 충족하려면 최대 처리량을 포기해야 합니다. 즉, 일명 분산 메모리 병렬 처리라고 하는 다중 노드 병렬 처리를 사용해 워크로드를 실행함으로써 속도는 빨라지고 효율성은 낮아진 상태에서 워크로드의 우선 순위를 지정하여 실행하는 것입니다.

널리 사용되는 HPC 애플리케이션은 대부분 이러한 다중 노드 병렬 처리로 실행할 수 있으며, 모두 ['암달의 법칙 \(Amdahl's law\)'](#)이 적용됩니다.

병렬 처리의 비효율성

컴퓨팅 입문자라면 병렬 컴퓨팅이 일반적으로 효율성에 불이익을 준다는 사실에 당혹감을 느낄 수도 있습니다. '암달의 법칙' 창시자인 Gene Amdahl은 병렬 처리 시 태스크가 확장할 때 효율성이 저하되는 방식을 보여주는 계산법을 만들었습니다. 그림 1은 그의 방정식을 사용한 예입니다.

여기서 유의해야 할 점은 '암달의 법칙'은 이론일 뿐, 실제 HPC 시뮬레이션 성능은 가정 단순화의 위반으로 인해 완벽한 상관 관계가 성립하지 않는다는 것입니다. 여기 나온 차트와 논의는 HPC 접근 원리를 설명하기 위한 것입니다.

그림 1. 다양한 병렬 분수부에서 가속 계수 상승

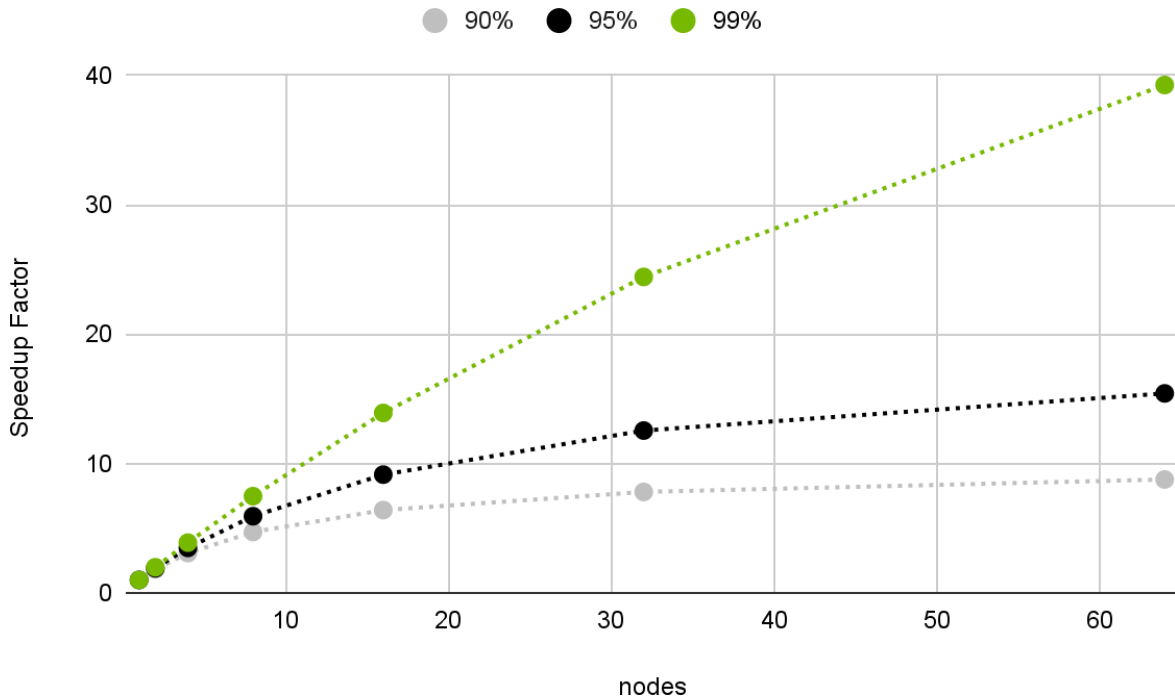


그림 1에는 다양한 병렬 분수부가 있는 애플리케이션에 더 많은 컴퓨팅 리소스를 사용해 가속 계수를 상승시키는 방법이 나와 있습니다.

그림 1을 해석하기 위해 몇 가지 정의를 살펴보겠습니다.

- 가속 계수: 한 노드의 런타임을 나누는 계수(제법이 클수록 좋음)
- 병렬 분수부: 태스크의 분수부(이 경우에는 병렬 처리가 가능한 시뮬레이션)

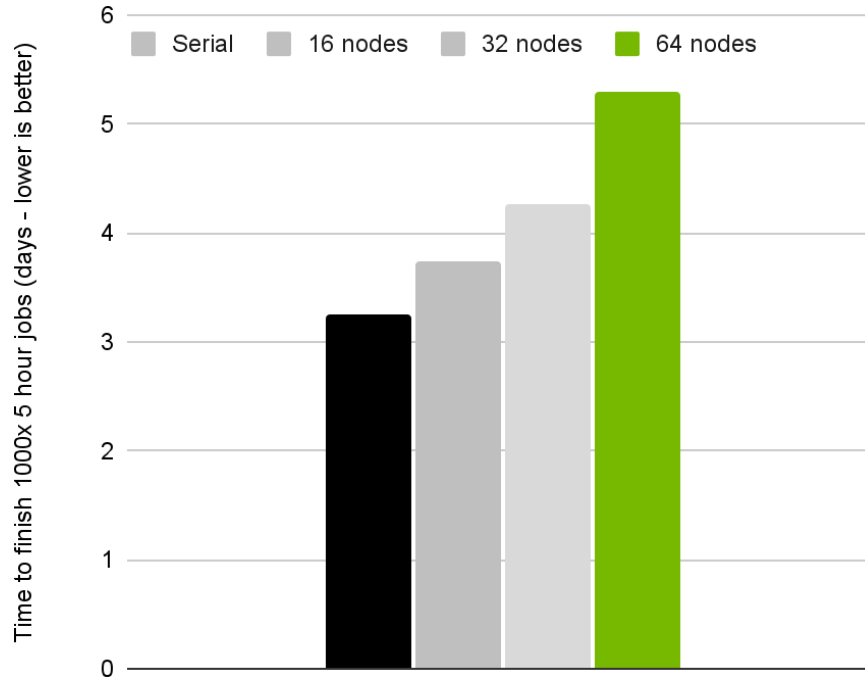
여기서 명확히 알 수 있는 점은 병렬 분수부가 클수록 가속 계수가 커진다는 것입니다. 이것은 태스크가 100% 병렬 처리될 때 해당하는 말입니다. 하지만 문제는 대부분의 태스크가 100%에 도달하지 못한다는 것입니다.

또한 이 차트에서 보여주듯이 다수의 노드에서 가속을 크게 높이는 것은 시뮬레이션의 병렬 분수부가 100%에 얼마나 근접하냐에 달려 있습니다. 이것은 효율성을 희생해야 하는 중요한 문제입니다. 95%가 병렬 처리되는 태스크에 대한 차트를 검토해보면 노드 수를 32개에서 64개로 늘려도 증분하는 가속 계수 측면에서 이점이 거의 없다는 것을 알 수 있습니다. 따라서 32개의 노드를 추가하더라도 런타임을 줄이는 데는 거의 효과가 없습니다.

물론, 처리량을 극대화하기 위해 탐구해 볼만한 내용입니다. 좀 더 구체적으로 설명하기 위해, 한 노드에서 5시간 동안 실행되며 99% 병렬 처리되는 시뮬레이션이 있다고 가정하겠습니다.

그리고 노드 수가 64개인 HPC 환경에서 이러한 시뮬레이션 중 1,000건을 가능한 한 빨리 처리해야 합니다. 여기서 질문입니다. 1,000개의 작업을 최대한 빠르게 완료하려면 각 시뮬레이션을 몇 개의 노드에서 실행해야 할까요?

그림 2. 병렬 처리 양을 각각 다르게 설정하여 5시간이 소요되는 1000개의 작업을 완료하는 데 걸리는 시간



‘직렬’이란 “한 개의 노드를 사용하는 것”을 뜻합니다.

아마 놀라시겠지만, 모든 시뮬레이션을 처리하는 가장 빠른 방법은 각 시뮬레이션을 단일 노드에서 실행하고, 그 중 64개를 동시에 실행하는 것입니다. 그렇다면 시뮬레이션을 왜 병렬 방식으로 실행하는 것일까요? 여기에는 몇 가지 이유가 있는데, 대표적으로 다음과 같습니다.

쉽게 설명하자면, 1,000개의 태스크 워크로드 전체가 완료되기 전에 앞서 사용자에게 중간 결과를 제시하는 것이 대체로 효과적이기 때문입니다. 이 경우에는 가장 값비싼 리소스인 연구원의 업무 시간에 맞게 최적화하는 것이 더 효율적일 수 있습니다. 작업당 16개 노드에서 1,000개의 시뮬레이션 워크로드 전체를 완료하는 데 12시간이 넘게 걸리지만, 각 시뮬레이션은 5시간 동안 실행되는 대신에 21분 만에 완료됩니다.

연구원이 보기에 단일 시뮬레이션의 결과가 흥미롭고 유용하거나, 혹은 시뮬레이션 결과에 대해 몇 가지 후처리를 수행해야 하는 경우에는 인사이트를 제공하는 시간을 최적화하는 작업에 다중 노드 병렬 처리가 포함될 가능성이 매우 높습니다. 다중 노드 병렬 처리를 사용해야 하는 상황은 다음과 같습니다.

1. 많은 양의 메모리가 필요하고 단일 노드에서 실행할 수 없는 대규모 시뮬레이션에는 주로 병렬 프로그래밍이 필요합니다. 병렬 처리에는 흔히 일종의 도메인 분해가 수행되기 때문에 노드당 필요한 메모리의 양은 병렬 태스크에 참여하는 노드의 수만큼 대략적으로 줄어듭니다.
2. 앞의 경우와 비슷하게, 때로는 감당할 수 없을 만큼 긴 시간이 걸리는 시뮬레이션을 단일 노드에서 실행해야 하는 경우가 있습니다. 이 경우에는 병렬 처리의 비효율성을 수용해야만 합니다.
3. 이상적으로 여겨지는 양보다 적은 양의 병렬 처리가 스케일링에 좀 더 효과적인 경우가 있습니다. 즉, 컴퓨팅 리소스를 두 배로 늘리면 런타임이 절반 미만으로 줄어드는데, 이를 보통 [초선형 스케일링](#)이라고 합니다.

성능 지표 측정

HPC를 사용해 본 대부분의 사람들은 지난 수십 년 동안 슈퍼 컴퓨터의 순위를 정하는 데 사용된 Top500과 관련 HP-Linpack 벤치마크를 즉시 떠올릴 것입니다. FP64 행렬 연산은 전 세계 슈퍼 컴퓨팅 센터에서 흔히 볼 수 있는 기존의 3D 물리 시뮬레이션에 적합한 아날로그입니다.

그러나 HPC의 특성이 변화하고 있습니다. 데이터 집약적 워크로드, 혼합/감소 정밀도 알고리즘, AI 및 머신 러닝(ML)은 예측 속도와 정확도를 한층 개선할 수 있도록 HPC의 실행 방식을 보강하고 있습니다.

이 섹션에서는 HP-Linpack에서 지원하고 있지 않지만 새롭게 떠오르고 있는 몇 가지 방법과 정밀도에 중점을 두는 정량적 비교에 사용할 수 있는 몇 가지 새로운 틀에 대해 설명합니다. 이러한 측정값 중 적어도 한 가지는 앞으로 차세대 슈퍼 컴퓨터를 고를 때 구매를 결정짓는 주요 요인이 될 것으로 예상됩니다.

[Graph 500](#) - 데이터 처리와 3D 물리 시뮬레이션 간의 차이를 강조하도록 설계된 데이터 집약형 워크로드입니다. 이 두 가지는 HPC에 중요한 처리 방식이지만, 데이터 중심의 워크로드는 HPC 커뮤니티에서 홀대받는 분위기입니다. 그래프 알고리즘은 다수의 데이터 집약형 워크로드의 핵심 부분으로, 이 벤치마크는 HPC 제안 프로세스에서 사용되는 조달 비교 방법들을 잘 조율하도록 설계되었습니다.

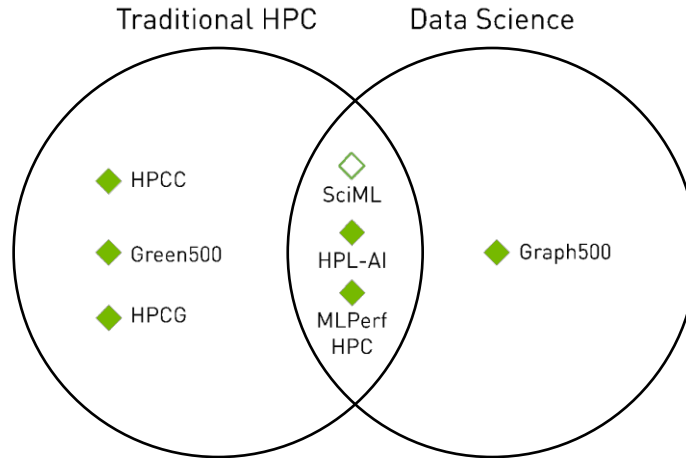
[Green 500](#) - top500의 한 분기로, 와트당 FLOPS라는 결합 메트릭을 제공하기 위해 실행 시 최대 와트 사용량으로 나눈 top500 값에 대해 제출된 HP-Linpack 점수를 사용합니다.

[HPCC\(HPC Challenge\)](#) - HP LINPACK, DGEMM, STREAM PTRANS, RandomAccess, FFT, b_eff 기반의 통신 지연 시간 및 대역폭을 포함하는 종합 벤치마크 세트입니다.

HPCG(High-Performance Conjugate Gradient) - 다양하고 광범위한 중요 HPC 애플리케이션에 더 밀접하게 일치하는 계산 및 데이터 액세스 패턴을 연습하도록 설계되었습니다. 또한 컴퓨터 시스템 설계자에게 이러한 애플리케이션의 집단 성능에 영향을 미치는 기능에 대한 투자를 장려하도록 설계되었습니다.

HPCG에 포함된 대체 연산으로는 희소 행렬-벡터 곱셈, 벡터 업데이트, 전역 내적 등이 있습니다.

그림 3. 데이터 과학과 AI/ML을 향한 HPC 벤치마크의 진화



HPL-AI - 64비트 부동 소수점 정밀도로 행렬 수학을 수행하는 HPL과 유사한

합성 벤치마크입니다. 이 AI 버전은

머신 러닝 모델이 훈련에 주로 낮은 정밀도의 수학(fp32, 16, 8, 심지어 int8까지)을 활용하고, 정확도를 낮추지 않으면서 높은 처리량을 달성한다는 사실을 인식하고 낮은 정밀도 수준에서 행렬 수학을 수행합니다.

MLPerf HPC - ML Commons는 머신 러닝에 중점을 둔 벤치마크 그룹을 정의하기 위한 또 다른 컨소시엄 접근 방식입니다. 훈련과 추론에 대한 측정이 분리되고, HPC 사용 사례를 위한 머신 러닝 훈련인 세 번째 HPC 중심 벤치마크가 강력한 스케일링이라 불리는 최대 성능과 약한 스케일링이라 불리는 처리량 모두에 대해 정의 및 측정됩니다. NVIDIA의 2021년 성과와 이를 달성해낸 비결을 확인해보세요.

SciML - 머신 러닝을 사용하는 과학 기술 컴퓨팅의 사례를 보여줍니다. 현재 개발 중인 컬렉션이기 때문에 위의 그림에서 속이 비어 있는 다이아몬드로 표시되어 있지만, 벤치마킹에 사용할 수 있습니다.

HPC의 트렌드

에너지 효율성 강조

컴퓨팅이 환경에 미치는 영향에 대한 인식에는 몇 가지 요인이 작용하는데, 대표적으로 성장률이 있습니다. 우리의 삶은 정보가 날마다 수집, 저장, 선별, 처리, 선택 및 제공되는 연속적인 정보 흐름에 둘러싸여 있습니다. 이러한 모든 정보는 인터넷 곳곳에 흩어져 있는 수십억 개의 처리 및 저장 장치에 의해 정제됩니다. 그리고 이러한 처리 장치의 수는 정보에 대한 수요를 충족하기 위해 기하급수적으로 증가하고 있습니다.

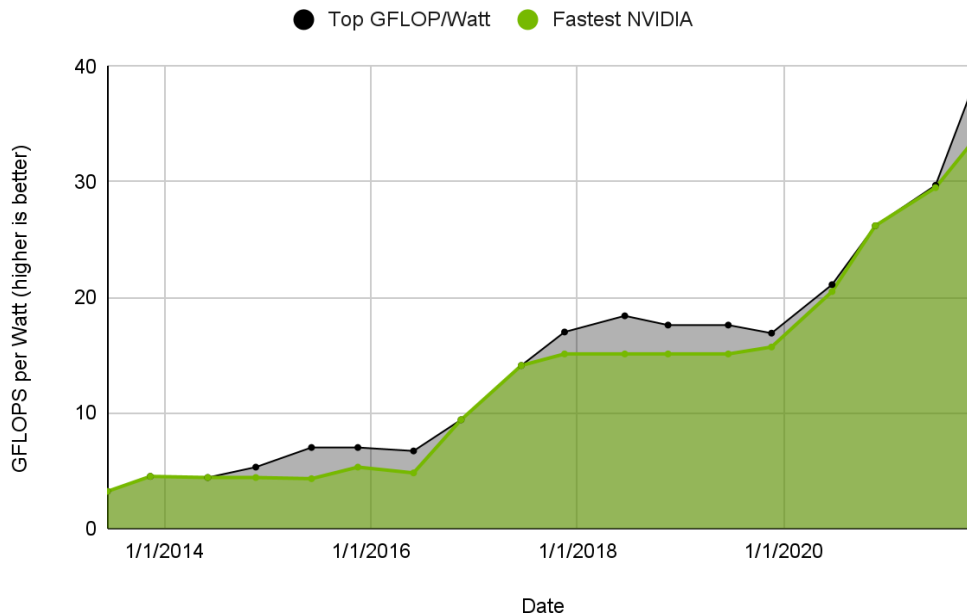
HPC는 전 세계 총 컴퓨팅 사용량에서 차지하는 비중이 적지만, 상품화를 거쳐 가장 무난한 장치에서 사용되는 기술들을 개발 및 입증하기 위한 도가니로 사용되고 있습니다.

이러한 사실을 깨닫게 된 업계 리더들은 2005년에 성능과 전력을 함께 추적하는 벤치마크인 Green500 목록을 발표했습니다. 이 목록은 이제 자사가 관리하는 HPC 환경의 전력 효율을 극대화하는 데 기여한 HPC 리더를 인정하는 기본 형식이 되었습니다.

HPC 시스템에서 소비되는 전력은 원자력, 지열, 바람, 태양열 같은 무탄소 에너지 시스템을 사용하지 않는 대부분의 지역에서 CO2 배출과 직접적인 연관이 있을 수 있습니다. 기후 변화에 대한 가시성과 우려가 높아짐에 따라 컴퓨팅 분야의 리더들은 기업의 탄소 중립을 달성하기 위해 과감한 행보를 이어왔습니다. 그리고 대부분이 자사가 관리하는 데이터 센터를 포함해 다른 부분에서도 이러한 임무를 다하고 있습니다.

NVIDIA는 지난 10년에 걸쳐 GPU 가속 컴퓨팅을 HPC 전반의 표준으로 정립함으로써 효율성 측면에서 상당한 발전을 이루었습니다. 그림 4에는 시간 경과에 따른 효율성 향상을 비롯해, NVIDIA가 이 중요한 지표에서 선두를 차지하게 된 보유 실적이 나와 있습니다.

그림 4. Green500 목록의 최근 8년 간 히스토리와 상위 NVIDIA GPU 시스템 점수의 중첩 부분



2021년 가을 GTC에서 NVIDIA는 "Earth-2"라는 새로운 슈퍼 컴퓨터를 개발할 것을 공약하면서 전 세계 기후 변화 연구에 대규모 투자를 단행할 것이라고 발표했습니다. NVIDIA 사장 겸 CEO인 Jensen Huang은 직접 [이에 대한 글을 블로그에 게시했습니다](#). Earth-2는 미터 스케일 해상도 시뮬레이션을 사용해 디지털 트윈에 지구의 기후 상태를 알립니다. 이러한 스케일은 최종적으로 구름 형성과 관련된 물리적 성질을 포착하게 되는데, 이는 정확한 장기 예측을 비롯해 반사 및 흡수된 햇빛을 모델링하는 데 중요한 부분입니다.

탄소 배출 감축

인류가 지구 기후에 미치는 영향에 대해 많은 발표가 있었던 것처럼 양적 통계 역시 다양합니다. [2018년에 시행된 한 연구](#)에 따르면, 전 세계 데이터 센터들이 글로벌 수요의 약 1%인 200TWh의 전력을 소비하면서 당해 CO2 배출량의 0.3%를 차지했다고 합니다. 하지만 이러한 계산이 어떻게 나왔는지를 두고 논쟁이 벌어졌습니다.

이러한 논쟁에서 몇 가지 의문이 제기되었습니다. 데이터 센터의 컴퓨터, 스토리지, HVAC 및 기타 구성 요소의 제조와 관련된 CO2를 포함시켜야 하는가? 현지의 발전 방식을 정확하게 설명하거나, 글로벌 평균을 사용하고 있는가?

데이터 센터의 환경 발자국에 대한 [또 다른 연구](#)에서는 주로 미국에 초점을 맞추고 있는데, 이는 전 세계 데이터 센터의 1/3이 미국에 있기 때문입니다. 이 연구에서는 총 온실 가스 배출량의 0.5%가 데이터 센터에서 나오는 것으로 추정했습니다.

또 다른 연구에서는 놀랍게도 데이터 센터가 전 세계 항공 산업과 거의 같은 양의 CO2를 배출하고 있다고 추정했습니다.

위의 연구와 상관없이, 데이터 센터가 미치는 영향에 초점을 맞추는 주된 이유는 하이퍼스케일의 데이터 센터가 기하급수적으로 증가하고 있기 때문입니다. 이러한 증가는 컴퓨팅의 효율성 향상 속도를 훨씬 능가하며, 이렇게 엄청난 양의 전기를 잡아먹는 사이트의 영향을 최소화할 수 있는 유일한 방법은 청정 재생 에너지만 공급하는 것입니다.

수냉(Water Cooling) 및 액침 냉각(Immersion Cooling)

데이터 센터에 청정 재생 에너지만 사용하도록 추진하는 것이 중요합니다. 데이터 센터의 에너지 소비 효율을 높이는 것도 이러한 목표를 달성할 수 있는 한 가지 방법입니다. 데이터 센터의 에너지 효율성을 측정하는 척도는 PUE(Power Usage Effectiveness)입니다. PUE에서는 완벽하지는 않지만, 데이터 센터로 들어오는 총 에너지를 해당 데이터 센터의 서버에 공급되는 전력으로 나눈 값을 분수부에서 정량화하려고 합니다.

여기서 목표는 데이터 센터 PUE를 1.00에 최대한 가깝게 만들어서 소비되는 모든 에너지가 냉각, 공기 이동, 물 펌핑, AC-DC 변환 등의 다른 기생적 요구 사항 총족이 아닌, 컴퓨팅 구동에 사용되도록 만드는 것입니다.

오늘날 대부분의 데이터 센터가 운전 온도 범위 내에서 장비를 가동하기 위해 냉각 공기를 사용하고 있습니다. 이러한 기준으로 인해 몇 가지 표준 기술과 빌드 관행이 개발되었고, 이로 인해 다음과 같이 데이터 센터의 비용과 비효율이 유발되고 있습니다.

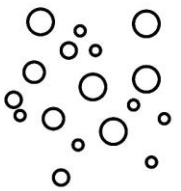
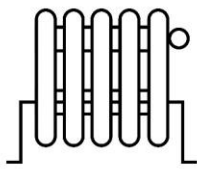
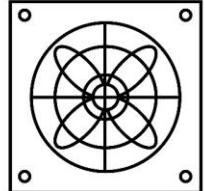
- 무거운 랙 중량을 견뎌야 하는 이중 바닥
- 움직일 수 있는 바닥 타일
- 데이터 센터 내 CRAC(Computer Room Air Conditioner) 장치
- 데이터 센터 내부와 바닥 아래의 공기를 순환시키는 HVAC 장치
- 인체에 해롭다고 여겨지는 소음 및 진동 상태

안타깝게도 공기 냉각의 열역학으로 인해 이렇듯 불가피한 비효율이 발생하고 있는데, 공기의 열전도율과 열용량이 주된 요인입니다.

미래를 내다보는 몇몇 데이터 센터들은 CPU, GPU, 메모리 및 네트워킹과 같이 서버에서 전력 소모가 많은 부분에 직접 수냉 방식을 사용하는 방법을 조사했습니다. 수냉식은 공냉식보다 훨씬 효율적이지만 펌프와 냉각기, 그리고 상당량의 배관이 필요합니다.

가장 혁신적인 데이터 센터들은 전체 서버를 전기적으로 절연된 일종의 열 전도성 액체에 담그는 [액침 냉각 방식](#)을 검토하고 있습니다. 액체의 열적 특성이 열전달과 정전용량 측면에서 1,000배 더 뛰어나기 때문에 이러한 액침 냉각은 매우 큰 장점이 있습니다. 그림 5는 이러한 기술을 기반으로 하는 데이터 센터의 PUE 차이를 보여줍니다.

그림 5. 데이터 센터의 냉각 기술 PUE 비교

	Two-phase immersion	Water cooling	Air cooling
			
PUE	1.02 - 1.05	1.05 - 1.10	1.40 - 1.75

2상 액침 냉각이라고 하는 특수 액침 냉각 사례에서는 섭씨 45~55도(화씨 113~131도)에서 끓는 특수 액체를 사용합니다. 이러한 유형의 액침 냉각은 냉각액의 기화열을 이용하기 때문에 훨씬 더 효율적입니다. 서버 부품에서 나오는 모든 폐열은 냉각액을 끓이는 데 사용됩니다. 그런 다음, 증기가 주변 온도 응축기로 순환되는 과정에서 액체로 다시 바뀌어 냉각 튜브로 되돌아갑니다.

장점:

- 데이터 센터 PUE가 급격하게 감소
- 이중 바닥이 더는 필요하지 않기 때문에 구축 비용이 저렴해짐
- 표준 액침 냉각의 경우, CRAC 시스템이 필요하지 않으며, 보다 간편하고 저렴한 열 교환기를 사용할 수 있음
- 이색적인 2상 액침 냉각의 경우, 열 교환기가 전혀 필요하지 않음
- 물 사용량이 대폭 줄거나 사라짐

단점:

- 액침 냉각 인증을 받은 보증 및 하드웨어 가용성
- 액체 누출 및 흘림
- 중량
- 바닥에 수평으로 놓인 냉각수 튜브로 인해 데이터 센터의 서버 밀도 증가

Microsoft의 경우, [특별 설계된 냉각액](#)에 서버 랙을 담그는 방식으로 에너지 사용량을 줄여 냉각 비용을 대폭 절감할 수 있다는 것을 입증했으며, 현재는 [수밀 컨테이너에 소규모 데이터 센터를 담아서](#) 해저에 매설하는 방법도 모색 중입니다.

HPC용 ARM64

Top500을 기준으로 2020년 6월 이후 세계에서 가장 빠른 슈퍼 컴퓨터는 750만 개 이상의 코어로 이루어져 있고 30MW로 구동되는 슈퍼 컴퓨팅 세계의 거석 Fugaku였습니다. 그러나 Fugaku가 돋보이는 것은 단지 스케일 때문만은 아닙니다. Fugaku는 Fujitsu에서 제작한 A64FX라는 전용 CPU인 ARM64 프로세서로 구동됩니다.

2019년 말 출시 당시, A64FX는 17 GFLOP/와트에 살짝 못 미치는 성능을 보이며 Green500 목록에서 1위를 차지했습니다. 그 후로 반도체 다이가 작아지고 라이브러리에서 프로세서 하드웨어를 보다 효율적으로 사용하게 되면서 17(위의 그림 5 참조)도 이제는 좋은 성능이라고 말할 수 없게 되어버렸습니다. 그러나 몇몇 민간 기업들이 [Neowise 코어](#)와 관련해 새로운 활동을 추진하고 있고 ARM64용 가속 컴퓨팅이 도입되면서 이 새로운 HPC 시스템 아키텍처가 온라인 환경에 제공됨에 따라 효율성이 크게 향상될 전망입니다.

커뮤니티에서 진지하게 검토하고 있는 HPC 플랫폼의 평가 기준 중 하나는 해당 플랫폼에서 생성 중인 중요하면서도 일반적인 HPC 애플리케이션의 존재 여부입니다. 이러한 평가 기준에 따르면, 이 문서가 작성되던 시점에 GAMESS, GROMACS, LAMMPS, OpenFOAM, WRF, NAMD 및 MILC용 포트가 존재한다는 점에서 [ARM64는 이미 진지하게 검토되고 있습니다](#). 다른 애플리케이션들은 아직 초기 단계이지만, 커뮤니티에 아직 발표되지 않은 채 작업 중인 개발 및 테스트 버전이 있을 수 있습니다.

플랫폼에 대한 애플리케이션이 존재하는 경우, 다운스트림 평가 기준은 해당 플랫폼이 세계 최고의 슈퍼 컴퓨팅 센터에서 채택되고 있는지 여부입니다. NVIDIA가 2021년 봄에 열린 [GTC에서 Grace 프로세서](#)를 발표하고, [스위스 국립 슈퍼 컴퓨팅 센터\(Swiss National Supercomputing Center\)](#)에서 HPE 플랫폼을 이기고 슈퍼 컴퓨터로 채택됨에 따라 ARM64도 주목받고 있습니다.

그러나 HPC를 겨냥한 ARM64 기반 프로세서를 공개적으로 논의하는 기업이 NVIDIA만 있는 것은 아닙니다. 지역의 HPC 기술 역량은 글로벌 경제이자 주권을 지키는 보루라는 점에서 민간 기업인 SiPearl은 [유럽 프로세서 이니셔티브\(European Processor Initiative\)](#)라고 하는 [유럽 컨소시엄](#)을 출범했습니다. EPI 프로젝트에 대한 SiPearl의 비전은 관련 HPC 기술 및 애플리케이션과 더불어, 유럽 슈퍼 컴퓨터를 위한 고성능 및 저전력 마이크로프로세서를 자체 설계하는 것입니다. 게다가 Amazon Web Services는 현재 2세대인 ARM64 기반의 Graviton 인스턴스를 가지고 있습니다. 3세대 Graviton은 클라우드에서 가성비를 중시하는 HPC 사례에 집중할 것으로 예상됩니다.

GPU 파워 튜닝

Intel과 AMD 모두 CPU 스테핑 기술을 제공하는데, 관리자는 이를 통해 CPU 주파수를 매우 세분화된 형태로 증감 및 축소할 수 있습니다. 이는 실행 중인 애플리케이션의 성능을 극대화하거나, 소비된 에너지의 와트당 작업 성능을 제어하는 데 도움이 됩니다. 물론, 광범위한 HPC 애플리케이션에 대해 HPC 센터 차원에서 이러한 작업을 완전 자동으로 수행하는 것은

측정, 체계화, 설정 및 유지 관리에 많은 시간이 소요되는 까다로운 프로세스입니다. 이러한 프로세스가 3-5년마다 각 애플리케이션에서 반복되는 것은 말할 것도 없습니다. 따라서 대부분의 경우, 환경을 생각하는 HPC 센터들만 이러한 프로세스를 시도하고 있습니다.

NVIDIA GPU에서도 이와 유사한 제어 기능을 제공합니다. 실제로 최근 [영국 케임브리지 대학교는 DGX-A100 시스템을 설치](#)하고 이러한 주파수 제어 기술을 사용해 A100을 디클로킹함으로써 성능 손실을 10-11%로 제한하면서 전력 소비량을 35-40% 절감했습니다.

일반적으로는, 와트당 GFLOPS를 위해 HPC 앱을 최적화하고 Green500 목록에서 높은 점수를 기록하는 경우에는 이것이 매우 큰 장점으로 다가옵니다. 그러나 실질적인 이점을 얻으려면 소비율이 문제가 아니라 태스크에서 소비되는 총 전력을 최적화해야 합니다. 즉, 태스크의 전력 소비율을 50% 줄이는 것은 좋지만, 태스크 실행 소요 시간이 75% 길어진다면 최대 속도와 전력으로 태스크를 실행하는 것보다 소비되는 총 전력이 더 커지게 됩니다.

다시 말하지만, 최적의 접근법을 알아내는 유일한 방법은 다양한 GPU 주파수에서 애플리케이션에 벤치마킹 측정을 수행하여 성능(런타임)과 전력(소비율)을 측정하는 것입니다. 이러한 데이터를 가지고 있으면 최적화를 완전히 제어할 수 있습니다.

관심 있는 분들은 아래 코드 스니펫에 나와 있는 것처럼 [NVIDIA SMI\(System Management Interface\)](#)를 통해 이러한 종류의 제어 기능을 사용할 수 있습니다.

```
-lgc --lock-gpu-clocks= Specifies <minGpuClock,maxGpuClock> clocks as a pair (e.g. 1500,1500) that defines the range of desired locked GPU clock speed in MHz. Setting this will supercede application clocks and take effect regardless if an app is running. Input can also be a singular desired clock value (e.g. <GpuClockValue>).
```

예 :

```
# nvidia-smi -pm 1
# nvidia-smi -i 0 -pl 250
# nvidia-smi -i 0 -lgc 1090,1355
DISPLAY=:0 XAUTHORITY=/run/user/129/gdm/Xauthority
nvidia-settings \
  -a [gpu:0]/GPUFanControlState=1 \
  -a [fan:0]/GPUTargetFanSpeed=75 \
  -a [fan:1]/GPUTargetFanSpeed=75 \
  -a [gpu:0]/GPUPowerMizerMode=1 \
  -a [gpu:0]/GPUGraphicsClockOffsetAllPerformanceLevels=75
```


직류 전원 데이터 센터

100년도 더 전에 Thomas Edison과 Nikola Tesla는 미국의 전기화 정신을 두고 이론적인 측면과 비즈니스적인 측면에서 전쟁을 벌였습니다. 결국, 많은 양의 전력에 대한 분배 효율성을 이유로 교류(AC)가 승리했습니다. 태양광을 제외하고 대부분의 사람이 거주하거나 일하는 곳의 반경 500미터 내에서는 전력 발전이 이루어지지 않는다는 점에서 이러한 결론은 바뀌지 않을 것으로 보입니다.

문제는 정보화 시대와 그 핵심 기술인 트랜지스터가 직류(DC)로 작동한다는 사실에 있습니다. 그렇기에 모든 디지털 회로에는 필요한 유형의 전력을 제공할 수 있도록 업스트림에 정류기(AC-DC 전원 공급 장치)가 있어야 합니다. 결과적으로 네트워크 스위치에서부터 메인프레임에 이르기까지 거의 모든 데이터 센터 구성 요소에는 변환을 수행하는 전원 공급 장치가 내장되어 있으며, 보통 이중화 처리되어 있습니다. 하지만 물리학적 관점에서 보면 에너지 변환이 완벽히 효율적이지 않기 때문에 그 과정에서 폐열을 생성하게 됩니다. 당연히 전력 변환 횟수가 많을수록 손실도 커집니다. 모든 컴퓨터 내부에 잔존하는 이러한 폐열은 다른 열원에 모이기 때문에 공냉식 데이터 센터 HVAC 및 CRAC 장치를 통해 폐열을 줄여야 합니다.

하지만 랙 또는 행 수준에서 AC 고전압 및 고전류를 DC로 변환하고, DC 전원을 각 서버와 스위치 및 저장 장치에 직접 분배할 수 있습니다. 이 기술에는 다음과 같이 몇 가지 칭찬할 만한 장점이 있습니다.

- 부품의 냉각을 위한 공기 흐름 제약을 제거
- UPS 백업이 가동 중일 때 DC에서 AC로의 변환 단계 제거
- PUE 정의에서 누락된 부분 수정(AC에서 DC로 변환 시 일부 전력 손실)
- 수많은 이중화 전원 공급 장치를 제거하여 중앙 집중식 AC-DC 변환기의 신뢰성을 크게 향상

더 높은 PUE와 최적화된 와트당 GigaFLOPS를 추구하는 과정에서 전력 배분의 재설계는 탐색할만한 가치가 있는 또 하나의 도구가 될 수 있습니다.

데이터 기반 검색

복잡한 환경과 디지털 장치의 폭발적인 증가로 인해 온갖 유형의 데이터가 갈수록 넘쳐나고 있습니다. 마찬가지로 디지털 혁명으로 인해 과학 내지는 과학적 데이터 수집 방식의 본질이 바뀌고 있습니다.

다시 말하자면, 기본적인 물리 법칙을 토대로 세상을 이해하려면 실증적 모델을 구축해야 합니다. 이러한 모델은 과학자가 행동을 예측하고 "가정" 질문에 대한 답을 구할 수 있도록 복잡한 물리적 시스템을 설명합니다. 이러한 질문들은 실증적 모델의 타당성을 확인 또는 부정하기 위한 실험이 됩니다. 이것이 바로 과학적 방법입니다.

오늘날 인간은 다년간의 주제 관련 경험과 직관, 그리고 수학적 감각을 바탕으로 새로운 물리학에 대한 가정을 세우고 이를 이해하고자 노력하고 있습니다. 이에 두 가지 보완적 방법이 등장했습니다. 하나는 순수한 상징적 틀에서 연구하는 이론가의 관점이고, 다른 하나는 이론과 일치 또는 불일치하는 관찰 결과를 수집하는 실험가의 관점입니다. 이 두 가지 관점을 오가면서 주변 세계에 대한 이해를 점진적으로 발전시키고 있습니다.

그러나 머신 러닝 기술이 실제로 적용되면서 새로운 기회가 등장하고 있습니다. 향상된 감지 기능과 유비쿼터스 고대 역폭 네트워킹, 그리고 데이터에 대한 무료 개론 덕분에 그 어느 때보다 많은 데이터를 사용할 수 있게 되면서 자동화된 귀납적 추론 접근 방식을 통해 과학적 발견을 가속하는 모델을 구축할 수 있는 기회가 생겼습니다.

물리학에서 우주의 언어는 편미분방정식(PDE)으로 작성됩니다. 전자기학, 중력, 유체 역학, 지구 역학, 상대성 이론 및 기타 여러 분야에서 PDE를 사용해 포착하고자 하는 현상을 설명하고 있습니다. 현재 사용되고 있는 PDE는 각 분야의 몇몇 이론가들이 개발한 것입니다. 하지만 데이터가 폭발적으로 증가함에 따라, 보편적인 역학을 새롭게 설명할 수 있는 새로운 방법이 등장했습니다. 이 방법이 유망한지 검증하기 위해 오랫동안 연구가 진행되었던 분야인 유체 역학을 [테스트 케이스](#)로 사용해서 이 방법이 순수하게 관찰로부터 지배적인 PDE를 도출할 수 있는지를 확인했습니다. 실제로 [캘리포니아공과대학교](#)는 몇몇 흥미로운 프로젝트에서 새로운 인사이트를 얻기 위해 데이터 활용을 전담하는 센터를 두고 있습니다.

또 다른 예로, 지구과학자들은 [지구에 대한 예측 모델](#)을 구축하기 위해 이 방법을 적용하고 있는데, 이는 기후적인 관점에서 뿐만 아니라, 전체 생물 군계 관점에서 지진, 쓰나미, 허리케인 등 자연 및 인적 재해의 영향을 최소화하기 위한 것입니다.

전용 실리콘 가속 기술의 부상

가속 컴퓨팅은 CPU 전용 접근 방식에 비해 성능 및 에너지 효율성 측면에서 많은 개선 기능을 제공할 수 있으며, 가속기의 기능을 활용하기에 적합한 HPC 워크로드가 많이 있습니다.

GPU, FPGA, ASIC 등 다양한 유형의 가속기가 있으며, 이들은 유연성과 성능 및 에너지 효율성을 각기 다른 방식으로 조율합니다. 한 가지 측면에서 이득을 얻으려면 다른 측면에서 손해를 감수해야 할 수 있습니다.

HPC 애플리케이션은 종류가 다양합니다. 따라서 사용자가 플랫폼의 제약을 받지 않으면서 자유롭게 프로그래밍이 가능하고 다양한 유형의 작업에서 상당히 빠른 속도를 제공할 수 있는 HPC 중심의 가속기가 필요합니다. 이와 동시에 HPC에서 에너지 효율성이 중요해지면서 소비 전력에 대한 이점(성능)을 극대화할 수 있는 가속 기술이 요구되고 있습니다.

HPC에 적합한 가속기를 구축하려면 성능, 에너지 효율성, 프로그래밍 가능성 및 유연성이 최적의 균형을 이뤄야 하는데, 이런 이유로 현재 GPU가 HPC의 가속기로 각광받고 있습니다.

[NVIDIA의 최신 H100](#) 과 같은 최신 GPU는 배정밀도 부동 소수점에서부터 8비트 정수 및 부동 소수점 형식에 이르기까지 다양한 숫자 형식을 대상으로 벡터 및 행렬 연산을 가속합니다. 채택율 통계를 보면 그 가치를 알 수 있습니다. NVIDIA GPU는 700개 이상의 HPC 애플리케이션을 가속하며, 엔지니어링, 의료 및 생명 과학, 천체 물리학, 고에너지 물리학 등 광범위한 영역을 아우릅니다(특정 기능은 그림 7 참조).

또 하나 주목할만한 HPC용 가속 기술이 있는데, 바로 범용 CPU에 통합되어 있는 와이드 벡터 유닛입니다. 이 접근 방식은 Fugaku 슈퍼 컴퓨터의 핵심 장치인 Arm 호환 프로세서, Fujitsu의 A64FX에 사용되고 있습니다.

또한 AI는 과학적 연구 및 발견의 최전선에서 HPC와 AI의 융합을 가속하는 등 점차 중요한 역할을 하고 있습니다. GPU는 AI 워크로드에 가장 널리 사용되는 가속기로, [NVIDIA A100](#)이나 H100과 같은 최신 GPU에는 AI 훈련 및 추론의 핵심인 행렬 수학 연산을 위한 가속 기능이 포함되어 있습니다.

또한 HPC와 AI의 융합은 HPC 워크로드에 특화된 AI 가속기 사용에 대한 관심을 불러일으켰습니다. 일례로 Lawrence Livermore National Laboratory는 [Cerebras Systems](#)와 [SambaNova Systems](#) 등의 AI 가속기 스타트업과 프로젝트를 공동으로 추진한다고 발표한 바 있습니다. 전문 AI 가속기를 공급하는 또 다른 업체인 [Graphcore](#) 역시, 자사 제품을 사용해 HPC 워크로드에서 머신 러닝 활용 범위를 늘리는 방안을 논의했습니다.

HPC의 데이터 소스(에지) 배치

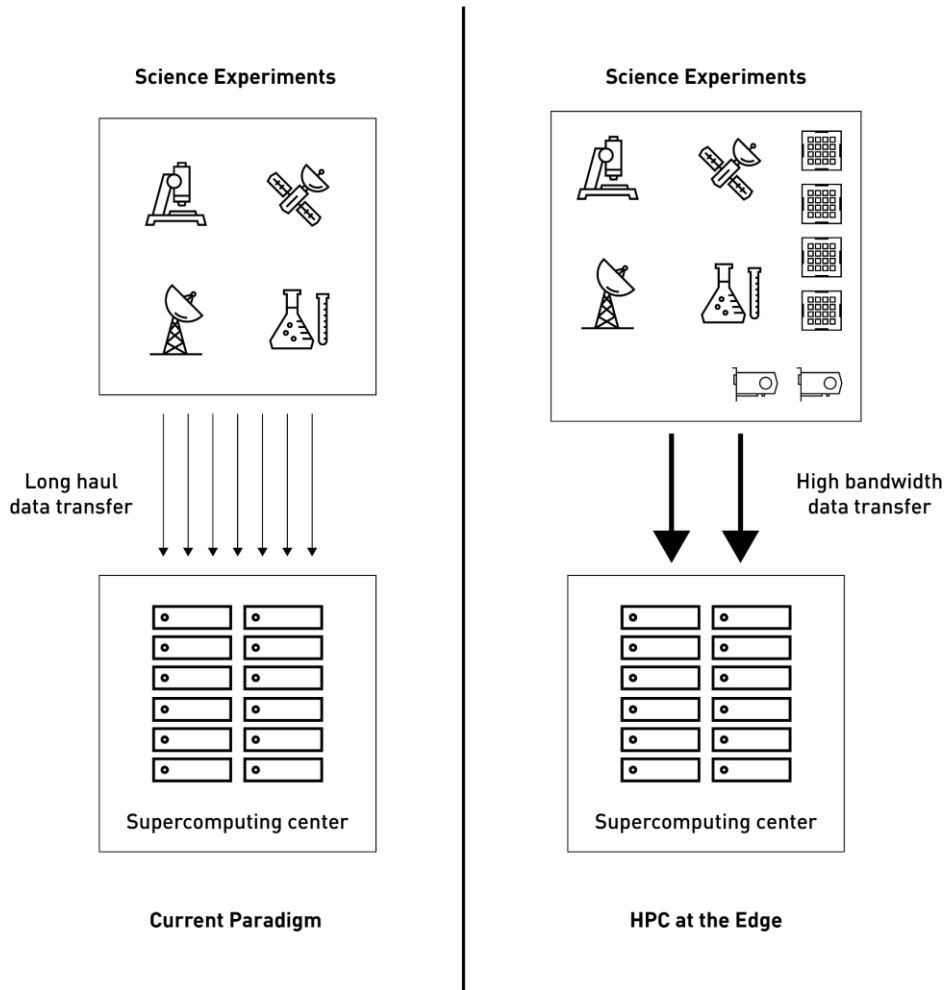
먼저, "에지 컴퓨팅"에서 "에지"는 무엇을 의미할까요? 에지 컴퓨팅은 데이터 소스와 인접한 위치에 컴퓨팅 엔진을 물리적으로 배치하는 것을 뜻합니다.

장점:

1. 데이터 소스에서의 전처리를 통해 일부 또는 대부분의 데이터를 추가로 전송 또는 처리할 필요가 없도록 필터링 효과를 제공
2. 데이터에 대한 응답 대기 시간 감소로 생산성 및 성과 향상
3. 소스에서 데이터를 처리함으로써 데이터 증가를 감당할 수 있는 확장성 확보
4. 중앙 시설을 업그레이드하는 대신 데이터 증가를 고려해 리소스 투자의 스케일링을 세분화할 수 있다는 점에서 개별 플랫폼의 비용 효과가 향상됨

에지 컴퓨팅은 실험과 이론을 통합하여 과학 실험의 새로운 패러다임을 가속합니다. 시뮬레이션 예측은 새로운 실험을 알리고 관찰을 개선할 수 있으며, 새로운 관찰 데이터는 모델을 신속하게 수정하여 시뮬레이션 모델을 개선할 수 있습니다. 경우에 따라 시뮬레이션은 디지털 방식으로 구동되는 실험 장치에 지시를 내려서 폐쇄 루프에서 데이터 수집을 최적화합니다. 이렇게 수집된 데이터를 시뮬레이션 대체 모델로 다시 투입해 더 많은 데이터를 조정 및 수집하여 정확도를 극대화하거나 불확실성을 최소화할 수 있습니다.

그림 6. 실험 데이터 처리 및 에지 HPC의 현재 및 향후 패러다임



하지만 대부분의 최첨단 과학 실험에서는 대체 모델을 사용할 때에도 실시간에 가깝게 데이터를 처리할 수 있도록 컴퓨팅 측면에서 데이터 집약적이고 응용 가능한 인프라가 필요합니다. 여기서 필요한 것이 바로 HPC입니다. 주요 연구 시설들이 엑사스케일 컴퓨팅 업그레이드를 통해 대규모 기기 업그레이드를 단행함에 따라 에지에 HPC를 통합한 슈퍼 컴퓨팅이 등장하고 있습니다. 딥 러닝 방식과 에지 처리 기술의 발전은 [APS-U\(Advanced Photon Source Upgrade\)](#), [ESRF-EBS\(European Synchrotron Research Facility Extremely Brilliant Source\)](#) 및 [PETRA-III](#)에서

업그레이드를 최대한 활용하고 컴퓨팅 문제를 해결하는 데 도움이 될 것으로 예상됩니다. 뿐만 아니라, HPC는 LHC(Large Hadron Collider), ATA(Allen Telescope Array), SKA(Square Kilometer Array) 같은 세계 최고의 과학 실험에서 인정받고 있습니다.

연속 데이터 스트림 처리, 머신 러닝 훈련 및 추론 툴, 기기 시각화 등을 결합해서 사용하는 방식으로 기기에서 전송된 데이터를 직렬로 수집하는 기존의 슈퍼 컴퓨팅 워크플로가 리팩토링되고 있습니다. 재설계된 이러한 워크플로는 현재의 과학적 요구에 적합한 규모인 동시에, 기기 업그레이드와 데이터 수집 속도 및 처리 요구 사항의 급증에 대비하고 있습니다.

양자 컴퓨팅이란 무엇인가?

양자 컴퓨팅은 중첩, 간섭 및 얽힘이라는 양자 역학 현상을 활용해 계산을 수행하는 기술입니다. 이 기술은 고전적인 비양자 컴퓨터보다 일부 문제를 더 빠르게 해결할 수 있는 것으로 알려져 있습니다.

몇몇 양자 알고리즘의 경우 대표적인 고전 알고리즘에 비해 양자 컴퓨터에서 스케일링이 더 손쉬운 것으로 알려져 있는데, 정수의 소인수를 찾는 Shor 알고리즘과 비정형 데이터 세트를 검색하기 위한 Grover 알고리즘이 여기에 해당합니다. 또한 양자 컴퓨터는 기존의 컴퓨터보다 대형 양자 시스템을 시뮬레이션하기에 적합하다고 알려져 있기 때문에 신약 개발과 에너지 등 많은 분야에서 응용 가치가 높습니다. 양자 컴퓨터의 응용 분야는 최적화와 머신 러닝 외에도 매우 다양합니다. 양자 컴퓨터의 이점을 누릴 수 있는 분야를 주제로 활발한 연구가 이루어지고 있습니다.

양자 컴퓨터를 몇 가지 서로 다른 사례에 물리적으로 구현하는 방법이 지난 20년에 걸쳐 크게 발전함에 따라 오늘날에 이르러 매우 구체적인 몇몇 태스크에서 고전적인 슈퍼 컴퓨터의 성능을 능가하는 양자 컴퓨터까지 등장하게 되었습니다. 그러나 이러한 태스크는 추상적이고 난해하며, 실제 애플리케이션의 태스크에서 고전 컴퓨터를 능가할 수 있는 양자 컴퓨터를 활용하려면 아직 갈 길이 멉니다.

한 가지 대표적인 문제로 스케일이 있습니다. 오늘날 양자 컴퓨터는 일반적으로 5-100큐비트 범위 내에 있는 반면, Shor 알고리즘을 풀거나 유용한 분자를 시뮬레이션하기 위해서는 수천 개 내지 수백만 개의 큐비트가 필요합니다. 여기에 충실도도 문제로 작용합니다. 양자 컴퓨터는 민감하고 아날로그적인 특성이 있기 때문에 본질적으로 오류가 발생하기 쉽고, 각 논리 연산에서 발생하는 오류는 양자 회로가 지속되는 동안 누적됩니다. 오늘날 양자 컴퓨터는 높은 오류율 때문에 일관성 있는 몇 가지 논리 연산만 실행하도록 기능이 제한되어 있습니다. 그래야 오류 누적에 의한 무작위 결과 생성을 방지할 수 있습니다. 알려진 알고리즘 가속 기능을 적용하기 위해서는 수천 개 내지 수백만 건의 작업을 실행해야 하기 때문에 이는 실용적이지 않습니다.

이렇듯 오늘날 양자 컴퓨터의 제한된 성능과 더불어 최고의 고전 알고리즘과 비교해 양자 알고리즘의 성능 수준을 이해하기 위한 집중 연구가 진행됨에 따라 강력한 시뮬레이션 툴은 이제 필수 요소가 되었습니다. 오늘날 연구자들은 오늘날 양자 컴퓨터의 수준을 능가하는 스케일과 성능으로 [새로운 양자 알고리즘을 개발, 테스트 및 디버그하기 위해 시뮬레이터](#)를 사용하고 있습니다.

고전 컴퓨터에서 양자 컴퓨터를 시뮬레이션하려면 엄청난 컴퓨팅 파워가 필요하지만, 다수의 하위 태스크가 병렬로 처리된다는 점에서 GPU에 적합한 방법입니다. 양자 컴퓨터를 효과적으로 시뮬레이션할 수 있는 능력은 미래의 양자 컴퓨터를 최대한 활용하는 방법을 이해하는 데 있어 새로운 인사이트와 획기적인 발전으로 이어질 것으로 기대됩니다. NVIDIA는 차세대 시뮬레이터를 지원 및 가속하기 위해 [cuQuantum 라이브러리](#)를 제공합니다.

양자 컴퓨터를 가치 있게 활용하려면 양자 컴퓨터가 고전 컴퓨터와 나란히 작동하면서 일부 작업을 수행하는 하이브리드 컴퓨팅 모델을 채택해야 한다는 주장이 널리 받아들여지고 있습니다. 오늘날, 선도적인 컴퓨팅 시설들은 하이브리드 데이터 센터를 구축하는 과정에서 미래의 양자 컴퓨터를 다른 HPC 리소스와 통합하는 방법을 모색 중입니다. 이러한 통합 시스템에는 통합 프로그래밍 모델과 소프트웨어 툴체인 필요합니다. 그래야 도메인 과학자가 원활하게 프로그래밍을 수행하고 양자 컴퓨터를 기존 워크플로에 투명하고 효과적인 방식으로 통합할 수 있습니다.

메타버스와 디지털 트윈의 부상

분석가와 언론 매체 모두가 "디지털 트윈"이라는 용어를 채택해 사용하고 있지만 그 정의는 다양합니다. 디지털 트윈을 객체의 디지털 모델을 지칭하는 또 다른 용어라고 생각하는 사람도 있고, 그 속성을 나타내는 용어로 생각하는 사람도 있습니다. 이는 장애물을 넘는 자전거가 될 수도 있고, [토카마크\(tokamak\)](#) 조건을 따르는 삼중수소 원자가 될 수도 있습니다. 일반적으로 디지털 트윈 모델은 동작을 예측하는 물리 시뮬레이션을 따릅니다. 하지만 여기서 문제가 되는 것은 디지털 트윈의 다음과 같은 속성입니다.

- 실시간으로 응답하지 않음
- 특정한 물리적 성질만 모델링함
- 센서의 데이터를 수집하거나 디지털 방식으로 나타내지 않음

그러나 [2세대 디지털 트윈](#)의 경우, 수치 시뮬레이션을 기반으로 연결 에셋에서 나온 운영 데이터를 모델 기반의 물리적 성질에 결합한다고 주장하는 사람들도 있습니다.

이러한 주장에 동의하기는 하지만 2세대 디지털 트윈으로 나아가기 위해서는 자극에 대한 모델 반응 예측이 크게 발전해야 할 것입니다. 물리학 PDE에 대한 근사치를 풀기 위한 수치적 방법은 가속 하드웨어에서 실행하더라도 결과를 실시간으로 생성할 만큼 빠르지는 않으며, 가까운 시일내에 이러한 성과를 달성하지도 못할 것입니다.

이 정도 수준에 도달하려면 수치 시뮬레이션을 실행하지 않고도 디지털 트윈에서 물리적 성질을 추정할 수 있는 기술이 필요합니다. [디지털 트윈의 백엔드에 AI 기술을 주입하면](#) 여러 가지 방법으로 디지털 트윈이 가치를 발휘할 수 있을 만큼 정확도를 충분히 유지하면서 필요한 속도를 달성할 수 있습니다.

보관되어 있는 분석 결과 및 이전 설계와 관련이 있는 시뮬레이터와 시뮬레이션 데이터는 AI 모델의 훈련 자료로 활용될 수 있으며, 이러한 자료는 다시 시뮬레이터 출력을 모방하고 실시간으로 디지털 트윈을 구동하는 데 사용될 수 있습니다. NVIDIA와 SIEMENS Energy는 [AI를 사용해 복합화력발전\(CCPP\)을 모델링하여 유사한 디지털 트윈 접근 방식을](#) 시연했습니다.

뒤에 나오는 "[방정식 향상 머신 러닝\(Equation Enhanced Machine Learning\)](#)" 섹션에는 AI가 훈련 과정에서 물리학 방정식을 제약 조건으로 사용해서 학습을 수행하는 새로운 기법이 소개되어 있습니다. 이렇게 하면 AI 모델이 성숙해짐에 따라 수행하는 예측 작업이 훈련 중에 사용된 물리 법칙을 따르게 됩니다.

NVIDIA는 30-60년 후의 날씨를 정확하게 예측하고 완화 전략을 모델링하고자 상세한 기후 모델링을 위한 완전한 [지구 디지털 트윈](#)을 구축한다고 전 세계에 발표했습니다. 기후 모델이 구름을 정확하게 렌더링하려면 지구 대기에 대한 메시가 1미터 스케일이어야 합니다. 현재의 컴퓨팅 가속화 속도로 볼 때 기존의 수치 모델에서 이 정도의 해상도로 모델을 실행하려면 앞으로 40년이 더 걸릴 것으로 예상됩니다. 따라서 디지털 트윈을 효과적으로 활용하려면 AI가 꼭 필요합니다.

NVIDIA에서는 데이터 및 물리학 기반의 머신 러닝 모델 구축을 지원하기 위해 Modulus라는 툴도 제공하고 있는데, 이 툴은 앞으로 "Earth-2" 디지털 트윈을 단기간에 실현하기 위한 핵심 기술이 될 것입니다.

HPC 핵심 요소

HPC 애플리케이션 지원 및 적용 범위

NVIDIA 플랫폼은 다음과 같이 다양한 수준에서 새로운 인사이트를 제공합니다.

- AI 프레임워크를 통한 인텔리전스 시뮬레이션
- 제1 원칙 알고리즘을 기반으로 하는 미시 및 거시 물리학 시뮬레이션
- 컴퓨터 그래픽을 통한 자연 현상의 시각적 표현 시뮬레이션

현재 재료 과학, 양자 화학, 지진, 날씨, 기후 등 광범위한 과학 영역에서 2,500개 이상의 [GPU 가속 애플리케이션](#)이 사용되고 있습니다. 그리고 NVIDIA Quantum [InfiniBand 기술](#) 덕분에 많은 HPC 시뮬레이터에서 중요한 역할을 하는 NVIDIA의 저지연/고대역폭 상호 연결을 활용하는 애플리케이션이 그 어느 때보다 많아졌습니다.

표 1. 분야별 HPC 애플리케이션







주: 녹색 텍스트는 GPU 가속 애플리케이션을 의미

유체 역학	물리학	미디어 및 엔터테인먼트	구조 역학
<ul style="list-style-type: none"> • Fluent • FUN3D • openFOAM • PowerFLOW • StarCCM+ 	<ul style="list-style-type: none"> • BQCD • Chroma • CPS • GENE • MILC • XGC 	<ul style="list-style-type: none"> • 3ds Max • Arnold • Blender • Maya • Renderman 	<ul style="list-style-type: none"> • ABAQUS • Ansys Mech. • CTS • LSDyna • NX Nastran
지진학	전자 설계	기후 및 날씨	우주론
<ul style="list-style-type: none"> • DELFI • ECLIPSE • Nexus • SpecFEM3D • tNavigator 	<ul style="list-style-type: none"> • Allegro • Clarity • Icepack • NCSim • VCS 	<ul style="list-style-type: none"> • GRAF • ICON • IFS • MPAS • WRF 	<ul style="list-style-type: none"> • CASTRO • Cholla • Gamer • HACC • RAMSES

금융 서비스	AI/ML/DL 데이터 과학	유전체학 및 단백질체학	양자 화학
<ul style="list-style-type: none"> 알고리즘학 Calypso MACS MOSES Pathwise TowersWatson 	<ul style="list-style-type: none"> MXNet PyTorch TensorFLOW 	<ul style="list-style-type: none"> GLIDE GPUBlast HMMER NCBI Blast openFOLD RosettaFOLD 	<ul style="list-style-type: none"> CP2K GAMESS Gaussian VASP NWChem NWChemEX
유체 역학		유체 역학	
<ul style="list-style-type: none"> AMBER CHARMM FEP+ GROMACS LAMMPS NAMD OpenMM 		<ul style="list-style-type: none"> DeepMDKit 	

시뮬레이션 및 시뮬레이션된 환경이 산업 전반을 혁신하기 위한 주력 분야로 빠르게 부상하고 있습니다. 표 2에는 NVIDIA에서 출시한 시뮬레이션 환경의 예시가 몇 가지 나와 있습니다.

표 2. 산업 전반의 NVIDIA 시뮬레이션 툴 및 환경

 Transportation	 Healthcare	 Commerce	 Manufacturing	 Robotics	 Quantum Computing
NVIDIA DRIVE	NVIDIA CLARA	NVIDIA MERLIN	NVIDIA OMNIVERSE	NVIDIA ISAAC	NVIDIA cuQUANTUM
End-to-end solutions for autonomous vehicles.	An application framework optimized for healthcare and life sciences developers.	An open-source framework for building high-performing recommender systems at scale.	A scalable, multi-GPU real-time reference development platform for 3D simulation and design collaboration.	An accelerated platform for robotics and AI.	An SDK of libraries and tools for accelerating quantum computing workflows

HPC 및 AI 컨버전스

인류 조상이 사용한 석기 도구에서부터 시공간의 물결을 측정할 수 있는 민감도 높은 기구에 이르기까지 기술은 하나의 발명이 그 다음 발명을 일으키는 복잡한 사다리 형태로 모델링될 수 있습니다.

따라서 새로운 도약을 위해 이질적으로 보이는 혁신 기술들을 결합해야 할 때도 있습니다. 오늘날 데이터 과학과 AI는 자동차 안전성과 금리 예측 등 다양한 영역에 대한 솔루션을 찾기 위해 기존의 HPC 기술을 융합하고 있습니다.

AI 기술의 성숙도가 빨라지면서 이전의 방법보다 빠르고 정확하게 결과를 생성할 수 있는 새로운 방식이 등장했습니다. 이와 동시에, 이렇게 강력한 기술을 적용할 수 있는 몇 가지 방법이 등장했는데, 이러한 방법들은 모두 가능한 한 빨리, 최대한 정확하게 사용자의 입력에 기반해 결과를 제공하는 것을 목표로 삼고 있습니다.

시뮬레이션 가속화

AI 모델을 구축하는 프로세스, 특히 훈련은 컴퓨팅 주도형 작업입니다. 다행스럽게도 컴퓨팅 속도를 최적화하는 것은 HPC 라는 명분에 가깝습니다. 하지만 안타깝게도 CPU에서는 이러한 최적화가 무어의 법칙에 따라 이루어졌으며 이제 이 법칙은 더는 유지되지 않는 것으로 보입니다.

그림 7. GPU 컴퓨팅, AI 및 ML 환경에서 구식이 되어버린 무어의 법칙

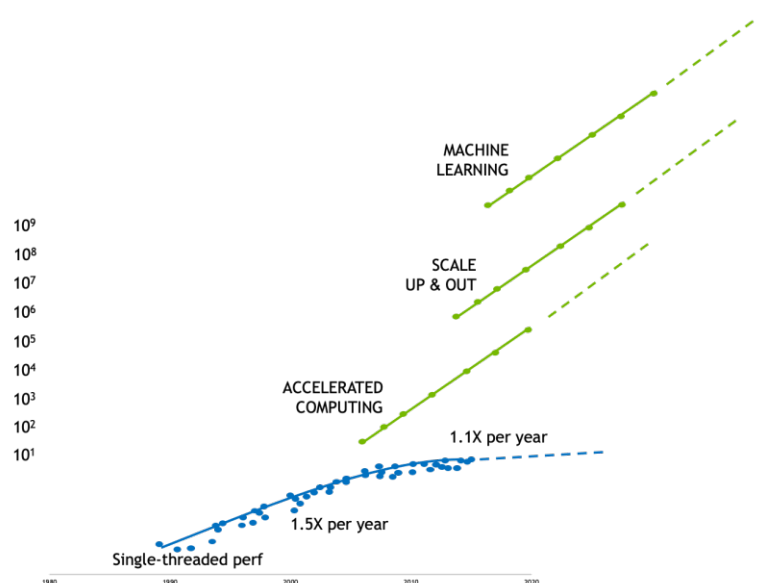


그림 7은 무어의 법칙 측면에서 CPU 트랙이 가속 컴퓨팅, AI 및 ML을 포함한 GPU 가속 컴퓨팅 기술의 가파른 기울기와 비교해 얼마나 쇠퇴한 상태인지를 보여줍니다. 무어의 법칙이 적용되는 CPU는 더 많은 기능을 통해 훨씬 가파른 성장 곡선을 보이는 GPU 가속 컴퓨팅 기술에 비해 불리합니다.

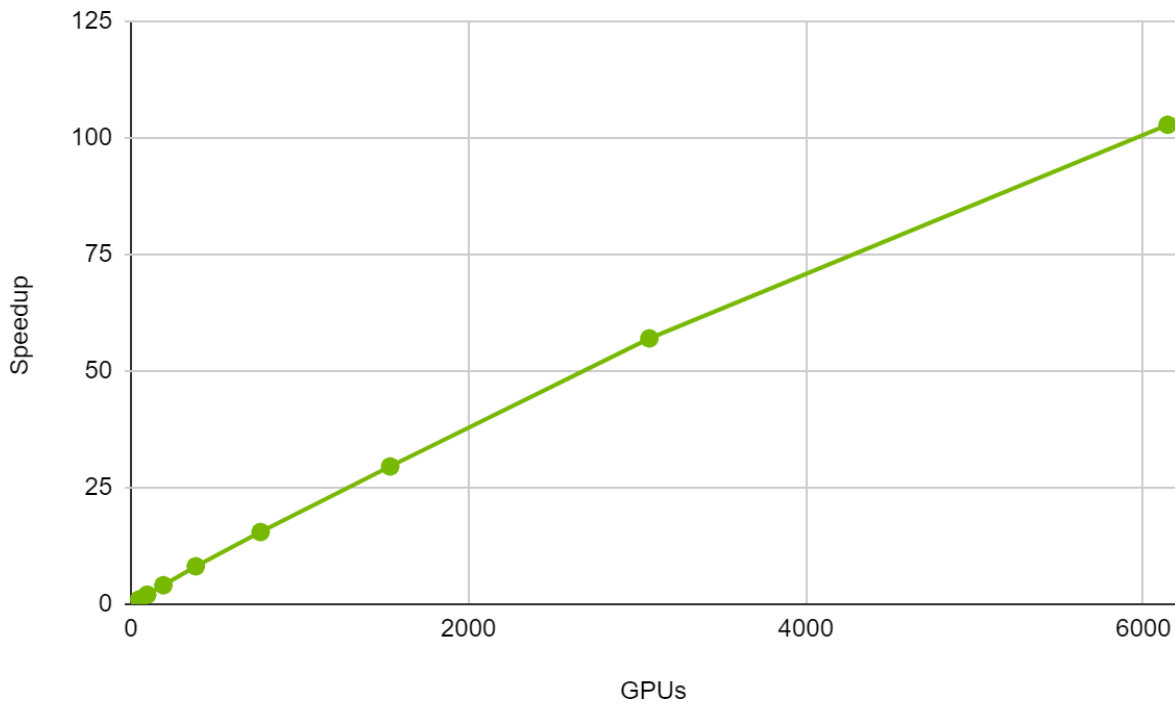
지난 2년 동안 AI와 HPC가 결합되면서 가속 컴퓨팅 하나만으로는 불가능했을 돌파구가 마련되었습니다. 일례로 2020년 미국과 중국의 연구원으로 구성된 팀이 "DeepMD-kit"라는 머신 러닝 모델과 LAMMPS라는 분자 역학 시뮬레이션 툴을 결합하여 **1억 개의 원자를 동시에 시뮬레이션**한 바 있습니다. 머신 러닝 모델은 원자 간 힘을 계산하는데, 이러한 힘은 표준 방법에서보다 1,000배 이상 빠르게 시스템 역학을 구동하기 때문에 시뮬레이션된 시스템의 크기를 10배 더 늘릴 수 있습니다.

그림 8에서 볼 수 있듯이, [Huerta](#)와 동료 연구원들은 궤도를 도는 한 쌍의 블랙홀에서 나오는 신호의 특성을 분석하기 위해 AI 모델을 구축해서 훈련했습니다. 이들 신호는 서로 나선형으로 휘어지면서 시공간 자체가 종처럼 울리게 되면서 발생합니다.

이 모델은 블랙홀이 병합되어 조용해지기 바로 직전에 생성된 중력과 신호를 설명하는 시계열 데이터를 통해 훈련되었습니다. 이러한 훈련 데이터는 과학적 측정을 통해 얻은 것이 아니라, 실험에 의한 측정을 모방해 시뮬레이션을 통해 생성된 데이터라는 점에서 HPC와 연관이 있습니다.

이러한 훈련 데이터를 생성하기 위해 대형 천체 물리학 시뮬레이션 모델이 필요했고, 매우 큰 HPC 시스템에서 스케일링되었습니다. 이로 인해 데이터를 생성하여 그에 따라 모델을 훈련하는 데 소요되는 시간이 대폭 줄었습니다. 4개의 CPU에서 24일이 걸리던 훈련이 1,536개의 GPU에서 1.2시간만에 완료된 것입니다. 그림 8은 기존의 HPC를 사용해 머신 러닝 모델의 훈련 데이터 생성을 스케일링하는 방법을 보여줍니다.

그림 8. 기존의 HPC를 사용한 훈련 데이터 가속



출처: [NSF 지원 사이버 인프라에서의 AI와 HPC의 융합 | Journal of Big Data | 전문\(springeropen.com\)에](#)
제시된 데이터와 그래프를 각색

시뮬레이션 작업을 수행하기 위해 머신 러닝 모델을 훈련하는 이러한 방법을 "대리 모델"이라고 하는데, 이는 기존의 HPC에 AI를 적용하는 여러 가지 방법 중 하나입니다. 위의 두 가지 예시는 HPC와 AI를 결합하여 개별 시뮬레이션을 더 빠르게 실행하는 방법을 보여줍니다. 아래 그림 9에서 스펙트럼의 끝부분 보라색이 바로 이 대리 모델입니다.

그림 9. HPC에 AI 적용

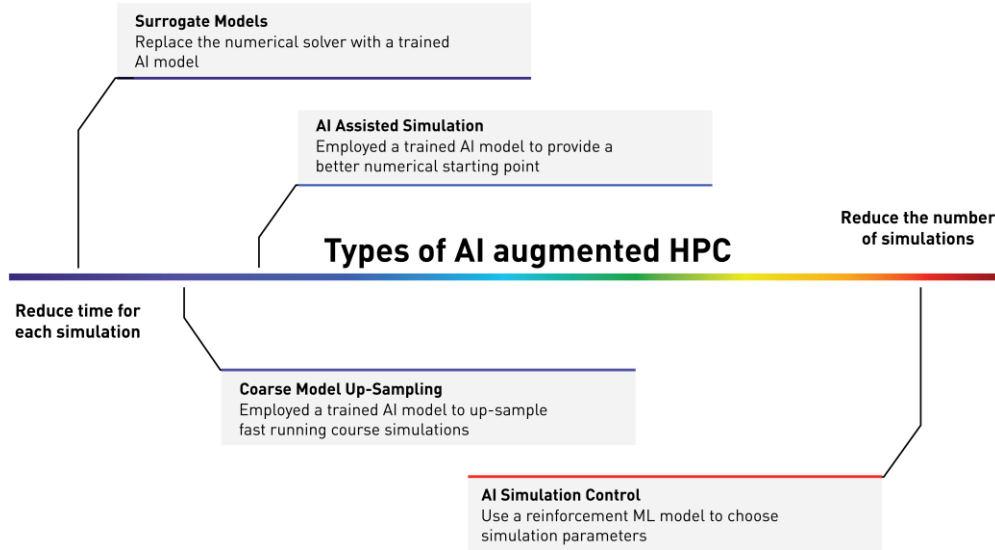


그림 9는 HPC에서 AI를 적용할 수 있는 방법을 스펙트럼으로 보여줍니다. 대부분의 방법이 단일 시뮬레이션의 가속화에 중점을 두고 있으며, 결합되었을 때 더 나은 사용자 서비스를 제공함으로써 HPC 환경에서 전반적으로 결과를 더 빠르게 도출하고 더 높은 처리량을 달성합니다. AI는 다양한 방식으로 기존의 HPC 워크플로에 적용될 수 있습니다. 개별 시뮬레이션을 가속하는 방법도 있고, 시뮬레이션 수를 최소화하는 방법도 있습니다.

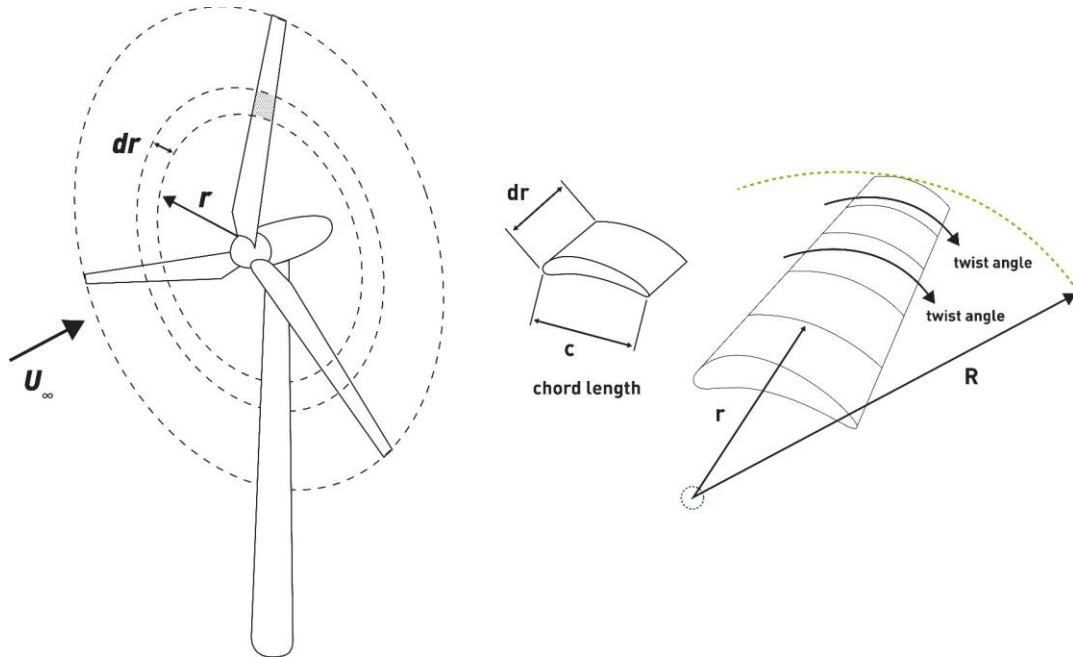
스펙트럼의 빨간색 부분은 시뮬레이션 가속보다 효율성에 더 중점을 둔 영역을 나타냅니다. HPC 사용자가 다중 매개변수 최적화라는 문제에 직면해 있을 때는 효율성을 고려해야 합니다. 특정 매개변수 세트가 시뮬레이션 결과에서 더 나아질 것인지 혹은 나빠질 것인지를 판단하는 방법이 있기는 하지만, 허용 가능한 설계 공간에서 이러한 매개변수의 값을 탐색하고 최적화하기 위해 수백 번 내지 수천 번의 시뮬레이션을 수행해야 할 때 이런 문제가 발생합니다. 이러한 문제를 일컬어 "실험 설계"(DOE)라고 하는데, 여기서 실험은 시뮬레이션과 같은 말입니다.

DOE는 시뮬레이션이 제품 설계 툴로 사용되는 상용 HPC에서 흔히 수행되는데, 대표적으로 항공우주, 신약 개발, 자동차, 전자 설계 등의 산업이 있습니다. HPC에 AI를 적용하는 방법을 보다 구체적으로 설명하기 위해 다음과 같은 예를 들어보겠습니다.

어떤 엔지니어가 풍력 터빈을 설계한다고 해봅시다. 이때 아래 그림 10에서와 같이 많은 매개변수가 존재하는데, 이들은 블레이드 개수, 직경, 에어포일, 비틀림, 루트 코드, 팁 코드, 각 블레이드의 테이퍼 등에서 터빈을 설명합니다. 풍력 터빈의 일반 매개변수화는 설계 공간 탐색의 한 예로, HPC에 AI를 적용할 수 있는 또 다른 기회이기도 합니다.

뿐만 아니라, 최대 팁 속도, 비용, 재료 가용성 및 구조적 한계에서부터 타워 모멘트에 이르기까지 설계 선택을 제한하는 제약 조건들이 많이 있습니다. 엔지니어는 고도에 따른 풍향 및 속도에 대한 통계 모델을 토대로 시뮬레이터를 이용해 허브의 터빈에서 생성된 토크를 계산할 수 있으며, 이 토크에 의해 전력 출력이 결정됩니다.

그림 10. 풍력 터빈 - HPC에 AI를 적용할 수 있는 기회



엔지니어가 주로 하는 작업 중 하나는 터빈을 설명하는 매개변수의 최적 값을 선택해서 안전 마진 내에서 전력 출력을 극대화하는 동시에 비용을 최소화하는 것입니다. 이 값을 선택하기 위해 엔지니어는 수백 번의 시뮬레이션을 수행해서 설계 공간을 탐색하고 최적 값의 근사치를 계산하게 됩니다. AI를 사용하면 시뮬레이션 자체에 관여하지 않으면서도 최적의 설계 세트에 도달하기 위한 시뮬레이션 횟수를 최소화할 수 있습니다.

[Astra Zeneca의 자동화된 신약 신속 개발 시스템](#)에서부터 전면 [3D 프린팅 AI 설계](#) 자동차에 이르기까지 많은 현장에서 이러한 방법이 사용되고 있습니다.

방정식 항상 머신 러닝

DNN(Deep Neural Networks)이나 CNN(Convolutional Neural Networks)과 같은 대부분의 머신 러닝 모델은 정형 데이터에서 조합된 특정 예제의 형태로 주요 입력 데이터를 수신합니다. 이러한 예제들은 큐레이션을 거쳐 레이블을 지정한 후 훈련 과정 시 AI 모델에 투입해야 합니다. 이러한 방식의 훈련에서 사용 가능한 정확도로 모델을 훈련시키기 위해서는

보통 수십만 개 내지 수백만 개의 예제가 필요하기 때문에 프로세스 속도가 느려지고 많은 인력이 투입될 수 있습니다.

이러한 유형의 방식에서는 경우에 따라 훈련된 모델이 생성되는데, 이러한 모델은 훈련 세트에서는 잘 작동하지만 새로운 데이터에 직면했을 경우에는 일반화된 규칙에 부합하지 않는 결과가 도출될 수 있습니다.

대표적인 예로 유체 흐름에 대한 AI 모델 생성을 들 수 있는데, 이 모델은 주어진 기하학적 조건을 토대로 유량 내에서 속도와 방향 및 압력을 예측합니다. 정성적 수준에서 유선과 압력이 올바르게 보이더라도, AI에서 생성된 솔루션에서 유동장(flow field)의 질량 또는 에너지 보존과 같은 기본 원리를 조사하면 예측이 이렇게 간단한 물리 법칙을 위반하는 것으로 밝혀져서 결과를 신뢰할 수 없게 됩니다.

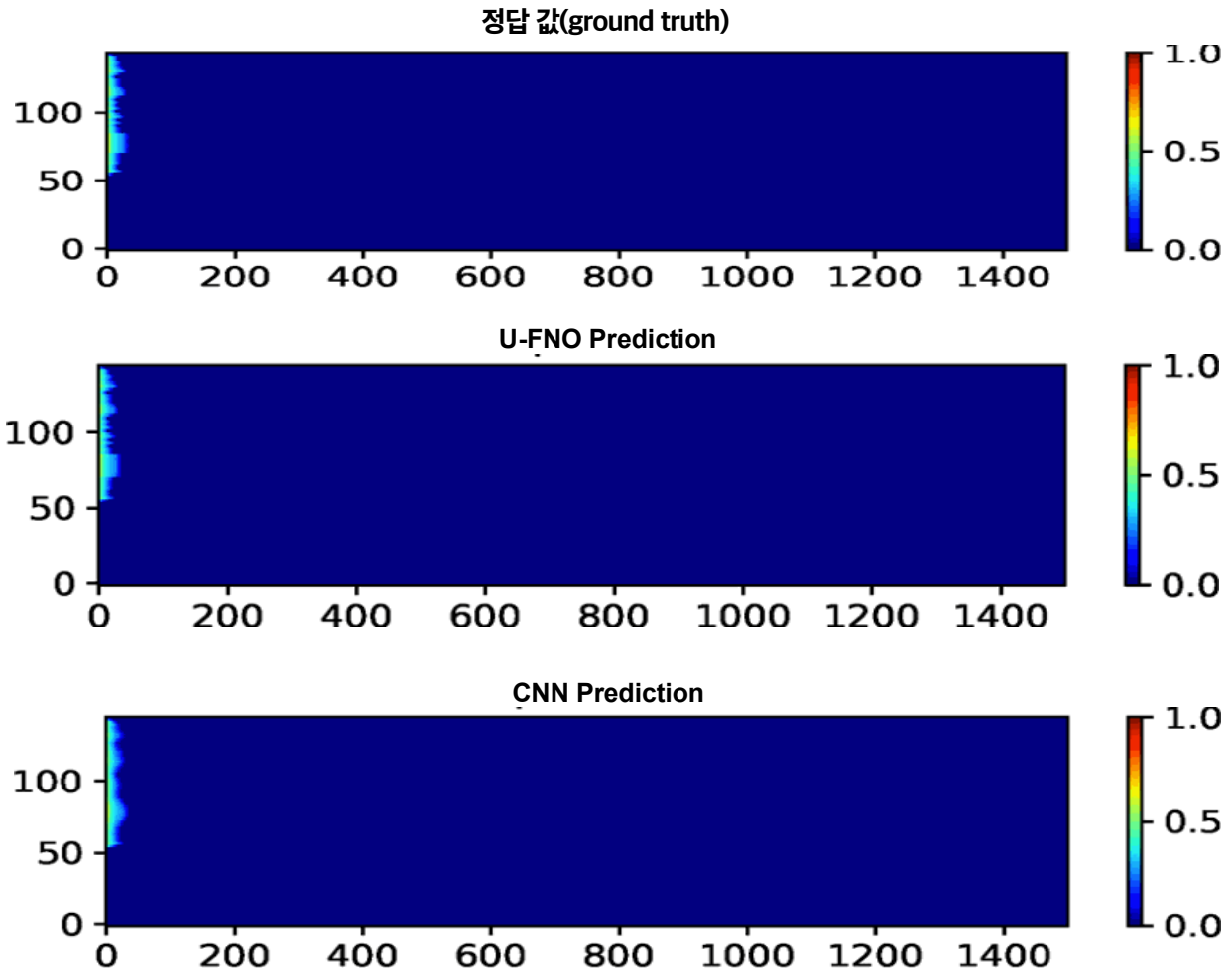
펜실베니아 대학의 Paris Perdikaris 부교수를 포함해 독창적인 연구자들은 훈련 중 손실 함수에서 방정식(이 경우 질량과 운동량의 보존을 설명하는 물리학 방정식)을 사용하는 대체 방법을 제안했습니다. 이러한 접근 방식을 보여주는 몇 가지 예제가 있습니다.

- PINN(Physics Informed Neural Network)
- FNO(Fourier Neural Operator) 또는 U-FNO(Unet modified Fourier Neural Operator)

U-Net 수정 FNO의 경우, 스탠포드 대학의 Gege Wen 박사 후보생과 공동 연구자들은 '지하 배출 탄소 처리' 사례로 시간 경과에 따른 CO₂ 포화 사례에서 이러한 정확도 차이를 입증했습니다. 이 연구에서 이들은 기존의 PDE 솔버 접근 방식과 CNN(Convolutional Neural Network) 방식을 이 모델과 비교했습니다.

그림 11은 참조 모델(PDE 기반 수치 솔루션), Unet 수정 FNO 및 CNN에서 30년에 걸쳐 예측된 이산화탄소 포화도의 차이를 보여줍니다. 여기에서 그림 11의 애니메이션 버전을 확인해 보세요.

그림 11. 정답 값(ground truth) 및 U-net 수정 FNO 예측과 비교한 CNN 예측



HPC 혁신

[트랜스포머 모델](#)은 문장의 단어나 DNA 시퀀스의 염기쌍과 같은 순차 데이터의 관계를 추적하여 컨텍스트와 그에 따른 의미를 학습하는 신경망입니다. Google의 [2017년 논문](#)에서 처음 소개된 트랜스포머 모델은 현재까지 발명된 모델 중 가장 최신이자 가장 강력한 모델 클래스 중 하나입니다.

트랜스포머 모델은 영향력 측면에서 가깝거나 먼 관계를 명시적으로 검색하고 학습한다는 점에서 다른 신경망들과 다릅니다. 이러한 이유로 레이블이 지정되지 않은 데이터를 사용할 수 있습니다. 다른 머신 러닝 알고리즘들은 구축 및 큐레이션 속도가 느린 레이블 지정 데이터를 사용하기 때문에 보통 가용 훈련 데이터의 양에 따라 정확도가 좌우됩니다. 따라서 트랜스포머 모델은 사람의 개입을 줄이면서도 훨씬 더 큰 데이터 세트를 사용할 수 있습니다. 이러한 이유로 연구원들이 가장 선호하는 모델이 되었습니다. ArXiv가 실시한 짧은 설문 조사에 따르면, 지난 2년 동안 과학 및 엔지니어링 부문에 게재된 전체 논문 초록의 70%에서 트랜스포머 모델이 언급된 것으로 나타났습니다.

트랜스포머 모델에 대한 새로운 응용 및 사용 사례는 아주 많습니다. 컴퓨터 기반 신약 개발, 기후 모델링 및 단백질 접힘 등의 분야에서 일종의 트랜스포머 모델을 사용하고 있습니다. 표 2에는 분야별 트랜스포머 모델에 대한 링크 목록이 포함되어 있습니다.

표 2. 분야별 트랜스포머 모델

Discipline	Product
Computer aided drug discovery (CADD)	MegaMolBart
기후 모델링	FourCastNet
단백질 접힘	AlphaFold2
	RoseTTAFold
	OpenFold2

코히어런트(coherent) 시스템과 가속기 메모리

메모리 일관성은 다중 요소 프로세서의 각 처리 요소에 해당하는 메모리 위치에서 모든 메모리 풀에 메모리 최고 속도로 액세스하기에 바람직한 조건입니다.

여기서 초고속 가속기가 전속력으로 실행되려면 데이터에도 그만큼 빠른 속도로 액세스할 수 있어야 하는데, 이러한 이유 때문에 고대역폭 메모리(HBM)를 GPU에 바로 장착하는 것입니다. 물론, 충분히 큰 메모리란 가속기 메모리보다 항상 더 클 것이기 때문에 가속기는 주 CPU 메모리에서 데이터를 가져올 수밖에 없게 됩니다. 경우에 따라 계산해야 할 데이터의 양이 많을 경우, 일종의 "JIT(Just In Time)" 컴퓨팅 파이프라인에서 계산이 이루어지는 동안 이러한 데이터 저장 및 가져오기가 숨겨질 수 있습니다. 하지만 이 기능이 항상 가능한 것은 아닙니다. 대부분의 경우에는 입출력(I/O)을 기다리는 동안 가속기 계산이 유휴 상태가 됩니다.

이를 위한 해결책은 대표적인 가속기 메모리 크기의 10~20배에 달하는 풋프린트에서 가속기가 주 메모리에 최고 속도로 액세스할 수 있는 아키텍처를 고려하는 것입니다. 이렇게 가속기가 CPU의 피어인 아키텍처를 "코히어런트"라고 합니다.

다음과 같이 몇 가지 표준이 새롭게 등장했습니다.

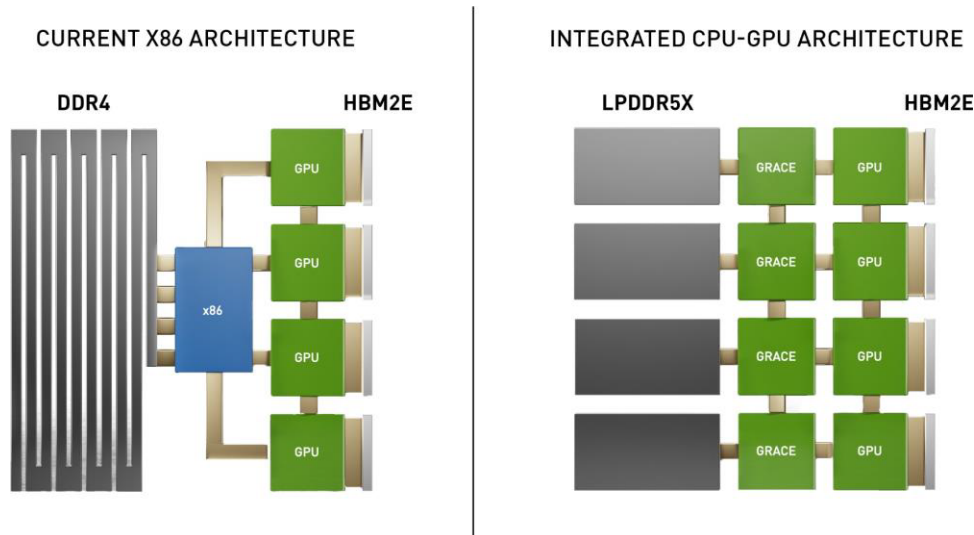
- [openCAPI - 코히어런트 가속기 프로세서 인터페이스](#)
- [CCIX\(Cache Coherence Interconnect for Accelerators\)](#)
- [이기종 시스템 아키텍처 기반](#)

2021년 3월, NVIDIA는 "Grace"라는 Arm 기반 아키텍처를 발표했는데, 이는 첫 출시 당시 함께 개발된 [최초의 슈퍼 컴퓨터 ALPS](#)와 더불어 메모리 코히어런트 플랫폼입니다. AMD 역시 ORNL(Oak Ridge National Laboratory) Frontier와 Lumi에서 사용할 [메모리 코히어런트 아키텍처 Trento](#)를 출시했습니다. [NVIDIA가 클라이언트 선정을 위해 제공하고 있는 포팅 플랫폼](#)에 대한 자세한 정보를 확인하실 수 있습니다.

이러한 새 아키텍처와 시스템은 성능과 기능이 크게 향상될 것이기 때문에 HPC 시스템이 메모리 대역폭에 구속을 받는 모든 조사 영역에서 발전하는 계기가 될 것으로 예상됩니다. 뿐만 아니라, 머신 러닝 기술이 HPC 애플리케이션에 적용되고 활용에 앞서 ML 모델을 훈련해야 하는 요구 사항으로 인해 메모리 일관성과 같은 기술들도 새로운 기회를 창출할 것으로 기대됩니다.

그림 12는 현재 x86 패러다임 아키텍처와 NVIDIA의 코히어런트 아키텍처인 Grace를 비교한 것입니다.

그림 12. 현재 x86 아키텍처와 NVIDIA의 코히어런트 Grace 아키텍처 비교



컴퓨터 네트워크의 발전

스케일링을 위한 네트워크 대역폭

HPC 시스템 설계의 일반적인 관행은 보통 메시지 전달과 파일 시스템 데이터 흐름에 사용되는 단일 고성능 네트워크를 사용하는 것입니다. 2021년 Top500에 오른 10대 슈퍼 컴퓨터 중 일부는 서버 노드당 여러 개의 네트워크 인터페이스 어댑터를 사용하고 있는데, 예를 들어 Summit 슈퍼 컴퓨터는 EDR(Enhanced Data Rate) 100GB/s InfiniBand가 2개이고 NVIDIA Selene는 HDR(High Dynamic Range) 200Gb/s InfiniBand가 8개입니다. 이러한 대표 시스템들은 프로비저닝, 모니터링 및 기타 유용한 저대역 활동에 사용되는 다른 신호 및 관리 네트워크를 가지고 있는 경우가 많은데, 이러한 관리 네트워크는 처리 중인 데이터를 전달하는 네트워크에 초점을 맞추고 있기에 이 전자책에서는 다루지 않습니다.

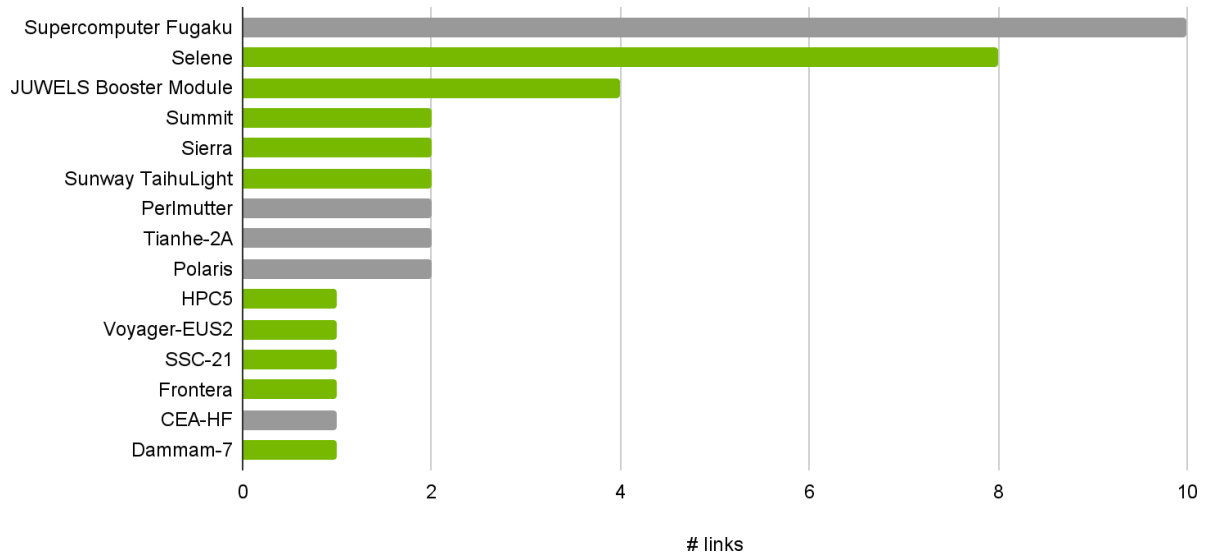
공급업체가 링크당 대역폭이나 노드당 링크 수를 반드시 공유할 필요는 없기 때문에 Top500에 오른 15대 슈퍼 컴퓨터에 대해 노드당 대역폭을 비교하기가 어렵습니다. 그러나 일부 공급업체들은 이러한 정보를 자발적으로 공유하고 있습니다.

데이터 사용량이 많은 컴퓨터와 딥 러닝 훈련이 HPC 시스템에서 실행되는 대표적인 워크로드 중 하나가 됨에 따라, 통신과 데이터 이동에 중점을 두고 훈련 성능을 스케일링하는 것이 더욱 중요해졌습니다. 가속 컴퓨팅으로 패러다임이 전환되면서 가속기당 약 200GB/s의 대역폭이 추가로 필요해졌습니다. 더 빠른 가속기가 출시됨에 따라 데이터를 지속적으로 공급하는 데 필요한 대역폭이 증가할 것이고, 이에 필요한 병렬 연결의 수도 비례하여 증가하게 될 것입니다. 즉, 상호 연결 대역폭 역시 꾸준히 증가하면서 가속기의 세대 간 병렬 연결의 수가 현재 억제되고 있습니다.

존스 홉킨스 대학과 AWS(Amazon Web Services)가 최근에 실시한 연구에서 이러한 현상이 나타나 있는데, 분산 훈련은 높은 네트워크 대역폭의 이점을 얻을 수 있고, 네트워크가 완전히 활용될 경우 스케일 팩터가 100%가까이 향상될 수 있다는 결과가 나왔습니다.

그림 13에는 노드당 네트워크 연결 수에 따라 정렬된 Top500 슈퍼 컴퓨터의 20대 컴퓨터 중 일부가 나와 있습니다. 그림에 나와 있는 슈퍼 컴퓨터별 노드당 네트워크 링크 수는 순위 순이 아닙니다. 녹색은 NVIDIA InfiniBand 네트워크를 나타내고, 회색은 다른 네트워크 유형을 나타냅니다.

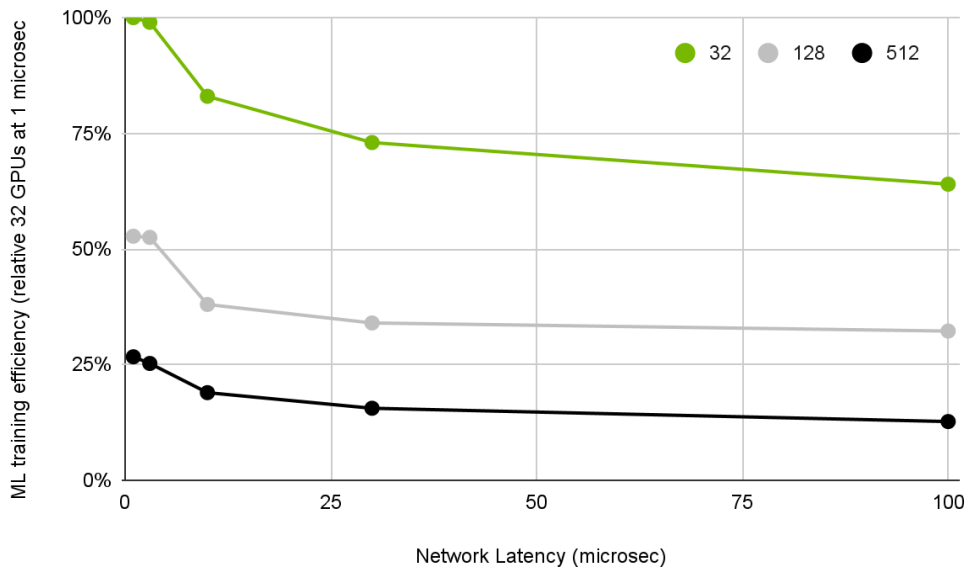
그림 13. 15대 슈퍼 컴퓨터에 대한 상호 연결 링크의 막대 그래프



최근 MIT(Massachusetts Institute of Technology)는 트랜스포머 모델인 ResNet50과 Megatron에서의 ML 훈련 성능에 [네트워크 성능\(지연 시간 및 대역폭 모두\)이 미치는 영향](#)에 대한 연구를 수행했습니다. 예상했던 대로 연구 결과, 대부분의 모델에서 GPU당 네트워크 대역폭이 증가함에 따라 동일한 수준의 정확도로 모델을 훈련하는 데 소요되는 시간이 증가한 것으로 나타났습니다.

이와 더불어, 네트워크 지연 시간을 살펴보고 네트워크 지연 시간이 증가함에 따라 ML 훈련 스케일링 효율성에 미치는 영향을 입증했습니다. 결과는 흥미로웠는데, GPU 개수가 적을 때보다 많을 때, 지연 시간에 의한 스케일링이 2배 더 심각하게 영향을 받는 것으로 나타났습니다. 그림 14에는 GPU 개수가 서로 다른 경우에 네트워크 지연 시간의 함수로서의 ML 훈련 스케일링 효율성이 나와 있습니다.

그림 14. 네트워크 지연 시간의 함수로서의 ML 훈련 스케일링 효율성



다중 GPU 및 다중 노드 가속을 위한 상호 연결

대규모의 언어 및 딥 러닝 추천 모델과 이들이 사용하는 신경망은 보다 수준 높은 인사이트를 제공하기 위해 더 큰 데이터 세트를 사용하게 되면서 그 크기와 복잡성이 커지고 있습니다. 그 결과, 활용이 가능한 시간 내에 이들 네트워크를 훈련시키기 위해 컴퓨팅 및 메모리 용량 요구 사항이 기하급수적으로 증가하고 있습니다. 다중 GPU 시스템은 이러한 문제를 해결할 수 있도록 선형에 가까운 성능 스케일링을 보여주었습니다. 그리고 아키텍처에서 이러한 다중 GPU 시스템의 성능 스케일링을 지원하는 핵심 요소는 바로 탄력적이고 유연한 고대역폭 GPU 간 통신입니다.

[NVIDIA NVLink](#)는 점대점 네트워크에서 GPU를 함께 연결하는 솔루션을 제공하며, 8개의 GPU 서버로 성능을 높였습니다. NVLink 기술은 모델이 온칩 GPU 메모리 크기를 초과하는 사용 사례를 지원하고, PCIe(Peripheral Component Interconnect Express) Gen5의 7배에 달하는 대역폭에서 고속 I/O(900GB/s)로 GPU 상호 연결을 지원합니다. [NVLink와 NVtags](#)를 사용해 HPC 애플리케이션의 성능을 최대 75% 높이는 방법을 확인해보세요.

하나의 NVLink(그림 15 참조)는 GPU당 18개의 NVLink 포트를 지원하며, 각 포트는 25GB/s의 속도로 작동합니다. 완전한 NVLink 점대점 연결을 위해서는 18개의 포트를 모두 동원해서 900GB/s의 속도로 2개의 GPU를 연결해야 합니다.

그림 15. 18개 포트를 탑재한 NVLink

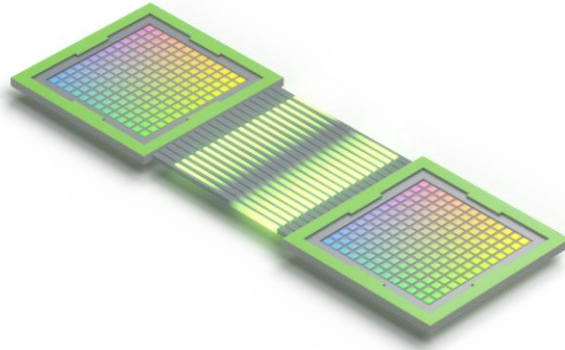
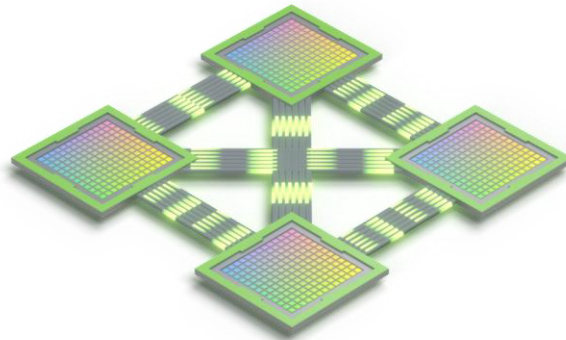


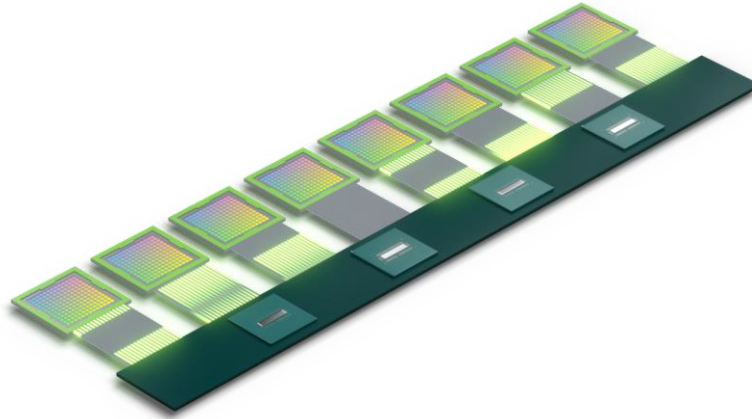
그림 16에는 다른 3개의 GPU 각각에 대해 GPU당 6개 포트를 사용하여 2개의 GPU 간에 300GB/s의 속도를 제공하는 4방향 연결 방식이 나와 있습니다.

그림 16. 4개의 GPU를 상호 연결하기 위한 NVLink 구성



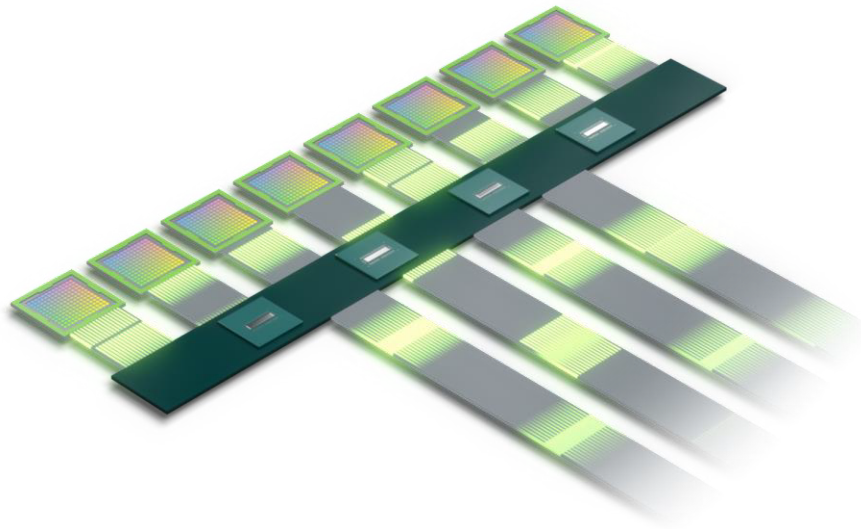
하지만 모든 GPU가 GPU 간 NVLink 대역폭을 낮추지 않고 서로 통신을 수행해야 하는 All-to-All 통신의 경우, 900GB/s의 속도를 지원하려면 NVSwitch와 같은 고급 스위칭 솔루션을 통해 GPU 서버의 성능을 높여야 합니다. 그림 17에 나와 있듯이, 각 NVSwitch는 900GB/s의 속도로 모든 GPU를 연결할 수 있도록 64개의 NVLink 포트를 제공합니다.

그림 17. 4개의 NVSwitch 장치를 사용해 8개의 GPU를 상호 연결



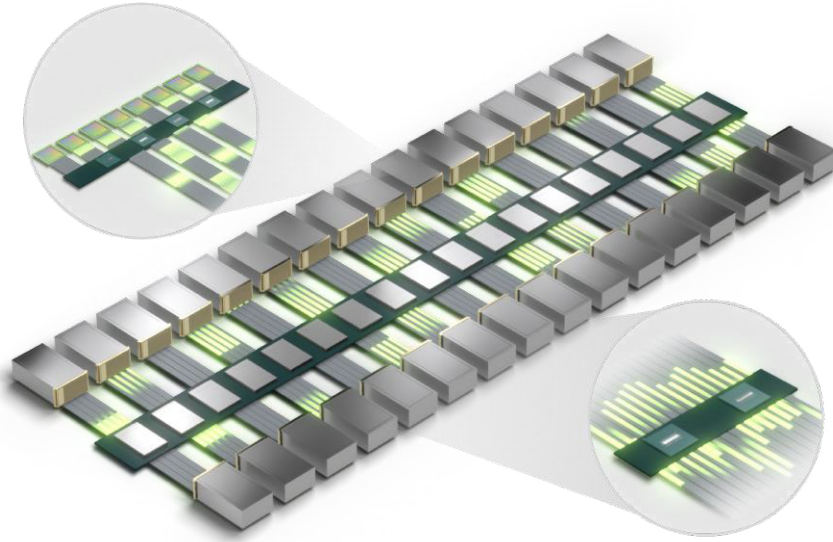
NVSwitch는 이러한 문제를 해결하고 모든 GPU에서 NVLink가 완벽하게 연결된 시스템을 제공함으로써 900GB/s의 최고 속도로 양방향 연결을 제공합니다. NVSwitch에 8~16개의 GPU가 연결된 서버 아키텍처에서는 GPU 메모리에 빠르게 액세스하여 640~1,280GB의 속도를 지원할 수 있습니다. 그림 18은 4세대 NVLink 기술이 외부 연결을 추가하여 노드 간 고성능 연결을 확장하는 방법을 보여줍니다.

그림 18. 4세대 NVLink 및 NVSwitch 구성



NVSwitch와 NVLink를 사용하면 크기가 더 큰 GPU 클러스터를 생성하여 주소 지정이 가능한 메모리 공간을 확장할 수 있습니다. 그림 19에는 하나의 NVLink 도메인에 256개의 GPU를 배치하여 총 20TB의 GPU 메모리를 제공하는 18개 NVLink 스위치의 32개 서버 구성이 나와 있습니다.

그림 19. 18개 NVLink 스위치의 32개 서버 구성



NVLink 스위치 시스템은 NVSwitch 기반 연결의 두 번째 티어를 추가하여 팻 트리(fat-tree) 네트워크 토폴로지를 생성합니다. NVSwitches의 첫 번째 레이어는 각 서버 노드 내부에 배치되고, 두 번째 레이어는 외부 랙 스위치인 NVLink 스위치에 배치되어 All-to-All 통신을 위해 모든 GPU를 연결합니다.

외부 NVLink 스위치와 NVLink 스위치 시스템을 사용하면 NVLink 상호 연결을 단일 서버의 물리적 경계를 넘어 더 큰 [NVIDIA DGX POD](#)로 확장할 수 있습니다. 이러한 구성에서는 하나의 NVLink 도메인에서 모든 GPU가 완전히 상호 연결됩니다. 멀티캐스트를 가속하고 [SHARP\(Scalable Hierarchical Aggregation and Reduction Protocol\)](#)를 통해 네트워크 내 축소를 수행하는 방식으로 3세대 NVSwitch에 집합적 연산을 위한 추가 최적화가 도입되었습니다.

3D FFT에서 6배 성능 향상과 "Mixture of Experts" 머신 러닝 기술에서 9배 빠른 훈련 처리량이 예상되는 데에는 GPU I/O 상호 연결 속도와 주소 지정이 가능한 GPU 메모리를 확장한 것도 한 요인으로 작용합니다. 이러한 성능 향상은 A100에서 H100으로 전환한 덕분이지만, HDR 200Gb/s InfiniBand에서 NDR 400Gb/s InfiniBand 및 NVLink 스위치 시스템으로 이동할 때 더 빠른 GPU 상호 연결 속도를 활용한 덕분이기도 합니다.

효율적인 통신을 통해 병렬 처리의 효율성을 최적화하고, 고대역폭 네트워크와 다수의 GPU 노드에서 조정 오버헤드를 최소화하는 방법은 여기 나온 게시물 [NVSHMEM을 통한 과학 컴퓨팅 스케일링](#)을 참조하시기 바랍니다.

클라우드 네이티브 슈퍼 컴퓨팅

활용 효율성을 달성하면서 슈퍼 컴퓨팅 시스템으로부터 최대한의 성능을 도출하는 것은 본래부터 최신 클라우드 컴퓨팅의 안전한 멀티테넌트 아키텍처와 양립되지 않는 것이었습니다. [클라우드 네이티브 슈퍼 컴퓨팅](#) 플랫폼은 업계 최초로 [보안 격리 및 멀티테넌시를 위한 최신 제로 트러스트 모델에 최고 성능과 클러스터 효율성을 결합하여](#) 두 기술의 장점을 제공합니다.

AI와 HPC가 광범위한 상용 사례에서 주요 컴퓨팅 환경으로 자리잡으면서 이제 슈퍼 컴퓨터는 광범위한 사용자에게 서비스를 제공하고, 보다 다양한 소프트웨어 에코시스템을 구축하여 논스톱 서비스를 동적으로 제공해야 하는 상황이 되었습니다. 새로운 슈퍼 컴퓨터는 멀티테넌시 환경에서 베어메탈 성능을 제공하도록 설계되어야 합니다. 슈퍼 컴퓨터 설계는 가장 중요한 임무인 최고 성능과 최저 오버헤드에 초점을 맞추고 있습니다.

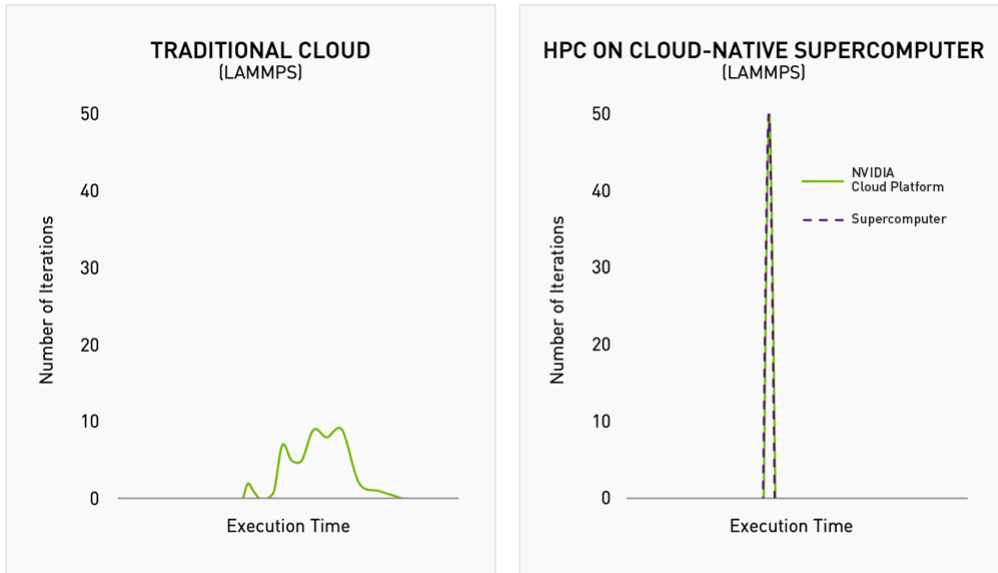
최소 권한 보안 정책 및 격리, 데이터 보호, 즉각적인 온디맨드 AI, HPC 서비스 등 클라우드 서비스 요구 사항을 충족하면서 이러한 성능 특성을 유지하는 것이 클라우드 네이티브 슈퍼 컴퓨터 아키텍처의 목표입니다. 클라우드 네이티브 슈퍼 컴퓨터 아키텍처는 영리 회사와 학계 및 정부 기관을 포함한 개방형 커뮤니티를 기반으로 개발됩니다.

멀티테넌트 슈퍼 컴퓨팅에서는 공유 인프라에서 다수의 사용자 애플리케이션이 실행되기 때문에 이러한 애플리케이션에서 생성되는 물리적 서버, 스토리지, 네트워크 및 I/O 트래픽 패턴을 중복 사용할 가능성이 있습니다. NVIDIA는 슈퍼 컴퓨터 데이터 센터의 컴퓨팅 리소스를 공유하기 위해 슈퍼 컴퓨팅에서 베어메탈 성능과 멀티테넌시 및 성능 격리를 결합한 클라우드 네이티브 슈퍼 컴퓨팅 아키텍처를 도입했습니다. 이 아키텍처는 NVIDIA Quantum-2 스위치 제품군, BlueField-3 DPU 및 ConnectX-7 네트워크 어댑터가 포함된 NVIDIA Quantum-2 InfiniBand 플랫폼의 한 부분입니다.

NVIDIA Quantum-2 InfiniBand는 네트워크 혼잡이 감지되면 이를 관리하고, 소스에서 혼잡을 줄이기 위해 제어를 실행합니다. 그러나 멀티테넌시에서는 사용자 애플리케이션이 인접 애플리케이션 트래픽과의 무차별적 간섭을 인식하지 못할 수 있기 때문에 예상 성능 수준을 제공하기 위해서는 격리가 필요합니다. Quantum-2 InfiniBand 스위칭 시스템에서는 사전 예방적 모니터링 및 혼잡 관리 기능이 필요한 트래픽 격리를 수행합니다. 이를 통해 성능 지터를 거의 없애고, 애플리케이션이 마치 전용 시스템에서 실행 중인 것처럼 예상 예측 성능을 보장합니다.

기존의 클라우드 환경에서는 LAMMPS가 여러 번 실행되기 때문에 I/O 지연이 발생하고, 이로 인해 실행 시간이 달라지게 됩니다. 혼잡 제어 및 격리 기능은 전용 슈퍼 컴퓨터 시스템에서 격리된 상태로 애플리케이션을 실행할 때처럼 좁은 범위에서 결정적이고 일관된 실행 시간을 제공합니다.

그림 20. 클라우드 네이티브 슈퍼 컴퓨터에서 기존의 클라우드의 실행 시간을 HPC와 비교



과학 컴퓨팅 및 AI 애플리케이션은 All-Reduce나 All-to-All과 같은 MPI 집합 연산을 광범위하게 사용합니다. 이러한 집합 연산을 호스트 CPU에서 스위치 네트워크로 오프로드하기 위해 NVIDIA Quantum InfiniBand 스위치에서 네트워크 내 컴퓨팅을 위한 SHARP가 처음으로 도입되었습니다. 이렇게 하면 엔드포인트 간에 데이터를 여러 번 전송할 필요가 없고, 집계 노드에 도달할 때 네트워크를 통과하는 데이터의 양이 줄어듭니다. 따라서 MPI 연산 시간이 대폭 단축됩니다.

NVIDIA Quantum-2 InfiniBand 스위치는 이전의 Quantum 스위치보다 32배 더 많은 AI 엔진을 제공하여 클라우드 데이터 센터 네트워크를 통한 대규모 데이터 집계의 멀티테넌트 확장성을 위해 SHARP를 동시 사용할 수 있도록 지원합니다. 스위치당 소형 메시지가 무한대로 감소하고 몇 개의 대형 메시지가 감소하는 흐름이 지원되기 때문에 공유 시스템에서 애플리케이션을 실행하는 멀티테넌트는 All-Reduce와 같은 복잡한 MPI 집합 연산에서 두 배 높은 성능을 활용할 수 있으며, SHARP를 사용할 때 클러스터 크기에 관계없이 일관되게 짧은 지연 시간을 경험할 수 있습니다.

["NVIDIA SHARP를 통한 네트워크 내 컴퓨팅" 동영상을 시청하세요.](#)

클라우드 네이티브 슈퍼 컴퓨팅을 지원하는 또 다른 핵심 요소는 바로 DPU(Data Processing Unit)인 NVIDIA BlueField입니다. 완전히 통합된 데이터 센터 온-칩 플랫폼인 BlueField는 호스트 프로세서 대신에 데이터 센터 인프라를 오프로드 및 관리하여 슈퍼 컴퓨터의 보안과 오케스트레이션을 지원합니다.

SmartNIC 오프로드

클라우드 서비스 공급자(CSP)가 실행하는 기술들은 미래의 데이터 센터를 향해 발전하고 있습니다. 이러한 기술들은 하이퍼바이저나 I/O와 같이 가치가 낮은 작업을 오프로드하여 서버 투자를 극대화하는 동시에, 실행 인스턴스 간의 하드웨어 수준 장벽을 유지하는 방식으로 보안을 강화할 수 있는 잠재력을 갖고 있습니다. 스마트 인터페이스 네트워크 카드(smartNIC)를 사용하면 성능, 보안 및 스토리지를 모두 획기적으로 개선할 수 있습니다. 비록 오늘날 대부분의 상용 SmartNIC 제품이 관념적이라는 점이 이러한 애플리케이션의 가장 큰 장애물이긴 하지만 말입니다.

이에 AWS의 [NITRO](#) 기술은 독점 하이퍼바이저에서 시작해 이후에 확장을 하는 방식으로 데이터 센터에서 광범위하게 서비스 오프로드를 사용하는 방법을 개척했습니다. 그 후 업계가 이러한 개념의 성공에 주목하였고 일부 하드웨어 공급업체가 네트워크 인터페이스 에지에서 서버 측 오프로드 프로세서를 제공한다는 계획을 발표했습니다.

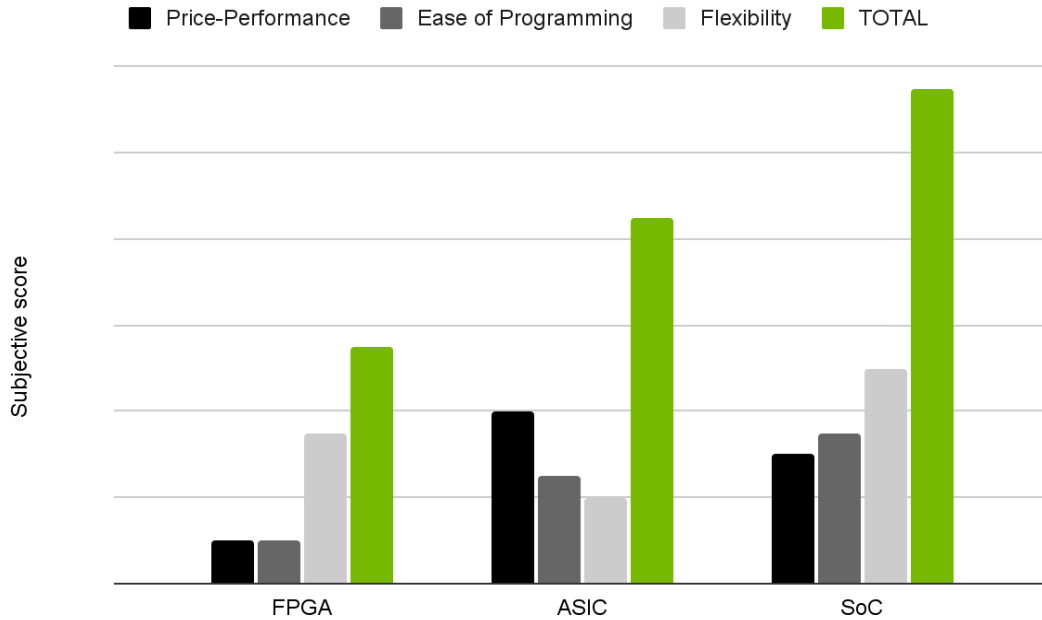
이는 서버 안팎으로 전달되는 데이터를 ASIC, FPGA 또는 SoC 같은 일종의 프로세서를 통해 우회시킬 수 있음을 뜻합니다. 새롭게 등장한 Pensando와 Fungible을 비롯해 Broadcom, Xilinx, Intel, NVIDIA 같은 유명 공급업체들이 관련 계획과 제품을 발표했습니다.

그렇다면 이 모든 것은 HPC와 어떤 관련이 있나요? 좋은 질문입니다.

현재로서는 HPC에서 사용되는 애플리케이션이 다른 경우보다 일종의 SmartNIC가 장착된 시스템에서 더 빠르게 실행된다는 것은 여전히 이론적인 주장에 불과합니다. 그러나 보안 또는 I/O 측면에서 항상 효과가 있을 것이라는 데에는 이견이 없습니다.

한 발 물러서서, 그림 21은 가성비, 프로그래밍 가능성, 유연성의 세 가지 관점에서 SmartNIC 프로세서 유형을 평가한 결과를 보여줍니다. 이것은 정성적이고 주관적인 평가이지만, 이러한 관점을 취한 데에는 그만한 이유가 있습니다. 자세한 내용은 [최적의 SmartNic 선택](#)에 관한 이 게시물을 참조하시기 바랍니다.

그림 21. 스마트 NIC 기술 비교



코히어런트 가속기와 네트워크 인터페이스 카드

컨버지드 가속기와 NIC란 무엇인가?

GPU는 AI, ML 및 HPC 애플리케이션을 가속하지만, GPU 안팎의 I/O는 전반적인 애플리케이션 성능에 주된 병목이 될 수 있습니다. 최신 PCIe 상호 연결을 활용하는 단일 컨버지드 카드에서 스마트 NIC의 고급 기능을 GPU 가속과 결합한 결과, 컨버지드 가속기 NIC라는 균형 잡힌 아키텍처가 탄생했습니다.

PCIe는 I/O 및 기타 기능에 대해 타사 추가 옵션을 광범위하게 지원한다는 점에서 x86 표준 및 HPC 서버에 사용되고 있습니다. 다중 GPU가 필요한 시스템에서 컨버지드 가속기 카드는 GPU 대 NIC에 최적의 1:1 비율을 적용하여 서버의 자체 x86 PCIe 버스에서 경합을 방지하므로 추가 장치를 통해 성능을 선형으로 스케일링할 수 있습니다. 이렇게 하면 CPU 호스트를 통과하는 데이터의 병목을 제거하고, GPUDirect RDMA 및 GPUDirect Storage 기술 (NVIDIA Magnum I/O 데이터 센터 I/O 가속 소프트웨어의 구성 요소)을 사용해 GPU-GPU 통신 및 GPU-원격 NVMe 스토리지에서 예측 가능하고 낮은 지연 시간을 보장할 수 있습니다.

이러한 컨버지드 설계는 AI 훈련과 같은 GPU 기반의 I/O 집약형 워크로드에 최적의 성능을 제공합니다. 이러한 아키텍처의 대표적인 예로 NVIDIA H100 GPU와 PCIe Gen5 스위치가 통합된 ConnectX-7 SmartNIC를 들 수 있습니다. 이러한 접근 방식은 메인스트림 PCIe Gen4나, 심지어 Gen3 서버를 활용해서도 특수 제작된 하이엔드 호스트 시스템에서나 가능한 성능 수준을 달성할 수 있으며, 단일 카드를 사용하기 때문에 전력, 공간 및 PCIe 장치 슬롯을 절약할 수 있다는 이점이 있습니다. UCX(Unified Communication - X) 프레임워크, UCX 및 NVIDIA Collective Communications Library, NCCL 같은 매그넘 I/O 소프트웨어 가속 라이브러리는 GPU로의 데이터 전송을 위해 최고 성능을 발휘하는 데이터 경로를 자동으로 사용합니다. 따라서 수정 작업 없이도 기존의 다중 노드 가속 애플리케이션에서 고성능과 확장성을 누릴 수 있습니다.

HPC 개발자 에코시스템

활발한 커뮤니티

HPC 기술을 평가할 때 장기적인 성공을 나타내는 대표적인 지표 중 하나는 유사한 작업을 수행 중인 활성 사용자 커뮤니티와 참여 공급업체입니다. x86 아키텍처 같이 일반적으로 사용되는 기술들은 거대한 사용자 커뮤니티와 수많은 툴을 보유하고 있기 때문에 해당 아키텍처를 기반으로 솔루션을 제공하는 잠재적 공급업체들의 관심을 끌 필요성이 비교적 적습니다. 반면, 커뮤니티 규모가 작은 맞춤형 기술이 유사한 수준의 성공을 달성하려면 고객별로 상당한 투자가 필요합니다.

[NVIDIA HPC 커뮤니티](#)는 활기차고, 활발하며, 성장을 거듭하고 있습니다. 다양한 애플리케이션 도메인 및 산업 분야에서 NVIDIA 기술(하드웨어, 소프트웨어, 시스템 및 플랫폼)을 활용하고 있는 개발자, 연구원, 과학자 및 엔지니어의 성공과 생산성은 과학적 혁신을 창출하기 위한 열쇠입니다.

HPC 커뮤니티의 고유한 요구 사항을 이해한 다음, 기술적 인게이지먼트, 전문성 및 비즈니스 모델의 수준이 서로 다른 광범위한 개발자 지원 서비스를 통해 이러한 요구를 충족할 때 이러한 성공을 거둘 수 있습니다. 표 3에는 커뮤니티 참여 및 학습을 위한 NVIDIA의 광범위한 지원 리소스 목록이 나와 있습니다.

표 3. 커뮤니티 참여 및 학습을 위한 NVIDIA 지원 리소스

무료 서비스 및 툴	
NVIDIA 개발자 포럼	개발자 포럼
	버그 보고 및 추적
	사전 훈련된 AI 모델
	항상 요청 컴파일러 및 라이브러리
온디맨드 교육 콘텐츠	NVIDIA GTC, YouTube
NVIDIA 문서 센터	docs.nvidia.com
GPU 해커톤 및 부트캠프	gpuhackathons.org
오픈소스 콘텐츠	github.com/NVIDIA
코드 샘플	github.com/nvidia/cuda-samples
전문 서비스	
훈련 과정	딥러닝 연구소
전문가 지원	컨테이너 지원
	컴파일러
컨설팅	AI 서비스 제공 파트너

NVIDIA는 훈련 기회, 기술 문서, SDK(Software Development Kit), 코드 샘플, 네트워킹 기회, 동료 및 전문가 지원 등을 포함해 NVIDIA 커뮤니티에 풍부한 리소스를 제공하고 있습니다.

표준, 개방형 및 포터블 병렬 프로그래밍 모델

HPC 업계에는 대상 하드웨어 플랫폼 전반에 걸쳐 다양한 수준의 지원 서비스가 제공되는 수많은 소프트웨어 개발 틀 및 프로그래밍 모델이 출시되어 있습니다. ISV(Independent Software Vendor), HW/SW 플랫폼 공급자, 오픈 소스 등에서 제공하는 옵션을 사용할 수 있습니다. 애플리케이션이 제대로 작동하고 있고 만족스럽게 수행되고 있더라도 개발자는 개방형과 비교해 폐쇄형 소프트웨어 스택의 장단점을 비롯해 생산성 및 이식성과 관련된 문제에 대해서도 고민해야 합니다.

CPU와 가속기, 심지어 DPU까지 혼합되면서 현재 HPC 환경은 혁신적이지만 복잡한 아키텍처를 제공하고 있습니다. 이러한 시스템에는 개발 작업을 조정해 서로 다른 하드웨어 요소의 동급 최고 기능을 활용할 수 있도록 지원하는 정교한 소프트웨어 솔루션이 필요합니다. 사용자는 이러한 시스템 요소를 손쉽게 혼합하고, 동종 CPU 전용 시스템이든 이종 CPU/GPU 또는 CPU/GPU/DPU 구성이든 관계 없이 솔루션을 효율적으로 프로그래밍할 수 있어야 합니다.

코딩 담당자는 성능 최적화와 개발 간소화의 균형을 도모하는 동시에, 벤더 종속성을 피해야 합니다. 시스템의 서로 다른 구성 요소를 처리하기 위해 여러 공급업체가 제공하는 소프트웨어 솔루션을 취향대로 이것저것 사용할 수도 있지만, 이는 그다지 생산적인 방법이 아닙니다. 소프트웨어 엔지니어는 소프트웨어 스택 구성 요소를 결합하는 것이 아닌, 시스템의 가치를 높이는 데 시간을 투자해야 합니다. 사용자가 향후 장치 구성 요소의 다양화를 원할 수 있기 때문에 공급업체 전용 툴 세트에 의해 제약을 받아서는 안 됩니다.

단일 프로그래밍 모델만으로는 오늘날 코딩 담당자의 모든 요구 사항을 해결하지 못할 가능성이 높습니다. 사용자가 모든 목표를 달성하려면 개방형 표준과 몇 가지 적절한 모델의 하이브리드 통합을 지원하는 개발 환경을 사용하는 것이 가장 좋습니다. HPC 애플리케이션을 위한 최적의 가속 컴퓨팅 프로그래밍 모델을 식별할 때 사용자가 작업에 적합한 도구를 자유롭게 선택할 수 있도록 [NVIDIA는 종합적인 SDK를 제공하고 있습니다.](#)

이 소프트웨어 스택은 CPU, GPU 및 네트워크를 포함한 전체 NVIDIA 플랫폼의 프로그래밍을 지원합니다. 개발 과정에서 플랫폼에 특화된 결정을 내리든, 표준 언어를 활용하든, 그 중간에서 일종의 하이브리드 코딩 방법을 사용하든 관계 없이 개발 스펙트럼 전반에 걸쳐 유연한 옵션을 제공함으로써 사용자가 애플리케이션을 최대한 활용할 수 있게 돕는 것이 목표입니다.

그림 23에는 NVIDIA HPC SDK에서 사용할 수 있는 4가지 주요 프로그래밍 모델 유형이 나와 있습니다.

- 가속 표준 언어
- 증분식 포터블 최적화
- 플랫폼 특화
- 가속 라이브러리

프로그래밍 옵션의 범위는 플랫폼 특화 및 지시문 기반 옵션에서부터 표준 접근 방식 및 성능 최적화 라이브러리에 이르기까지 매우 다양합니다

그림 23. NVIDIA HPC SDK에서 사용 가능한 주요 프로그래밍 모델 유형

ACCELERATED STANDARD LANGUAGES ISO C++, ISO Fortran	INCREMENTAL PORTABLE OPTIMIZATION OpenACC, OpenMP	PLATFORM SPECIALIZATION CUDA			
<pre>std::transform(par, x, x+n, y, y, [=] (float x, float y) { return y + a*x; }); do concurrent (i = 1:n) y(i) = y(i) + a*x(i) enddo import cunumeric as np ... def saxpy(a, x, y): y[:] += a*x</pre>	<pre>#pragma acc data copy(x,y) { ... std::transform(par, x, x+n, y, y, [=] (float x, float y) { return y + a*x; }); ... } #pragma omp target data map(x,y) { ... std::transform(par, x, x+n, y, y, [=] (float x, float y) { return y + a*x; }); ... }</pre>	<pre><u>global</u> void saxpy(int n, float a, float *x, float *y) { int i = blockIdx.x*blockDim.x + threadIdx.x; if (i < n) y[i] += a*x[i]; } int main(void) { ... cudaMemcpy(d_x, x, ...); cudaMemcpy(d_y, y, ...); saxpy<<<(N+255)/256,256>>>(...); cudaMemcpy(y, d_y, ...); }</pre>			
ACCELERATION LIBRARIES					
Core	Math	Communication	Data Analytics	AI	Quantum

플랫폼 특화에 투자할 의사나 능력이 없는 개발자에게 [NVIDIA는 병렬 처리 및 가속화를 제공](#)하고 있는데, 이러한 SDK는 매우 빠른 개발 속도와 높은 생산성을 제공함으로써 해당 프로세스를 원활하게 만들어줍니다.

HPC 개발자들은 대체로 도메인 과학자인데, 이들의 최대 과제는 과학적 인사이트의 도출 시간을 단축하는 것입니다. 계산 과학에서 결과를 얻으려면 두 가지가 필요한데, 하나는 런타임 성능이고 다른 하나는 개발 속도입니다. 해당 프로세스가 최대한 자동으로, 원활하게 진행되도록 하고, 이를 달성하기 위해 두 가지 전략을 권장하는 것이 목표입니다.

런타임 성능은 NVIDIA 성능에 최적화된 라이브러리 (코어, 수학, 네트워크용 통신, 데이터 분석, AI 및 딥 러닝용)와 Python 및 양자 컴퓨팅 회로 시뮬레이션용 라이브러리를 통해 달성할 수 있습니다. 모든 라이브러리는 이러한 사용 사례에 맞춤화되어 있습니다.

라이브러리를 사용할 때 가장 큰 장점은 API와 아키텍처를 수정할 수 있다는 것입니다. 그런 다음, NVIDIA의 전문가가 드롭인 사용 편의성, 플랫폼 특화 성능 및 하드웨어 세대를 초월한 지원을 위해 라이브러리를 최적화합니다. 성능 최적화가 된 라이브러리는 사용자 생산성과 애플리케이션 성능 간에 최적의 균형을 제공합니다.

사전 검증된 라이브러리를 사용하면 개발 속도를 달성할 수 있습니다. 프로그래머는 GPU 가속의 성능 이점을 활용해 HPC 애플리케이션을 직접적으로 빠르게 개선할 수 있습니다. 라이브러리 덕분에 개발자는 모든 구현 세부 사항에 대한 전문가가 될 필요 없이 하드웨어 기능을 활용할 수 있습니다. 최적화된 라이브러리의 경우, Tensor Core, 향상된 L2 캐시 또는 공유 메모리 같은 강력한 GPU 기능을 손쉽게 활용할 수 있게 해줍니다.

과학적 HPC만 하더라도 사용 사례가 무수히 많습니다. 특수 라이브러리를 사용할 수 없는 경우에는 표준, 특수 및 지시문 기반 프로그래밍 접근 방식을 사용하는 방법을 고려해야 합니다.

NVIDIA는 자동 GPU 가속을 지원하기 위해 ISO 표준, C/C++ 및 Fortran 지원, 그리고 비 ISO Python까지 제공합니다. 이들 언어는 멀티코어 CPU에서 자동으로 병렬 처리가 가능합니다. 표준 언어에서 지원되는 새로운 병렬 구조의 장점은 코드가 더 간결하고 깔끔해졌을 뿐만 아니라, 읽기 및 유지 관리가 쉽다는 것입니다.

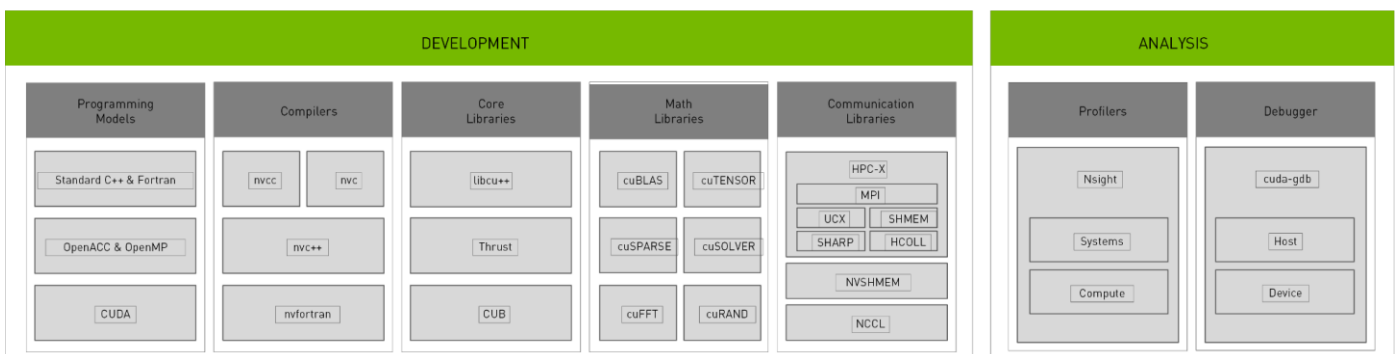
또한 ISO 모델을 따르면 강력한 이식성의 이점을 누릴 수 있습니다. 현재 나와 있는 경쟁사의 소프트웨어 스택들과 달리, 기존 NVIDIA 컴파일러는 CPU, GPU 또는 두 가지 모두로 이루어진 이종 시스템에서 표준 언어 애플리케이션 코드를 실행할 수 있습니다.

자세한 내용이 궁금하시면 표준 언어 사용의 이점에 대한 NVIDIA 게시물과 [C++](#) 및 [Fortran](#) 프로그래밍 지원에 대한 게시물을 읽어보시기 바랍니다.

NVIDIA는 표준화된 언어 접근 방식을 지원하기 위해서는 먼저 하드웨어 기능의 혁신적인 발전이 있어야 한다는 것을 잘 알고 있습니다. 그래야 개발자가 OpenACC나 OpenMP 같은 방법을 활용한 지시문 기반 프로그래밍 방법을 통해 점진적으로 이식성을 최적화할 수 있습니다.

하드웨어 기능의 발전과 성능 향상의 이점을 즉각적이면서도 온전하게 누리고 싶어하는 선도적인 사용자를 위해 개발자는 NVIDIA GPU 및 CPU를 대상으로 하는 특수 [CUDA 프로그래밍](#)을 항상 활용할 수 있습니다. 그림 24에 나와 있듯이, NVIDIA HPC-SDK는 몇 가지 구성 요소로 이루어져 있고 이들은 개발과 분석의 두 가지 단계로 나누어집니다.

그림 24. 개발 및 분석을 위한 NVIDIA HPC-SDK 지원



이러한 HPC 개발자 환경은 CPU, GPU 및 네트워크를 포함해 NVIDIA 플랫폼을 프로그래밍하기 위한 컴파일러 및 툴이 종합된 단일 컬렉션으로, 선호도에 따라 다양하게 이용할 수 있습니다. NVIDIA [Developer Zone 웹 포털](#)에서 제공되는 기존의 설치 프로그램을 사용할 수도 있고, Spack을 통해 제공되는 NGC 사이트 외부의 컨테이너를 사용할 수도 있으며, AWS, Azure 같은 주요 CSP 마켓플레이스에서 액세스할 수도 있습니다.

이러한 SDK는 앞서 설명한 모든 주요 프로그래밍 모델(표준 C/C++ 또는 Fortran, Python, OpenACC 및 OpenMP 같은 지시문 기반 프로그래밍, CUDA를 통해 궁극적으로 성능을 최적화하기 위한 특수 접근 방식)을 지원합니다. 이러한 지원은 SDK의 컴파일러 세트를 통해 제공됩니다.

- nvcc: NVIDIA CUDA 컴파일러
- nvc/nvc++: NVIDIA C/C++ 컴파일러
- nvfortran: NVIDIA Fortran 컴파일러

이러한 컴파일러들은 자동 GPU 가속을 제공할 뿐만 아니라, 병렬식으로 벡터화된 고성능 코드 지원 아키텍처를 통해 x86, OpenPOWER 및 Arm 서버를 포함하여 멀티코어 CPU에 대한 자동 병렬 처리를 제공합니다.

HPC SDK는 다음과 같이 광범위하고 다양한 [GPU 가속 라이브러리](#)를 제공합니다.

- 코어 라이브러리: [Thrust](#), [CUB](#) 및 [libc++](#)
- 수학 라이브러리: [cuBLAS](#), [cuTENSOR](#), [cuSPARSE](#), [cuSOLVER](#), [cuFFT](#) 및 [cuRAND](#)
- 통신 라이브러리: Magnum IO (HPC-X, [NVSHMEM](#), 및 [NCCL](#))

물론, HPC 소프트웨어 개발 키트에는 다음과 같이 디버깅 및 프로파일링 지원에 필요한 분석 툴도 포함되어 있습니다.

- 디버거: [CUDA-GDB](#)
- 프로파일러: [Nsight Systems](#) 및 [Nsight Compute](#)

이러한 분석 툴은 코드 기능을 수정하고 검증하는 데 도움이 되며, 성능 최적화 및 조정이 가능하도록 애플리케이션 실행의 병목에 대한 인사이트를 제공합니다.

앞서 언급했듯이, [NVIDIA HPC SDK는 무료로 다운로드](#)할 수 있으며, 매년 여러 차례 업데이트됩니다.

다양한 벤치마크 가속 예제를 포함해 HPC SDK를 설명하는 주문형 동영상을 보려면 개발자 프로그램에 참여하고, [NVIDIA On-Demand](#) 포털을 방문해 [최신 HPC 소프트웨어 심층 분석\(A Deep Dive into the Latest HPC Software\)](#)를 시청하시기 바랍니다.

발표자는 C/C++, Fortran, 지시문 같은 프로그래밍 모델을 위한 다양한 옵션을 단계별로 설명합니다. 그런 다음, 표준 언어 구성을 활용해서 생산성을 높이고 CPU 전용 플랫폼이 아닌 하이브리드 GPU 시스템을 활용해 궁극적인 성능 이점을 누리는 등 단 한 줄의 코드 변경도 없이 아키텍처를 리타겟팅하는 방법을 보여줍니다.

이 동영상에는 13배 빨라진 Lulesh C/C++ 코드 예제와 프로그래밍 방법을 통해 실행 속도가 7배나 획기적으로 향상 (플랫폼에 새롭게 도입된 NVIDIA A100 GPU까지 고려하면 거의 60배나 속도가 향상)된 MAIA 코드에 대한 내용이 포함되어 있습니다. 이와 달리, Fortran(Do Concurrent 프로그래밍 루프 구성) 가속 예제에는 계산 화학 코드인 NWChem과 GAMESS가 포함되어 있습니다.

클라우드, VM, 컨테이너

HPC는 수십 년에 걸쳐 기술의 다각화와 통합이라는 주기를 거치고 있습니다. 통합은 획기적인 신기술이 시장에 진입해서 새로운 표준으로 자리 잡을 때 일어납니다. 이러한 획기적인 신기술은 단일 또는 소수의 공급업체에서만 제공됩니다.

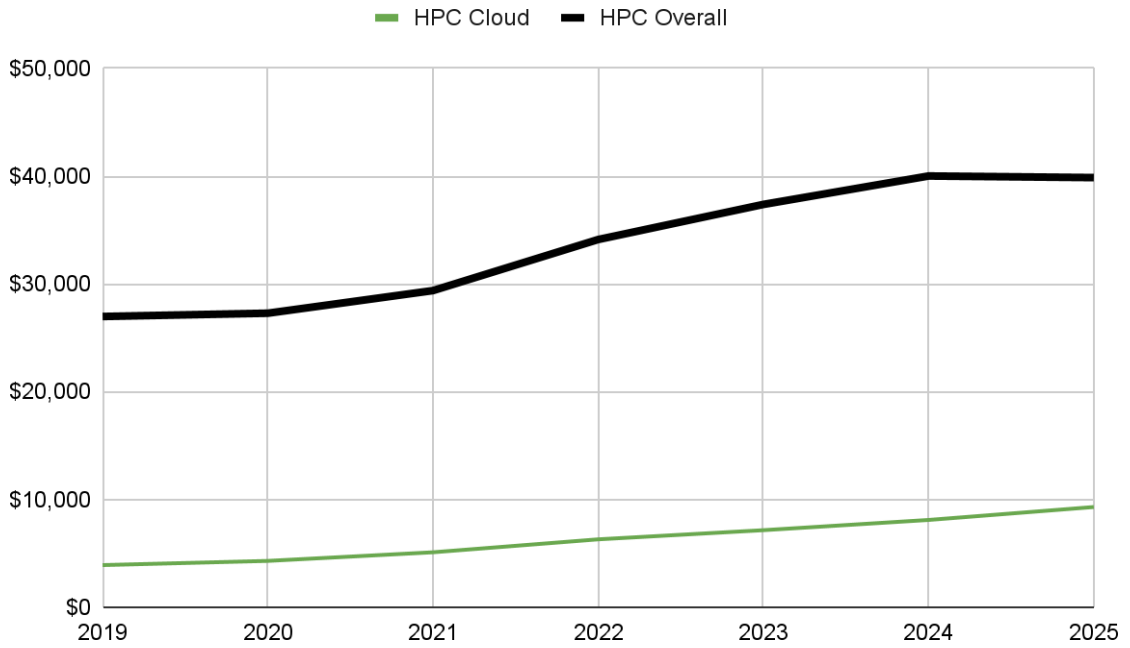
다각화는 새롭고 잠재력이 높은 패러다임이 등장했지만, 기술 표준은 아직 등장하지 않았을 때 발생합니다. 많은 공급업체들이 이를 기회로 삼아 솔루션을 제공하는 등 새로운 패러다임에 뛰어들면서, HPC 비즈니스라는 도가니에서 가장 효율적이고 가치 있는 제품을 제외한 모든 제품이 타서 사라지는 소멸의 과정이 발생하고 있습니다.

초창기에는 대부분의 HPC 사례에서 클라우드가 그리 심각하게 고려되지 않았습니니다. 네트워크는 이더넷을, 스토리지는 CIFS(Common Internet File System) 또는 NFS(Network File System)를 기반으로 삼았고, 코어는 하이퍼스레드로 분할되었습니다. [NASA의 한 논문](#)에서 클라우드 컴퓨팅 기술이 HPC 워크로드를 고려할만큼 제대로 설계되지 않았고, GFLOP당 비용도 너무 비싸다는 결론을 내리면서 전반적으로 회의론적인 분위기가 조성되었습니다.

10년이 흐르고 하드웨어와 소프트웨어가 몇 세대를 거친 지금, 몇몇 클라우드 서버의 시설과 기능은 당연히 HPTC(High-Performance Throughput Computing) 태스크를 처리하기에 이르렀고, 일부는 세분화되고 매우 민감한 병렬 워크로드를 겨냥하고 있습니다. [Microsoft Azure](#) 기반의 가상 슈퍼 컴퓨터가 세계에서 가장 빠른 [10대 컴퓨터에 진입했던](#) 2021년 11월 Top500 목록이 이러한 현실을 보여주고 있습니다. HPC 커뮤니티의 일부 사람들을 크게 당황시킨 것은 이 슈퍼 컴퓨터가 Hyper-V 기반의 가상 머신으로만 이루어져 있다는 사실이었습니다!

[Hyperion은 Supercomputing 2021 브리핑](#)을 통해 클라우드 기반 슈퍼 컴퓨팅이 진지하게 받아들여지고 있음을 다시 한번 입증했습니다. 이 브리핑에서는 2025년까지 HPC의 25%가 클라우드에서 실행될 것으로 예측하고 있습니다.

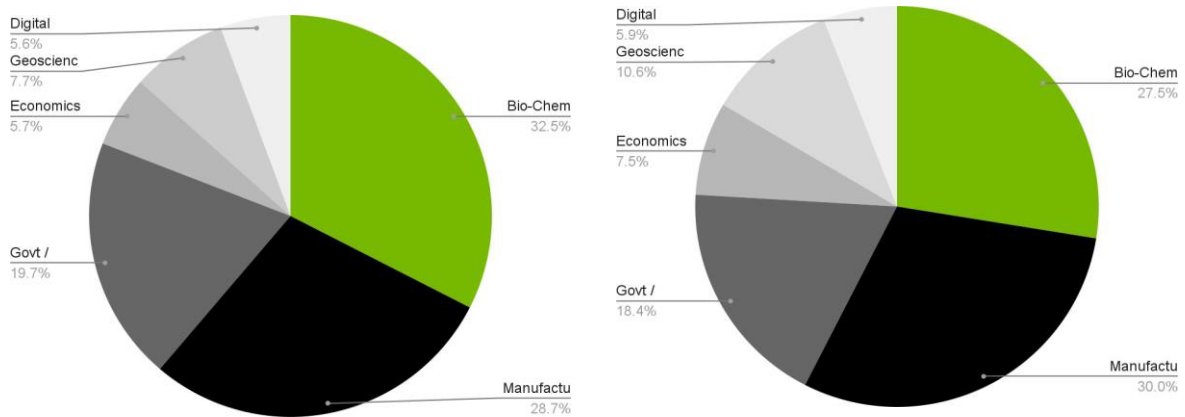
그림 25. Hyperion: HPC 클라우드 및 HPC의 전반적 성장



Hyperion은 HPC의 클라우드 사용량이 꾸준히 증가하여 2025년이면 전체 사용량의 25%에 육박할 것으로 내다보고 있습니다.

회의론자들은 당연히 그 다음 질문을 던질 것입니다. "그럼 지금은 누가 클라우드에서 HPC를 사용하고 있습니까?"

그림 26. 수직 시장별 Hyperion HPC



다시 말하지만, 그림 26의 파이 차트에 나와 있는 것처럼 Hyperion은 NVIDIA를 위해 이에 대한 특성 분석을 수행했습니다. 현재 HPC 클라우드 사용량의 3분의 2는 항공우주, 자동차, **신약 개발 및 연구**, 전산 화학, 재료 과학 등의 분야를 위한 제조 활동에 사용되고 있습니다. 이러한 상업적 기업이 HPC를 클라우드로 전환하는 데 앞장서고 있다는 것은 흥미로운 사실이 아닐 수 없습니다. 클라우드 컴퓨팅 사용을 오랜 세월 반대해온 이유 중 하나가 바로 보안이었기 때문입니다. 기업들은 실제 구내 밖에 있는 HPC에서 수행되는 지적 재산 생성 활동을 사용하는 데 불편을 겪고 있었습니다. 통계(그림 26)에서 알 수 있듯이, 클라우드로서 HPC를 수행하는 비즈니스 사례로 인해 이러한 흐름이 바뀌었습니다.

클라우드에서 HPC를 지지하는 사람들은 "클라우드 HPC는 시간보다 돈이 많은 사람들을 위한 것"이라고 버릇처럼 말합니다. 클라우드에서 실행하는 것이 로컬 HPC 리소스가 존재하고 사용 가능할 경우에 이를 사용하는 것보다 비용이 많이 든다는 것을 인정하는 말입니다. 또한 비즈니스 사례가 HPC에서 클라우드를 사용하기에 적합한 경우에는 "버스팅"이라는 개념을 살짝 언급하기도 합니다. 기한이 임박했거나 추가 단기 용량이 절실히 필요한 경우에 가장 좋은 해결책은 클라우드 컴퓨팅 리소스를 거의 즉각적으로 사용할 수 있게 하는 것입니다.

HPC 사용자들의 입장이 갈리고 있는 또 다른 주제는 바로 컨테이너화입니다. 가상 머신과 마찬가지로, 컨테이너는 몇몇 엔터프라이즈 및 백오피스 워크로드와 더불어 AI와 ML의 실질적인 표준이 되었습니다. 그러나 본래의 컨테이너 기술인 Docker의 보안에 대한 우려가 커지고 애플리케이션과 하드웨어 사이에 소프트웨어 레이어를 배치하는 것에 대한 우려가 생기면서 전 세계 슈퍼 컴퓨팅 센터들은 컨테이너 사용에 대해 입장이 갈리게 되었습니다.

컨테이너는 사용자 및 관리 측면에서 베어메탈에서 실행할 때와 비교해 몇 가지 중요한 이점이 있지만, 단점도 분명히 존재합니다. 표 4에는 HPC 애플리케이션에서 컨테이너를 사용할 때의 장점과 단점이 나와 있습니다.

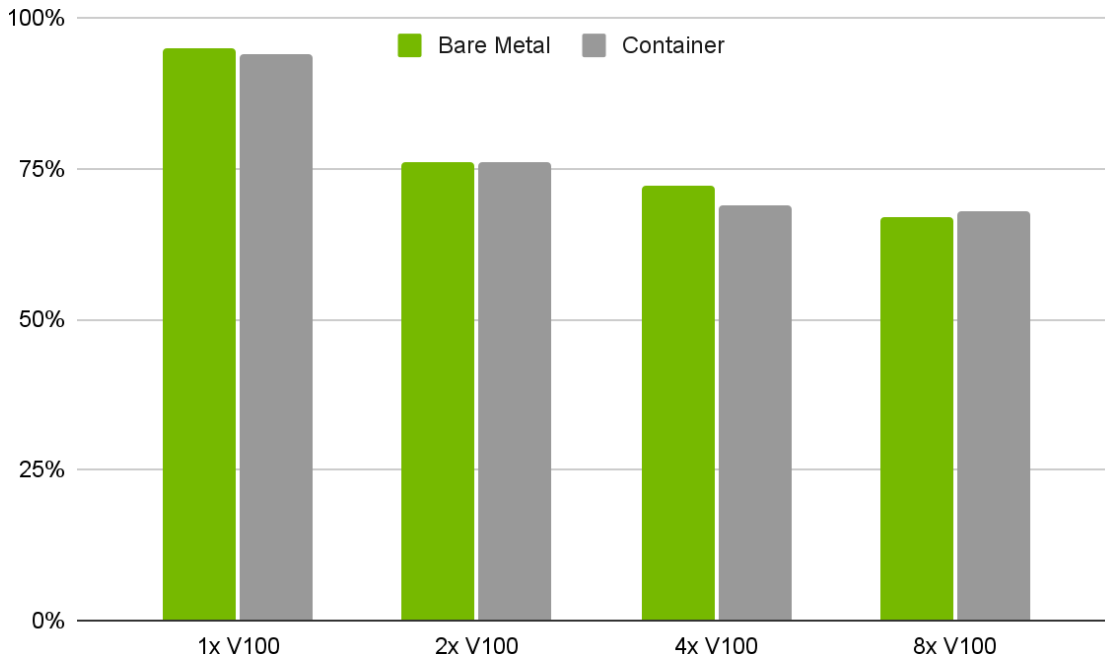
표 4. HPC 애플리케이션에서 컨테이너의 장점과 단점

장 점	단 점
태스크 분리	과학자를 위한 구축에 어려움
유연한 앱 버전	과학자를 위한 맞춤화에 어려움
신속한 시스템 마이그레이션	넓은 공간(크기) 차지
결과 재현성	보안성

[특이성\(Singularity\)](#) 덕분에 Docker에서의 보안 및 다중 노드 MPI 이슈가 컨테이너로의 전환 과정 초기에 해결되었지만, 성능 문제는 계속되고 있습니다. NVIDIA 기술은 베어메탈, 가상 머신 및 컨테이너에서 작동하지만, 기본적으로 NVIDIA는 [컨테이너 기술을 지지하고 있습니다](#). 또한 위의 섹션에서 설명했듯이, HPC와 AI/ML가 결합되면서 상호 운용이 필요한 것으로 보입니다. 컨테이너는 AI/ML의 표준이기 때문에 컨테이너에서 HPC가 지원되면 이러한 병합이 더욱 촉진될 것입니다.

그림 27에는 동일한 서버에서 실행 중인 공통 HPC 애플리케이션의 차트가 나와 있습니다. 하나는 베어메탈에서 실행되고, 다른 하나는 컨테이너에서 실행되고 있습니다. 둘 간에는 성능 차이가 없다는 점에 유의하시기 바랍니다.

그림 27. MILC HPC 애플리케이션에서 컨테이너와 베어메탈 비교



결론

이 전자책의 내용이 HPC 시스템 설계의 고려 사항을 이해하고, 가까운 미래에 HPC 시스템을 형성하게 될 현재 및 향후 기술 동향을 파악하는 데 도움이 되었기를 바랍니다. 더 나아가, 애플리케이션 지원 및 HPC 애플리케이션 성능 최적화에 대한 추가적인 논의가 HPC가 목적이 아닌 수단이라는 점을 이해하는 데 있어 이 문서의 핵심 내용을 뒷받침하는 근거가 될 수 있기를 바랍니다.

여기서 각각의 영역을 언급한 이유는 진화와 성장을 거듭하는 과학으로 간주해야 하는 기술이 무엇인지를 보여주기 위한 것이었습니다. 양자 컴퓨팅에서부터 슈퍼 컴퓨터에서의 AI 배포에 이르기까지 HPC가 필요한 사례는 끝이 없으며, 개별 시스템의 성능을 높이는 과정은 동적입니다. HPC 연구와 SDK가 확장되고 있기 때문에 NVIDIA [HPC 홈페이지](#)에서 추가 업데이트 소식을 주기적으로 확인하시기 바랍니다.